INTERNATIONAL TELECOMMUNICATION UNION

# ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

# X.638

(10/96)

SERIES X: DATA NETWORKS AND OPEN SYSTEM COMMUNICATION

OSI networking and system aspects – Efficiency

---

# Minimal OSI facilities to support basic communications applications

ITU-T Recommendation X.638

(Previously CCITT Recommendation)

---

ITU-T X-SERIES RECOMMENDATIONS

**DATA NETWORKS AND OPEN SYSTEM COMMUNICATION**

| | |
|---|---|
| PUBLIC DATA NETWORKS | X.1–X.199 |
| Services and facilities | X.1–X.19 |
| Interfaces | X.20–X.49 |
| Transmission, signalling and switching | X.50–X.89 |
| Network aspects | X.90–X.149 |
| Maintenance | X.150–X.179 |
| Administrative arrangements | X.180–X.199 |
| OPEN SYSTEM INTERCONNECTION | X.200–X.299 |
| Model and notation | X.200–X.209 |
| Service definitions | X.210–X.219 |
| Connection-mode protocol specifications | X.220–X.229 |
| Connectionless-mode protocol specifications | X.230–X.239 |
| PICS proformas | X.240–X.259 |
| Protocol Identification | X.260–X.269 |
| Security Protocols | X.270–X.279 |
| Layer Managed Objects | X.280–X.289 |
| Conformance testing | X.290–X.299 |
| INTERWORKING BETWEEN NETWORKS | X.300–X.399 |
| General | X.300–X.349 |
| Satellite data transmission systems | X.350–X.399 |
| MESSAGE HANDLING SYSTEMS | X.400–X.499 |
| DIRECTORY | X.500–X.599 |
| OSI NETWORKING AND SYSTEM ASPECTS | X.600–X.699 |
| Networking | X.600–X.629 |
| **Efficiency** | **X.630–X.649** |
| Naming, Addressing and Registration | X.650–X.679 |
| Abstract Syntax Notation One (ASN.1) | X.680–X.699 |
| OSI MANAGEMENT | X.700–X.799 |
| Systems Management framework and architecture | X.700–X.709 |
| Management Communication Service and Protocol | X.710–X.719 |
| Structure of Management Information | X.720–X.729 |
| Management functions | X.730–X.799 |
| SECURITY | X.800–X.849 |
| OSI APPLICATIONS | X.850–X.899 |
| Commitment, Concurrency and Recovery | X.850–X.859 |
| Transaction processing | X.860–X.879 |
| Remote operations | X.880–X.899 |
| OPEN DISTRIBUTED PROCESSING | X.900–X.999 |

*For further details, please refer to ITU-T List of Recommendations.*

# FOREWORD

The ITU-T (Telecommunication Standardization Sector) is a permanent organ of the International Telecommunication Union (ITU). The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, March 1-12, 1993).

ITU-T Recommendation X.638 was prepared by ITU-T Study Group 7 (1993-1996) and was approved under the WTSC Resolution No. 1 procedure on the 5th of October 1996.

---

NOTE

In this Recommendation, the expression “Administration” is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

## CONTENTS

*Page*

## SUMMARY

This Recommendation provides the concepts, definitions and profile of the minimal upper layer facilities necessary to support basic connection-oriented communications applications. These facilities consist of a subset of the facilities defined in the ACSE, and the Presentation and Session service definitions. A basic communications application simply requires the ability to open and close communications with a peer and to send and receive messages with that peer. It is expected that a large portion of OSI applications will be satisfied by such minimal functionality.

This is one of a series of Recommendations to support the minimal OSI functions in the upper layers. It is technically aligned with ISO/IEC ISP 11188-3.

# MINIMAL OSI FACILITIES TO SUPPORT BASIC COMMUNICATIONS APPLICATIONS

*(Geneva, 1996)*

## 1 Scope

This Recommendation introduces the concept of the minimal set of OSI upper layer facilities for Basic Communications Applications (BCA). The upper layer facilities considered in this Recommendation are encompassed by the existing ACSE, Presentation and Session Recommendations. A basic communications application simply requires the ability to open and close connection-oriented 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 satisfied by such minimal functionality.

## 2 Normative References

The following Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision: all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

### 2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model.*
- ITU-T Recommendation X.207 (1993) | ISO/IEC 9545:1994, *Information technology – Open Systems Interconnection – Application layer structure.*
- ITU-T Recommendation X.210 (1993) | ISO/IEC 10731:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: Conventions for the definition of OSI services.*
- ITU-T Recommendation X.214 (1995) | ISO/IEC 8072:1996, *Information technology – Open Systems Interconnection – Transport service definition.*
- ITU-T Recommendation X.215 (1995) | ISO/IEC 8326:1996, *Information technology – Open Systems Interconnection – Session service definition.*
- ITU-T Recommendation X.216 (1994) | ISO/IEC 8822:1994, *Information technology – Open Systems Interconnection – Presentation service definition.*
- ITU-T Recommendation X.217 (1995) | ISO/IEC 8649:1996, *Information technology – Open Systems Interconnection – Service definition for the association control service element.*
- ITU-T Recommendation X.225 (1995) | ISO/IEC 8327-1:1996, *Information technology – Open Systems Interconnection – Connection-oriented session protocol: Protocol specification.*
- ITU-T Recommendation X.226 (1994) | ISO/IEC 8823-1:1994, *Information technology – Open Systems Interconnection – Connection-oriented presentation protocol: Protocol specification.*
- ITU-T Recommendation X.227 (1995) | ISO/IEC 8650-1:1996, *Information technology – Open Systems Interconnection – Connection-oriented protocol for the association control service element: Protocol specification.*
- ITU-T Recommendation X.245 (1995) | ISO/IEC 8327-2:1996, *Information technology – Open Systems Interconnection – Connection-oriented session protocol: Protocol Implementation Conformance Statement (PICS) proforma.*

– ITU-T Recommendation X.246 (1996) | ISO/IEC 8823-2:1997, *Information technology – Open Systems Interconnection – Connection-oriented presentation protocol: Protocol Implementation Conformance Statement (PICS) proforma.*

– ITU-T Recommendation X.247 (1996) | ISO/IEC 8650-2:1997, *Information technology – Open Systems Interconnection – Protocol specification for the association control service element: Protocol Implementation Conformance Statement (PICS) proforma.*

– ITU-T Recommendation X.690 (1994) | ISO/IEC 8825-1:1995, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*

## 2.2 Paired Recommendations | International Standards equivalent in technical content

– CCITT Recommendation X.650 (1992), *Open Systems Interconnection (OSI) – Reference model for naming and addressing.*

ISO 7498-3:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 3: Naming and addressing.*

– ITU-T Recommendation X.637 (1996), *Basic connection-oriented common upper layer requirements.*

ISO/IEC ISP 11188-1:1995, *Information technology – International Standardized Profile – Common upper layer requirements – Part 1: Basic connection oriented requirements.*

## 2.3 Additional references

– ISO/IEC TR 10000-1:1995, *Information technology – Framework and taxonomy of International Standardized Profiles – Part 1: General principles and documentation framework.*

– ISO/IEC TR 10000-2:1995, *Information technology – Framework and taxonomy of International Standardized Profiles – Part 2: Principles and taxonomy of OSI profiles.*

– Internet RFC 1698, *Octet Sequences for Upper-Layer OSI to support Basic Communications Applications.*

# 3 Definitions

This Recommendation makes use of the following definitions.

