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

Output from the December 1993 Open Systems Environment Implementors' Workshop (OIW)

SIG Chair: **Neil Koorland (Microsoft)**
SIG Editor: **Rich Ankney (Fischer International)**

Foreword

This part of the Stable Implementation Agreements was prepared by the Message Handling Systems Special Interest Group (X.400 SIG) of the Open Systems Environment Implementors' Workshop (OIW).

Text in this part has been approved by the Plenary of the X.400 SIG. This part replaces the previously existing chapter on this subject. There is no significant technical change from this text as previously given.

Future changes and additions to this version of these Implementor Agreements will be published as change pages. Deleted and replaced text will be shown as strikeout. New and replacement text will be shown as shaded.

Table of Contents

List of Figures

List of Tables

Part 7 CCITT 1984 X.400 Based Message Handling System

NOTE - The classification schema used in this chapter (see table 7) pre-dated TR 10 000 and was the basis of extensive harmonization, as such: No attempt will be made to align this chapter with TR 10 000.

0 Introduction

This is an implementation agreement developed by the Implementor's Workshop sponsored by the National Institute of Standards and Technology to promote the useful exchange of data between devices manufactured by different vendors. This agreement is based on, and employs protocols developed in accord with, the OSI Reference Model. While this agreement introduces no new protocols, it eliminates ambiguities in interpretations.

This is an implementation agreement for a Message Handling System (MHS) based on the X.400-series of Recommendations (1984) and Version 5 of the X.400 Series Implementor's Guide from the CCITT. It is recommended that product vendors consult later versions of this guide. Figure 1 displays the layered structure of this agreement.

This agreement can be used over any Transport protocol class. In particular, this MHS agreement can be used over the Transport protocol class 0 used over CCITT X.25, described in clause 5.2 of this document. In addition, this MHS agreement can be used over the Transport profiles used in support of MAP (Manufacturing Automation Protocol) or TOP (Technical and Office Protocols). Note that the MAP or TOP environment must support the reduced Basic Activity Subset (BAS) as defined in X.410.

The UAs and MTAs require access to **directory** and **routing** services. A Directory Service is to be provided for each (vendor-specific) domain. Except insofar as they must be capable of providing addressing and routing described hereunder, these services and associated protocols are not described by this agreement.

User Agent Layer	CCITT X.420
Message Transfer Agent Layer	CCITT X.411
Reliable Transfer Service Layer	CCITT X.410
Presentation Layer	CCITT X.410 Sec. 4.2
Session Layer	See clause 5.9

Figure 1 - The layered structure of this implementation agreement.

1 Scope

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

a) PRMD to PRMD,

b) PRMD to ADMD,

c) ADMD to ADMD, and

d) MTA to MTA (within a PRMD, e.g., for MTAs from different vendors).

In case A, the PRMDs do not make use of MHS services provided by an ADMD. In cases B and C, UAs associated with an ADMD can be the source or destination for messages. Furthermore, in cases A and B, a PRMD can serve as a relay between MDs, and in cases B and C an ADMD can serve as a relay between MDs. Figure 2 illustrates the interfaces to which the agreement applies.

X.400 protocols other than the Message Transfer Protocol (P1) and the Interpersonal Messaging Protocol (P2) are beyond the scope of this agreement. Issues arising from the use of other protocols or relating to P1 components in support of other protocols are outside the scope of this document. This agreement describes the minimum level of services provided at each interface shown in figure 2. Provision for the use of the remaining services defined in the X.400 Series of Recommendations is outside the scope of this document.

With the exception of intra domain connections, this agreement does not cover message exchange between communicating entities within a domain even if these entities communicate via P1 or P2. Bilateral agreements between domains may be implemented in addition to the requirements stated in this document. **Conformance to this agreement requires the ability to exchange messages without use of bilateral agreements.**