## 3.1 Reference model definitions

### 3.1.1 Basic Reference Model definitions

This Recommendation 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) conformance;

i) presentation-connection;

j) Presentation Layer;

k) presentation-service;

l) session-connection;

m) Session Layer;

n) session-protocol;

o) session-service;

p) Transport Layer.

### 3.1.2 Naming and addressing definitions

This Recommendation makes use of the following terms defined in CCITT Rec. X.650 | 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.

## 3.2 Service conventions definitions

This Recommendation makes use of the following terms defined in ITU-T Rec. X.210 | ISO/IEC 10731:

a) primitive;

b) request (primitive);

c) indication (primitive);

d) response (primitive); and

e) confirm (primitive).

## 3.3 Presentation definitions

This Recommendation 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:

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.

## 3.4 Session definitions

This Recommendation 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:

– session selector.

## 3.5 Application Layer Structure definitions

This Recommendation makes use of the following terms defined in ITU-T Rec. X.207 | ISO/IEC 9545:

a) application-context;

b) application-entity invocation;

c) control function; and

d) application-service object.

## 3.6 ACSE service definitions

This Recommendation makes use of the following terms defined in ITU-T Rec. X.217 | ISO/IEC 8649:

a) application-association; association;

b) Association Control Service Element;

c) requestor;

d) acceptor;

e) association-initiator;

f) association-responder.

## 3.7 Taxonomy of profile definitions

This Recommendation 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.

## 3.8 Definitions of this Recommendation

**3.8.1 API specification; application programmatic interface specification**: The functional specification of the local manifestation of the facilities of an identified stack specification. An API is normally defined as a set of procedure calls in a particular programming language.

**3.8.2 API; application programmatic interface**: An implementation of an identified API specification.

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

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

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

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

**3.8.7 mOSI platform**: An implementation of the mOSI platform specification.

**3.8.8 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.

**3.8.9 platform**: An implementation of an identified platform specification.

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

**3.8.11 PDV-processor**: Part of an implementation which wraps and unwraps the “pdv envelope” around the syntax sent and received in the identified presentation context.

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

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

**3.8.14 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.

**3.8.15 specific basic communications application**: An application that is not referenced by any ISP.

**3.8.16 stack; stack implementation**: An implementation of an identified stack specification.

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

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

**3.8.19 transport-provider**: A provider of those transport services which are defined in ITU-T Rec. X.214 | ISO/IEC 8072.

## 4 Abbreviations

The following abbreviations are used in this Recommendation:

| | |
|---|---|
| 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 |
| CCR | Commitment, Concurrency and Recovery |
| CULR | Common Upper Layers Requirements |
| FTAM | File Transfer, Access and Management |
| ICS | Implementation Conformance Statement |
| IEC | International Electrotechnical Commission |
| ISO | International Organization for Standardization |
| ISP | International Standardized Profile |
| ITU-T | International Telecommunication Union – Telecommunication Standardization Sector |
| mOSI | Minimal OSI Upper Layer Facilities |
| MTA | Message Transfer Agent |
| OSI | Open Systems Interconnection |
| PDU | protocol-data-unit |
| PDV | Presentation Data Value |
| PICS | Protocol Implementation Conformance Statement |
| POSIX | Portable Operating System Interface for Computer Environments |
| PPDU | presentation-protocol-data-unit |
| PRL | Profile Requirements List |
| RTSE | Reliable Transfer Service Element |
| SPDU | session-protocol-data-unit |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| TP | Transaction Processing |
| TSDU | transport-service-data-unit |
| VTP | Virtual Terminal Protocol |

# 5 Introduction

This clause introduces the concept of minimal OSI and gives definitions.

## 5.1 Minimal OSI upper layer facilities

Minimal OSI (mOSI) is a minimal set of OSI upper layer facilities that supports Basic Communications Applications (BCAs). A **Basic Communications Application (BCA)** simply provides functions to open and close communications with a peer and to send and receive messages with that peer.

The OSI upper layer facilities employed are defined by the existing Association Control Service Element 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 called the **mOSI specification** or the **mOSI stack specification** and is described in this Recommendation.

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

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

## 5.2 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 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.

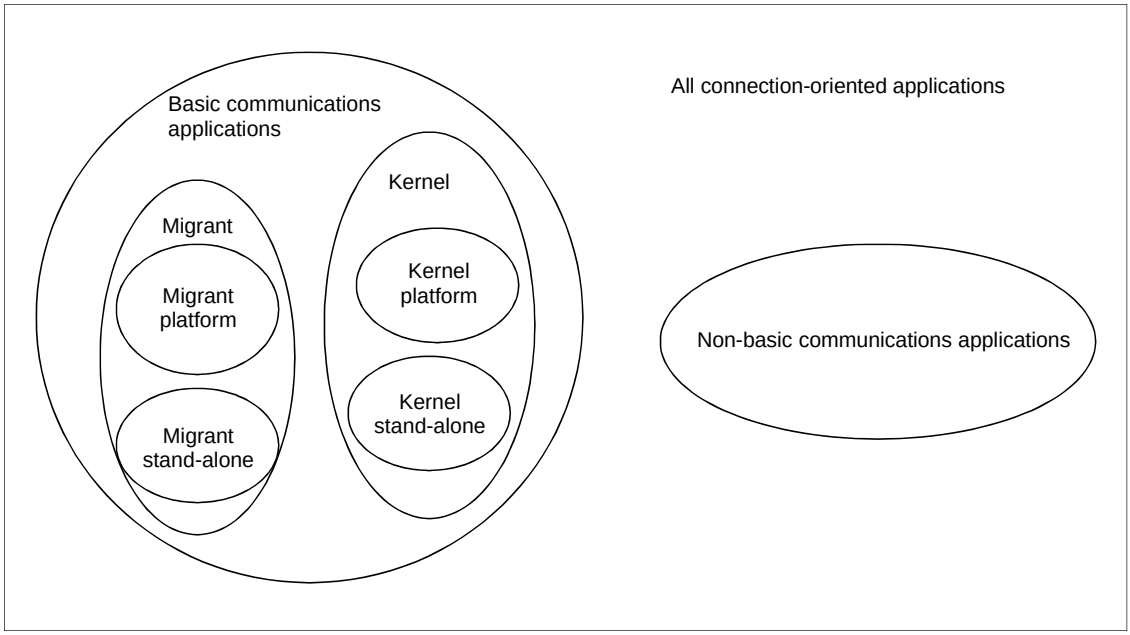


FIGURE 1/X.638

**Universe of all OSI-based connection-oriented applications**

### 5.2.1 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 than basic communications services.

### 5.2.2 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.

### 5.2.3 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-Windows system (X) and IPS applications are examples of potential migrant applications.

#### 5.2.3.1 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;

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

d) VTP without destructive interrupt facility;

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**.

#### 5.2.3.2 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;

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.

### 5.3 Users of the mOSI stack specification

This Recommendation is intended to address implementors of migrant and kernel applications using either a platform or accessing the mOSI stack directly (stand-alone users). This Recommendation is also intended to be referenced by International Standard Profile ISP designers.

### 5.4 OSI upper layer efficiency

A full function upper layer stack and its API are intended for those applications that need more than the basic communications services, i.e. applications such as the MTA-to-MTA transfer of X.400 messages, distributed Transaction Processing with 2-phase commit, etc. 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.

However, migrant and kernel applications, which represent the overwhelming majority of potential OSI applications, require scarcely more than 5% of the overall functionality that a full upper layer stack can provide.

Having an API which only provides mapping onto these required 5% of the overall functionality would make such OSI applications easier to implement and far more efficient to operate. Any perception that OSI is bound to be complex and inefficient is based entirely on the erroneous assumption that every OSI application must be mapped onto a full function upper layer stack.

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.

## 6 The mOSI model

This clause presents the mOSI model and defines some additional terms used in this Recommendation. The mOSI model, as shown in Figure 2, illustrates the mOSI stack in three different environments.

It can be viewed in two contexts:

a) abstractly – where the various elements represent abstract “specifications”; or

b) concretely – where the elements represent those of an implementation.

### 6.1 Common elements

There are common elements in all three environments shown in Figure 2. They are:

– basic communications application;

– pdv-processor;

– mOSI stack;

– transport services; and

– transport provider.

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 Recommendation.

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 an mOSI stack is one protocol engine, not three.

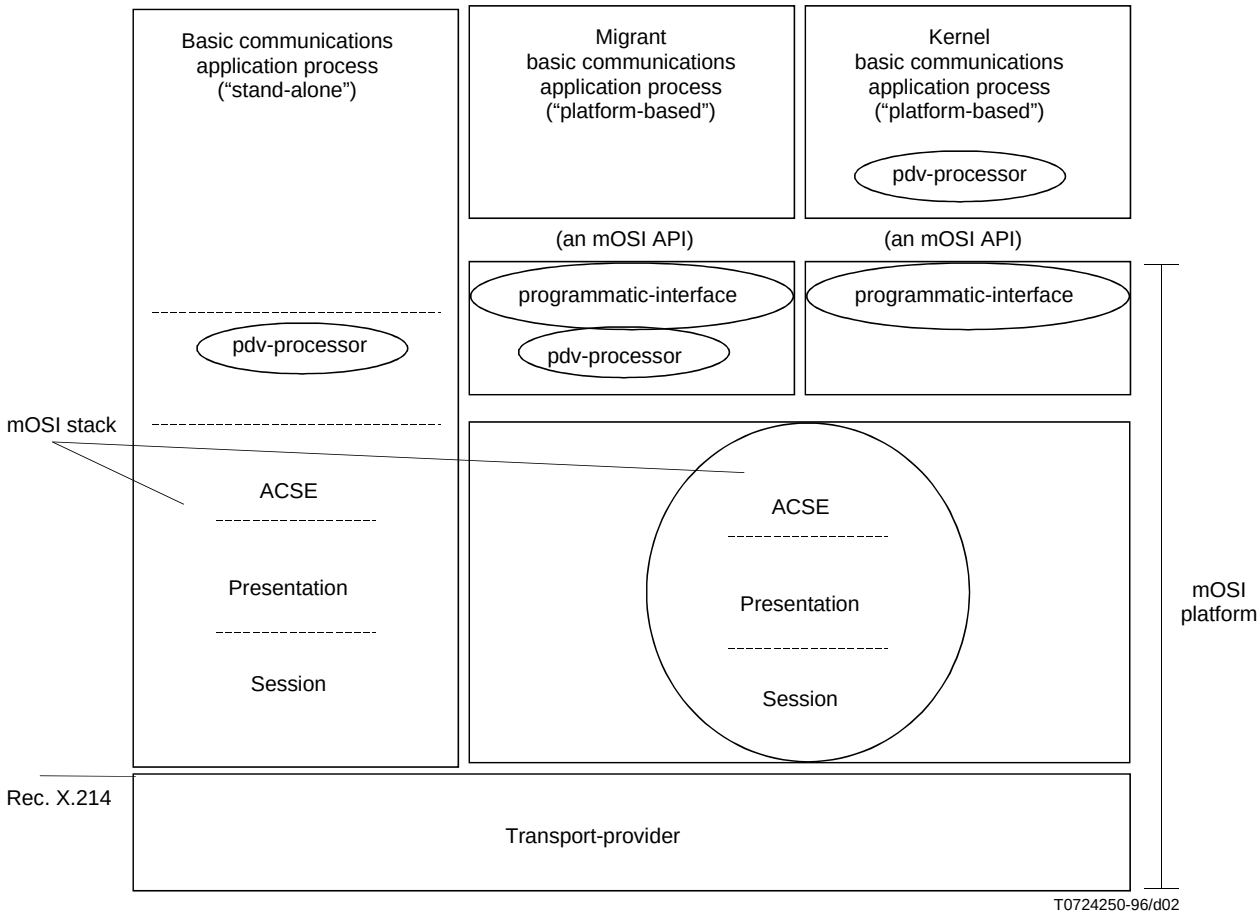


FIGURE 2/X.638

**mOSI model**

From the perspective of the presentation protocol (see ITU-T Rec. X.226 | 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 presentation protocol defines the encoding of the outer envelope around a pdv and the encoding for groups of pdv's (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 considered in the OSI model to be done by the Presentation Layer. The Application Layer is concerned only with abstract values, with no defined encoding. Consequently, in a real system the boundary between Application Layer and Presentation Layer cannot be precisely represented (since real systems are not abstract). In the implementation-oriented mOSI model, any necessary encoding/decoding of the transfer syntax sent between connected applications is outside the scope of the mOSI stack. The encoding produces presentation-data-values in the appropriate presentation context. The Presentation protocol includes the specification of a pdv envelope. 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 2, 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[1)].

1) This does not preclude sending a Presentation provider Abort (ARP-PPDU).

This Recommendation does not address the four lower OSI layers (Transport, Network, Link, and Physical Layers). They are considered outside the scope of this Recommendation. However, a **transport-provider** is needed to transport the ACSE, presentation, and session PDUs of an mOSI implementation. As such, the transport-provider supplies **transport services** equivalent to those defined in the OSI Transport Layer service definition (see ITU-T Rec. X.214 | ISO/IEC 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.

## 6.2 Applications

### 6.2.1 Standalone applications

A **standalone application** is one that includes the application pdv-processor and the mOSI stack as a single unit application[2]. In an implementation, the mOSI stack may be a series of separate modules with its own internal programmatic interface or as a single state machine.

### 6.2.2 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.

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

An **mOSI API** represents the interface to the mOSI stack. It provides the minimal features of the OSI upper layers as defined in this Recommendation.

## 6.3 Types of basic communications applications

mOSI identifies two types of basic communications: **migrant** applications and **kernel** applications (see 5.2.3.1). Depending on the type of the application, the pdv-processor could either be a part of the platform or a part of each platform-based application.

### 6.3.1 Migrant applications

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

a) application context[3]; and

b) presentation context[4] – 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 5.2.3.2) 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.

---

[2] Many ISP are written from the point of view of standalone applications. However, the actual implementation of the ISP could result in a platform-based application.

[3] See ITU-T Rec. X.207 | ISO/IEC 9545 for details.

[4] See ITU-T Rec. X.226 | ISO/IEC 8823-1 for details.

### 6.3.2 Kernel application

A kernel application (see 5.2.3.1) 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 2 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.

# 7 Details of the mOSI concept

## 7.1 Kernel and options

**mOSI** consists of the Kernel functional units of ACSE, presentation, and session plus the session Duplex functional unit. To this, two optional sets of features may be added:

– **with authentication** (the ACSE Authentication functional unit); and

– **with application context negotiation** (the ACSE Application Context Negotiation functional unit).

### 7.1.1 Association establishment

For association establishment, the following roles are possible (see Annex D):

a) association initiator only; or

b) association responder only; or

c) both association initiator and association responder.

For the purposes of this Recommendation, this set of roles is expressed by the variable *Establishment-role*. The variable may assume one of the following values: "initiator", or "responder", or "both." This variable is used in Annexes A, B, and C to define conditionally the requirements of the PICS of ACSE, Presentation, and Session.

## 7.2 Normal data transfer

For normal data transfer, the following roles are possible (see Annex D):

a) normal data requestor only; or

b) normal data acceptor only; or

c) both normal data requestor and acceptor; or

d) neither normal data requestor nor acceptor.

For the purposes of this Recommendation, this set of roles is expressed by the variable *Normal-data-role*. The variable may either be null or it may assume one of the following values: "requestor", or "acceptor", or "both." The variable is used in Annexes B and C to define conditionally the PICS of Presentation and Session.

## 7.3 Association release

For association release, the following roles are possible (see Annex D):

a) release-requestor only; or

b) release-acceptor only; or

c) both release-requestor and release-acceptor; or

d) neither release-requestor nor release-acceptor.

For the purposes of this Recommendation, this set of roles is expressed by the variable *Release-role*. The variable may either be null or it may assume one of the following values: "requestor", or "acceptor", or "both." The variable is used in Annexes A and C to define conditionally the PICS for ACSE and Session.

## 7.4 Relationship to base standards

### 7.4.1 ACSE

This Recommendation includes rules for incorporating the Kernel functional unit and, optionally, the Authentication functional unit and Application Context Name Negotiation functional unit. It also supports the roles for association establishment and release identified in ITU-T Rec. X.227 | ISO/IEC 8650-1.

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 Recommendation X.637 also apply.

### 7.4.2 Presentation Layer

This Recommendation includes rules for incorporating the Presentation Kernel functional unit.

The required facilities of Presentation 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 Recommendation X.637 also apply.

### 7.4.3 Session Layer

This Recommendation includes rules for incorporating the Session Kernel and Duplex functional units.

The required facilities of session are specified in Annex C.

The requirements expressed in Recommendation X.637 also apply.

## 7.5 Transport-provider

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

A transport-provider is needed to support the exchange of ACSE, Presentation, and Session PDUs of an mOSI implementation. The transport-provider must supply services equivalent to those defined in the OSI Transport Layer service definition (ITU-T Rec. X.214 | ISO/IEC 8072).

# 8 Application of this Recommendation

## 8.1 Types of user

This Recommendation defines the mOSI facilities in terms of identified features of the upper layer PICS proformas – the ACSE (ISO/IEC 8650-2), the Presentation Layer (see ITU-T Rec. X.246 | ISO/IEC 8823-2), and the Session Layer (see ITU-T Rec. X.247 | ISO/IEC 8327-2).

The identified features of these PICS proformas are specified in Annexes A, B, and C, respectively.

This Recommendation may be referenced by two classes of entities, upper layer *users* and upper layer *providers*:

- mOSI *users* represent basic communications applications. mOSI users may be profiles (such as A-profiles defined 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 Recommendation to represent this set of mOSI users.
- mOSI *providers* represent implementations of the upper layer facilities that provide (at a minimum) the facilities defined in this Recommendation. The term **referencing implementation** is used in this Recommendation to represent this set of mOSI providers.

A profile (ref sp) or a bca (an mOSI user) may claim *compliance*[5] to this Recommendation if the OSI upper layer facilities that it requires may be expressed by those facilities defined in this Recommendation. Subclause 11.1 summarizes the requirements for making such a statement and Annex D provides the proforma for the requirements compliance statement.

A referencing implementation of the OSI upper layers (an mOSI provider) may claim *conformance*[6] to this Recommendation if the OSI upper layer facilities that it provides include the facilities defined in this Recommendation. That is, an implementation may contain more upper layer facilities than those required to be conformant to this Recommendation. However, they must contain at least those of this Recommendation. Subclause 11.2 summarizes the requirements for making a conformance statement. Annex E provides the proforma for the profile implementation statement.

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

## 8.2 Conventions

### 8.2.1 Profile status indicators

This Recommendation states mOSI functionality by defining rules for forming an mOSI Profile Requirements List (PRL) based on the PICS proformas of ACSE, Presentation, and Session. The rules for defining an 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 Recommendation 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 9.3 and 8.2.2). An open parameter is a variable that is only referenced in one table entry. An open parameter is assigned the profile status of "*" (see Table 1).

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 PRL.

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 an 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.

---

5) **Compliance** deals with one *specification* referencing another specification; **conformance** deals with a *physical implementation* that references another specification.

6) *Ibid*.

TABLE 1/X.638

**Profile status identifiers**

| | Identifier | Meaning when referenced by a specification | Meaning when referenced by an implementation |
|---|---|---|---|
| 1 | m | Mandatory – The feature shall be required for support. The status of the feature shall remain mandatory in a referencing specification (see Note). | Mandatory – The implementation shall support the feature, i.e. its syntax and procedures shall be implemented as specified in the base standard or in Rec. X.637. 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 Rec. X.637. 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 not support the feature:<br>• Supporting the feature means that the feature shall be handled as if it was mandatory.<br>• Not supporting a feature depends on receiving or sending:<br>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).<br>The feature shall be the subject of an ISP conformance test. |
| 3 | * | Open – 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 – Same as optional. |
| 4 | x | Excluded – The feature shall not be used in a referencing specification. The status of the feature shall remain excluded in a referencing specification. | Excluded – 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 – The requirement for the support of this feature is not covered by this Recommendation. The status of the feature shall remain out of scope in a referencing specification. | Out of scope – 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 – 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 – 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 – Support for the feature is further defined in this Recommendation 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 – 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:<br>– the feature has been implemented;<br>– the feature has not been implemented; or<br>– not applicable. |
| | NOTE – 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. | | |

### 8.2.2 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. A variable (for the purpose of this Recommendation) is used as a simple replacement for two or more table entries and/or in an IF/THEN/ELSE conditional statement to derive a table entry. A variable's name is always written in *italics*. The variables used are:

- *Establishment-initiator*: Capability to establish an association or connection.
- *Establishment-responder*: Capability to respond to an establishment initiation.
- *Establishment-responder-reject*: Capability to reject an establishment initiation.
- *Release-requestor*: Capability to release an association or connection.
- *Release-acceptor*: Capability to accept a release request.
- *Normal-data-requestor*: Capability to request normal data transfer.
- *Normal-data-acceptor*: Capability to accept normal data transfer.
- *Authentication*: Capability to perform authentication.
- *Application-context-negotiation*: 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 9.3. 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.

### 8.2.3 Definition of mnemonics

Mnemonics are used in Annexes A, B, and C to facilitate the interpretation of conditional statements throughout the PRLs of Annexes A, B, and C. These mnemonics are introduced in column 4 of several tables in these annexes. They are:

- *A-CON-initiator* (see A.2.1/1).
- *A-CON-responder* (see A.2.1/2).
- *A-REL-requestor* (see A.2.2/1).
- *A-REL-acceptor* (see A.2.2/2).
- *A-FU(AU)* (see A.4/2).
- *A-FU(ACN)* (see A.4/3).
- *P-DATA-requestor* (see B.3.1.1.2/1).
- *P-DATA-acceptor* (see B.3.1.1.2/2).
- *P-REL-requestor* (see B.3.1.1.3/1).

– *P-REL-acceptor* (see B.3.1.1.3/2).

– *P-REFUSE* (see B.3.1.2/3).

– *S-EXP-T* (see C.3.2/1).

These mnemonics are Boolean variables. They reflect whether the feature of the corresponding table entry is supported or not. A mnemonic has the value TRUE if the corresponding feature is supported, otherwise FALSE.

# 9 Compliance and conformance

## 9.1 Profile or specification of a basic communications application

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

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

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

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

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

d) require that all of this Recommendation'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 Recommendation X.637 and not conflict with the requirements of this Recommendation[7)].

NOTE 1 – The meaning of the status indicators (see 8.2.1) when used by a referencing specification is given in Table 1, column 3.

NOTE 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 Recommendation by either of the following ways:

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

b) It may claim compliance by reference to this Recommendation instead of repeating the provisions of this Recomendation.

NOTE 3 – It is recommended that a referencing specification claim compliance with this Recommendation by referencing it instead of repeating their provisions.

## 9.2 OSI upper layer stack implementation

An implementation may reference this Recommendation to claim that it supports some or all of the features specified in this Recommendation. The referencing implementation may in fact support additional upper layer features without violating any of those in this Recommendation.

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

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

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

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

---

7) See clause 8 and ITU-T Rec. X.637 and ISO/IEC ISP 11188-1, Annex B.

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

e) conform to the requirements of Recommendation X.637.

NOTE – It is recommended that a referencing implementation use the tables in Annex E to specify its profile implementation conformance statement.

### 9.3 Facilities, roles and options

This Recommendation defines **mOSI compliance and conformance** in terms of facilities, roles and options. This Recommendation 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 Recommendation, each role (or an optional facility) is referenced by a variable (see 8.2.1). 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:

– mandatory ("m");

– optional ("o");

– 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.

## 10 Implementation considerations

This Recommendation 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 clause makes several suggestions about the implementation of the mOSI specified subset of the OSI's 3 upper layers. Experience has shown that this implementation approach yields smaller and faster implementations – especially when compared to some of the well known OSI proof-of-concept implementations.

### 10.1 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 there are no formally defined interfaces 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.

## 10.2 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 Internet RFC 1698 on "Octet Sequences for Upper-Layer OSI to support Basic Communications Applications"[8)] 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 PDUs.

## 10.3 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 and these may be 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.

## 10.4 Interfaces for mOSI implementations

Portability for an mOSI implementation can be significantly enhanced through the use of X/Open's XTI interfaces. XTI provides a generic interface for connect/disconnect and send/receive. The XTI interface is explicitly for bca. The XTI for mOSI interface[9)] provides a common interface for networked applications. It simplifies migration of networked applications from one open networking environment to another.

The standard XTI can be used to provide transport services to an mOSI implementation, thus making an mOSI implementation easily portable to any network transport supporting XTI, e.g. Internet, NetBIOS, and a number of proprietary networks. Thus, XTI can be used both to interface mOSI to a network transport service and to provide mOSI services to networked applications. Figure 3 illustrates the use of the XTI interfaces for mOSI implemented as a single module.

---

8) Internet RFC 1698, "Octet Sequences for Upper-Layer OSI to support Basic Communications Applications": P. Furniss: London & IETF, October 1994.

9) XTI: "X/Open Transport Interface (XTI) version 2 X/Open CAE specification C410 (Appendix H, Minimum OSI Functionality)", X/Open Company Limited, Apex Plaza, Sorbury Road, Reading, Berkshire, RG1 1AX, UK, 1994.

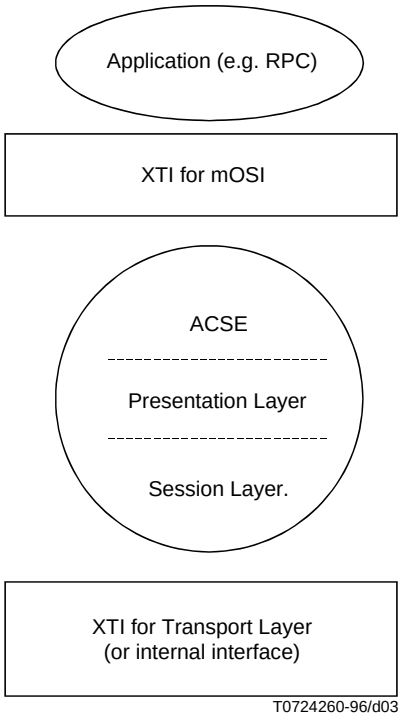


FIGURE 3/X.638

**XTI Interface for mOSI**

# Annex A

## 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 9.3).

This annex uses the tables in the ACSE PICS proforma (see ITU-T Rec. X.247 | 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 Recommendation.

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.

NOTE – PICS proforma clauses, A.1-A.4, are not covered by this Recommendation. The questions are answered by an implementor of ACSE.

## A.1 Global statement of conformance – [PICS proforma, A.5]

| | Question | Answer | PICS proforma reference |
|---|---|---|---|
| 1 | Are all mandatory features supported? | Yes | A.5/1 |

## A.2 Supported roles – [PICS proforma, 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*.<br>[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*.<br>[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, 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, 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] | A.8/3 | [A-FU(ACN)] |
| | [1] The value of *Authentication*.<br>[2] The value of *Application-context-negotiation*. | | | |

## A.5 Supported APDUs – [PICS proforma, 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 “–”.<br>[2] If [A-CON-responder] then “m” else “–”.<br>[3] If [A-REL-requestor] then “m” else “–”.<br>[4] If [A-REL-acceptor] then “m” else “–”. | | | | |

## A.6 Supporting APDU parameters – [PICS proforma, 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[1] | m[1] | 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 | * | *[5] | A.10.1/3 | If either is supported, then both shall be supported. |
| 4 | Calling AE Qualifier | * | *[5] | A.10.1/4 | |
| 5 | Calling AP Invocation-identifier | * | *[5] | A.10.1/5 | If either is supported, then both shall be supported. |
| 6 | Calling AE Invocation-identifier | * | *[5] | A.10.1/6 | |
| 7 | Called AP Title | * | *[5] | A.10.1/7 | If either is supported, then both shall be supported. |
| 8 | Called AE Qualifier | * | *[5] | A.10.1/8 | |
| 9 | Called AP Invocation-identifier | * | *[5] | A.10.1/9 | If either is supported, then both shall be supported. |
| 10 | Called AE Invocation-identifier | * | *[5] | A.10.1/10 | |
| 11 | ACSE Requirements | c[2] | c[2] | A.10.1/11 | |
| 12 | Authentication-mechanism Name | c[3] | c[3] | A.10.1/12 | |
| 13 | Authentication-value | c[3] | c[3] | A.10.1/13 | |
| 14 | Application Context Name List | c[4] | c[4] | A.10.1/14 | |
| 15 | Implementation Information | i | i | A.10.1/15 | |
| 16 | User Information | * | *[5] | A.10.1/16 | |

[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] Omission of this parameter, i.e. using the default value, is regarded as support for this parameter.

[2] If [A-FU(AU) or A-FU(ACN)] then “m” else “–”.

[3] If [A-FU(AU)] then “m” else “–”.

[4] If [A-FU(ACN)] then “m” else “–”.

[5] The status value for this item is not in accord with the current version of ITU-T Rec. X.247 | ISO/IEC 8650-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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[1] | m[1] | 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 | * | *[6] | A.10.2/3 | If either is supported, then both shall be supported. |
| 4 | Responding AE Qualifier | * | *[6] | A.10.2/4 | |
| 5 | Responding AP Invocation-identifier | * | *[6] | A.10.2/5 | If either is supported, then both shall be supported. |
| 6 | Responding AE Invocation-identifier | * | *[6] | A.10.2/6 | |
| 7 | Result | m | m | A.10.2/7 | |
| 8 | Result Source-diagnostic | m[5] | m[5] | A.10.2/8 | |
| 9 | ACSE Requirements | c[2] | c[2] | A.10.2/9 | |
| 10 | Authentication-mechanism Name | c[3] | c[3] | A.10.2/10 | |
| 11 | Authentication-value | c[3] | c[3] | A.10.2/11 | |
| 12 | Application Context Name List | c[4] | c[4] | A.10.2/12 | |
| 13 | Implementation Information | i | i | A.10.2/13 | |
| 14 | User Information | * | *[6] | A.10.2/14 | |

[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] Omission of this parameter, i.e. using the default value, is regarded as support for this parameter.

[2] If [A-FU(AU) or A-FU(ACN)] then “m” else “–”.

[3] If [A-FU(AU)] then “m” else “–”.

[4] If [A-FU(ACN)] then “m” else “–”.

[5] If [A-FU(AU)] then “m” with a value range of 0-14; otherwise “m” with a value range of 0-10.

[6] The status value for this item is not in accord with the current version of ITU-T Rec. X.247 | ISO/IEC 8650-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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 | * | *[1] | A.10.3/1 | |
| 2 | User Information | * | *[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.247 | ISO/IEC 8650-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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 | * | *[1] | A.10.4/1 | |
| 2 | User Information | * | *[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.247 | ISO/IEC 8650-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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 | |
| 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, 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.11.2/1 | 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.11.2/2 | |
| 3 | EXTERNAL | c[1] | c[2] | A.11.2/3 | |
| 4 | Other | c[1] | c[2] | A.11.2/4 | |

[1] If [A-FU(AU)] then "*" else "–".

[2] If [A-FU(AU)] then "m" else "–".

# Annex B

## mOSI requirements for Presentation Layer facilities

This annex contains the mOSI specifications for completing the Presentation Layer Profile Requirements List (PRL) for the selected features, roles and options (see 9.3).

This annex uses the tables in the Presentation Layer PICS proforma (see ITU-T Rec. X.246 | 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 Recommendation.

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.

NOTE – PICS proforma clauses A.1-A.4 are not covered in this Recommendation. The questions are answered by an implementor of the Pres. service.

### B.1 Global statement of conformance – [PICS proforma, 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, 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, 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*.<br>[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*.<br>[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 | [P-REL-requestor] |
| 2 | Acceptor | c[2] | A.7.1.1.3/2 | [P-REL-acceptor] |
| | [1] The value of *Release-requestor*.<br>[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 “–”.<br>[2] If [A-CON-responder] then “m” else “–”.<br>[3] The value of *Establishment-responder-reject.*<br>[4] If [P-DATA-requestor] then “m” else “–”.<br>[5] If [P-DATA-acceptor] then “m” else “–”. | | | | |

NOTE – The remainder of the subclauses in A.7 are out of the scope (i) of this Recommendation.

## B.4 Supported PPDU parameters – [PICS proforma, 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[1] | 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 | i | i[1] | A.8.1/5 | May be used for simple encoding |
| 6 | Protocol version | m[2] | m[2] | A.8.1/6 | If sent, the value shall be “version 1”; the default value is “version 1”. |
| 7 | Presentation requirements | i | i[1] | 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[1] | 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.<br><br>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 | * | *[1] | A.8.1/10 | |

[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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.246 | ISO/IEC 8823-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

[2] Omission of this parameter, i.e. using the default value, is regarded as support for this parameter.

NOTE – The X.410 (1984) parameters are out of the scope (i) of this Recommendation.

### 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[2] | m[2] | 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[1] | A.8.2/5 | Not used by BCA |
| 6 | User session requirements | i | i[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.246 | ISO/IEC 8823-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

[2] Omission of this parameter, i.e. using the default value, is regarded as support for this parameter.

NOTE – The X.410 (1984) parameters are out of the scope (i) of this Recommendation.

### 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[2] | m[2] | A.8.3/3 | |
| 4 | Default context result | i | i[1] | A.8.3/4 | |
| 5 | Provider reason | o[1] | o[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.246 | ISO/IEC 8823-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

[2] Omission of this parameter, i.e. using the default value, is regarded as support for this parameter.

NOTE – The X.410 (1984) parameters are out of the scope (i) of this Recommendation.

### 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 |

NOTE – The X.410 (1984) parameters are out of the scope (i) of this Recommendation.

### 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 | o[1] | o[1] | A.8.5/1 | |
| 2 | Event identifier | * | *[1] | A.8.5/2 | |
| | [1] The status value for this item is not in accord with the current version of ITU-T Rec. X.246 \| ISO/IEC 8823-2 due to a known defect. A defect report has been raised on this item to resolve this difference. | | | | |

NOTE – PICS proforma, subclauses A.8.6 and A.8.7 are out of the scope (i) of this Recommendation.

### 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] If [P-DATA-requestor] then “m” else “–”.<br>[2] If [P-DATA-acceptor] then “m” else “–”. | | | | |

NOTE – PICS proforma, subclauses A.8.9 through A.8.14 are out of the scope (i) of this Recommendation.

### 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.8.15/2 | |
| 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 [P-REL-requestor] then “m” else “–”.<br>[2] If [P-REL-acceptor] then “m” else “–”. | | | | |

## B.5 Support of syntaxes – [PICS proforma, 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-itu-t asn1(1) basic-encoding(1)} | m | Rec. X.690 | Rec. X.637, clause 8 |
| 2 | Object identifier | {joint-iso-itu-t standard(0) X.637(11188-1) mosi(3) default-transfer-syntax(2) version(1)} | * | F.2 | None |

NOTE – 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-itu-t association-control(2) abstract-syntax(1) apdus(0) version1(1)} | m |
| 2 | Object identifier | {joint-iso-itu-t standard(0) X.637(11188-1) mosi(3) default-abstract-syntax(1) version(1)} (Annex F) | * |

NOTE – 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 Rec. X.637, 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 Rec. X.637, 8.1.2 |
| 4 | Only primitive encoding used for OCTET STRING | * | See Rec. X.637, 8.1.7 (Note 2) |
| 5 | Only primitive encoding used for BITSTRING | * | See Rec. X.637, 8.1.7 (Note 2) |
| | NOTE 1 – A referencing implementation shall receive all forms of encoding.<br>NOTE 2 – Rec. X.637 restricts encoding of an OCTET STRING or a BIT STRING to only one level of constructed encoding. | | |

### B.5.4 PDV structure of User Data parameters

| | Restriction | Profile | Constraint / [mnemonic] |
|---|---|---|---|
| 1 | Limit on number of PDVs in User Data parameter | * | |
| 2 | Limit on number of PDVs in a single PDV-list value | * | |

# Annex C

## 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 9.3).

This annex uses the tables in the Session Layer PICS proforma (see ITU-T Rec. X.245 | 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 Recommendation.

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.

NOTE – PICS proforma clauses A.1-A.4 are not covered by this Recommendation. 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, A.4.2]

| | Version | Profile | PICS proforma reference | Constraint / [mnemonic] |
|---|---|---|---|---|
| 1 | Version 1 | i | | |
| 2 | Version 2 | m | A.4.2/2 | |

NOTE – PICS proforma, A.4.3 is not covered in this Recommendation. The question is answered by an implementor of the Session service.

## C.2 Global statement of conformance – [PICS proforma, 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, 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 | Re-use 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 > 10 240 (S-CONNECT) | i | A.6.2/9 | Allows approximately 10 K 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, 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*.<br>[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*.<br>[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*.<br>[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 10 K 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*.<br>[2] The value of *Establishment-responder*.<br>[3] The value of *Establishment-responder-reject*.<br>[4] The value of *Release-requestor*.<br>[5] The value of *Release-acceptor*.<br>[6] The value of *Normal-data-requestor*.<br>[7] The value of *Normal-data-acceptor*.<br>[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][3] | c[2][3] | A.7.1.3/1 | 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[3] | i[3] | A.7.1.3/2 | This Profile recommends only sending GT as the null category 0 SPDU. |

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

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

[3] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

NOTE – The remainder of the subclauses in A.7 are out of the scope (i) of this Recommendation.

## C.5 Supported SPDU parameters – [PICS proforma, 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[1] | A.8.1.1/1 | Not used by BCA |
| 2 | Common Reference | i | i[1] | A.8.1.1/2 | Not used by BCA |
| 3 | Additional Reference Information | i | i[1] | A.8.1.1/3 | Not used by BCA |

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### C.5.1.2 Connect/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 | i | 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.5.1.3 Single 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[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

## 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[1] | A.8.4.1/1 | Not used by BCA |
| 2 | Common Reference | i | i[1] | A.8.4.1/2 | Not used by BCA |
| 3 | Additional Reference Information | i | i[1] | A.8.4.1/3 | Not used by BCA |

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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 | For BCA, extended concatenation is out of scope. |
| 2 | TSDU maximum size | i | i | A.8.4.2.1/2 | |
| 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.5.2.3 Single 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[1] | 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[1] | A.8.4.3/3 | |
| 4 | Calling Session Selector | m[2] | o | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

[2] Omission of this parameter, i.e. using the default value, is regarded as support for this parameter.

### 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[1] | A.8.5.1/1 | Not used by BCA |
| 2 | Common Reference | i | i[1] | A.8.5.1/2 | Not used by BCA |
| 3 | Additional Reference Information | i | i[1] | A.8.5.1/3 | Not used by BCA |

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

#### 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[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

NOTE – PICS proforma, A.8.8 is out of scope (i) of this Recommendation.

### 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 | i | i | A.8.9/1 | |
| 2 | Enclosure Item | i | i[1] | A.8.9/2 | |
| 3 | Reflect Parameter Values | o | o[1] | 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. |

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

### 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[1] | 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.

[1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 | ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference.

NOTE – Subclauses A.8.12 through A.8.15 are out of the scope (i) of this Recommendation.

### 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[1] | A.8.16/2 | |
| 3 | User Data | i | i[1] | A.8.16/3 | |
| | [1] The status value for this item is not in accord with the current version of ITU-T Rec. X.245 \| ISO/IEC 8327-2 due to a known defect. A defect report has been raised on this item to resolve this difference. | | | | |

NOTE – The remainder of the clauses in the Session PICS proforma are out of the scope (i) of this Recommendation.

# Annex D

# Requirements compliance statement proforma

## D.1 Requirements questionnaire

This annex may be used by a profile or the specification of a basic communications application[10)] to claim that its upper layer requirements comply to this Recommendation. Such a claim indicates that upper layer requirements of the referencing specification are exactly identified by some or all of the features specified in this Recommendation. The requirements questionnaire (see Table D.1) is the basis for the mOSI compliance statement. It is intended to be completed by the designers of the referencing specification.

## D.2 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 9.3). 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 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 Recommendation as indicated by Tables D.1 and D.2 plus any supplemental information mentioned in D.3.

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

[10)] For the purposes of this annex, the term “referencing specification” will refer to a “profile or the specification of a basic communication application”.

all “i” (and “–” for non-supported roles), Table D.2 need not be completed. In this case, line 13 of Table D.1 would have the answer “m”, or “o”, or “i”, respectively.

## D.3 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/X.638

**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 shall not be “i” |
| 5 | *Normal-data-acceptor* | m; o; i | | |
| 6 | *Release-requestor* | m; o; 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 | 2 or more | | One of these shall be the presentation-context used for ACSE-PDUs. |
| 12 | ITU-T Rec. X.637 and ISO/IEC ISP 11188-1 compliance?[11)] | Yes | | If the answer is not “yes”, the referencing specification may not claim mOSI compliance. |
| 13 | Status values for all open (*) parameters (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. |
| 14 | Maximum number of PDV’s required | 1 or more | | |

[11)] See clause 2 and ITU-T Rec. X.637 and ISO/IEC ISP 11188-1, Annex B.

TABLE D.2/X.638

**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 qualifier for each. |
| 2 | [AARQ] | Called AE title | | | |
| 3 | | Calling invocation ids | | | Includes both the AP invocation identifier and the AE invocation identifier for each. |
| 4 | | Called invocation ids | | | |
| 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 | Abort Source | | | |
| 14 | [ABRT] | User Information | | | |
| 15 | A.7.1 [AARQ and | Form 1 (Directory name) | | | For Receiver, compliant answer is "m" or "–", that is, if AE titles are supported for receiving, both forms are mandatory. |
| 16 | AARE] | Form 2 (Object id + integer) | | | |
| 17 | B.4.1 [CP] | CPC Type | | | |
| 18 | B.4.5 [ARP] | Event Identifier | | | |
| 19 | B.5.1 Object Identifier | {joint-iso-itu-t standard(0) X.637 (11188-1) mosi(3) default-transfer-syntax(2) version(1)} | | | |
| 20 | B.5.2 Object Identifier | {joint-iso-itu-t standard(0) X.637 (11188-1) mosi(3) default-abstract-syntax(1) version(1)} | | | |

TABLE D.2/X.638 *(concluded)*

**Open parameters (*)**

| | Referenced table (in Annexes A, B and C) | Parameter | Specification's statement: Sender [a] | Specification's statement: Receiver [a] | Constraint / value |
|---|---|---|---|---|---|
| 21 | B.5.3<br><br>ASN.1 encoding forms | Only definite encoding | | | |
| | | Indefinite encoding for all constructed types | | | |
| | | Only minimal number of octets for definite form length encoding | | | |
| | | Only primitive form encoding for OCTET STRING | | | |
| | | Only primitive form encoding for BIT STRING | | | |
| 22 | B.5.4 | Limit on number of PDV's in User Data | | | |
| | | Limit on number of PDV's in a single PDV-list value | | | |
| | [a] Compliant answer for each row is "m", "o", "i", "x" or "–", unless indicated otherwise. | | | | |

### D.3.1 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.3.2 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.3.3 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.4 A referencing specification's upper layer PRL

As discussed in 9.3, this Recommendation 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 Recommendation.

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 choice" column.

# Annex E

# Implementation conformance statement proforma

## E.1 Implementation questionnaire

This annex may be used by an implementation to claim that it supports some or all of the features specified in this Recommendation. The implementation may in fact support more of the upper layer facilities of this Recommendation – without violating any of the facilities of this Recommendation.

The implementation questionnaire (see Table E.1) is the basis for the mOSI implementation conformance statement. It is intended to be completed by the designers of the referencing implementation.

## E.2 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 9.3). 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.

An implementation can answer yes to a particular entry in Table E.1 or E.2 only if it supports the corresponding items defined in the tables in Annexes A, B and C. An answer of "yes" to an entry in E.1 or E.2 means the implementor would fill in the tables of Annexes A, B and C as if the associated mOSI variable had the value "m" (and the implementation was conformant).

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 summarize the upper layer facilities it supports. 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 Recommendation. The implementation may in fact support more of the upper layer facilities of this Recommendation – without violating any of its features.

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), Table E.2 need not be completed. In this case, line 11 of Table E.1 would have the answer “yes” or “no”, respectively.

TABLE E.1/X.638

**Implementation support questionnaire**

| | Item / variable | Conformant answer | Implementation’s answer | Constraint / value |
|---|---|---|---|---|
| 1 | *Establishment-initiator* | Yes; no | | Both shall not be “no”.<br>If “yes”, the associated variable is implicitly assigned the value of “m”; otherwise the variable is “i”. |
| 2 | *Establishment-responder* | Yes; no | | |
| 3 | *Establishment-responder-reject* | Yes; no; – | | Answer shall be “–” if *Establishment-responder* is “no”.<br>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”.<br>If “yes”, the associated variable is implicitly assigned the value of “m”; otherwise the value is “i”. |
| 5 | *Normal-data-acceptor* | Yes; no | | |
| 6 | *Release-requestor* | Yes; no | | Both may be “no”.<br>If “yes”, the associated variable is implicitly assigned the value of “m”; otherwise the value is “i”. |
| 7 | *Release-acceptor* | Yes; no | | |
| 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 | 2 or more | | One of these shall be the presentation-context used for ACSE-PDUs. |
| 12 | ITU-T Rec. X.637 and ISO/IEC ISP 11188-1 conformance?[12)] | 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. |
| 14 | Maximum number of PDV’s supported | 1 or more | | |

[12)] See clause 2 and ITU-T Rec. X.637 and ISO/IEC ISP 11188-1, Annex B.

TABLE E.2/X.638

**Open parameters**

| | Referenced table (in Annexes A, B and C) | Parameter | Implementation's statement: Sender [a] | Implementation's statement: Receiver [a] | Constraint / value |
|---|---|---|---|---|---|
| 1 | A.6.1 | Calling AE title | | | Includes both AP title and AE qualifier for each |
| 2 | [AARQ] | Called AE title | | | |
| 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 qualifier |
| 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 | Abort Source | | | |
| 14 | [ABRT] | User Information | | | |
| 15 | A.7.1 [AARQ and | Form 1 (Directory name) | | | For Receiver, conformant answer is "yes" or "–" |
| 16 | AARE] | Form 2 (Object id + integer) | | | |
| 17 | B.4.1 [CP] | CPC Type | | | |
| 18 | B.4.5 [ARP] | Event Identifier | | | |
| 19 | B.5.1 Object identifier | {joint-iso-itu-t standard(0) X.637 (11188-1) mosi(3) default-transfer-syntax(2) version(1)} | | | |
| 20 | B.5.2 Object identifier | {joint-iso-itu-t standard(0) X.637 (11188-1) mosi(3) default-abstract-syntax(1) version(1)} | | | |
| 21 | B.5.3 ASN.1 encoding forms | Only definite encoding | | | |
| | | Indefinite encoding for all constructed types | | | |
| | | Only minimal number of octets for definite form length encoding | | | |
| | | Only primitive form encoding for OCTET STRING | | | |
| | | Only primitive form encoding for BIT STRING | | | |
| | B.5.4 | Limit on number of PDV's in User Data | | | |
| 22 | PDV structure | Limit on number of PDV's in a single PDV-list value | | | |
| | [a] Conformant answer for each row is "yes", "no" or "–". | | | | |

TABLE E.3/X.638

**Presentation and Session selectors**

| | Referenced table (in Annexes A, B and C) | Parameter | Conformant support answer: Sender | Conformant support answer: Receiver | Implementation's statement: Sender | Implementation's statement: Receiver |
|---|---|---|---|---|---|---|
| 1 | B.4.1/1 [CP] | Calling presentation selector | Yes; no; – | Yes; – | | |
| 2 | | Called presentation selector | Yes; – | Yes; no; – | | |
| 3 | B.4.2 [CPA] | Responding presentation selector | Yes; no; – | Yes; – | | |
| 4 | B.4.3 [CPR] | Responding presentation selector | Yes; no; – | Yes; – | | |
| 5 | C.5.1.3 [CN] | Calling session selector | Yes; no; – | Yes; – | | |
| 6 | | Called session selector | Yes; – | Yes; no; – | | |
| 7 | C.5.2.3 [AC] | Calling session selector | Yes; no; – | Yes | | |
| 8 | | Responding session selector | Yes; no; – | Yes; – | | |

## E.3 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.3.1 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.3.2 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.3.3 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.3.4 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 lists 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

# 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 Recommendation.

## 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 (PDV's). 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-itu-t standard(0) X.637 (11188-1) mosi(3) default-abstract-syntax(1) version(1)}**

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

NOTE 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 does not 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-itu-t standard(0) X.637 (11188-1) 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-itu-t standard(0) X.637 (11188-1) mosi(3) default-application-context(3)}**

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

## ITU-T RECOMMENDATIONS SERIES

| | |
|---|---|
| Series A | Organization of the work of the ITU-T |
| Series B | Means of expression: definitions, symbols, classification |
| Series C | General telecommunication statistics |
| Series D | General tariff principles |
| Series E | Overall network operation, telephone service, service operation and human factors |
| Series F | Non-telephone telecommunication services |
| Series G | Transmission systems and media, digital systems and networks |
| Series H | Audiovisual and multimedia systems |
| Series I | Integrated services digital network |
| Series J | Transmission of television, sound programme and other multimedia signals |
| Series K | Protection against interference |
| Series L | Construction, installation and protection of cables and other elements of outside plant |
| Series M | Maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits |
| Series N | Maintenance: international sound programme and television transmission circuits |
| Series O | Specifications of measuring equipment |
| Series P | Telephone transmission quality, telephone installations, local line networks |
| Series Q | Switching and signalling |
| Series R | Telegraph transmission |
| Series S | Telegraph services terminal equipment |
| Series T | Terminals for telematic services |
| Series U | Telegraph switching |
| Series V | Data communication over the telephone network |
| **Series X** | **Data networks and open system communication** |
| Series Z | Programming languages |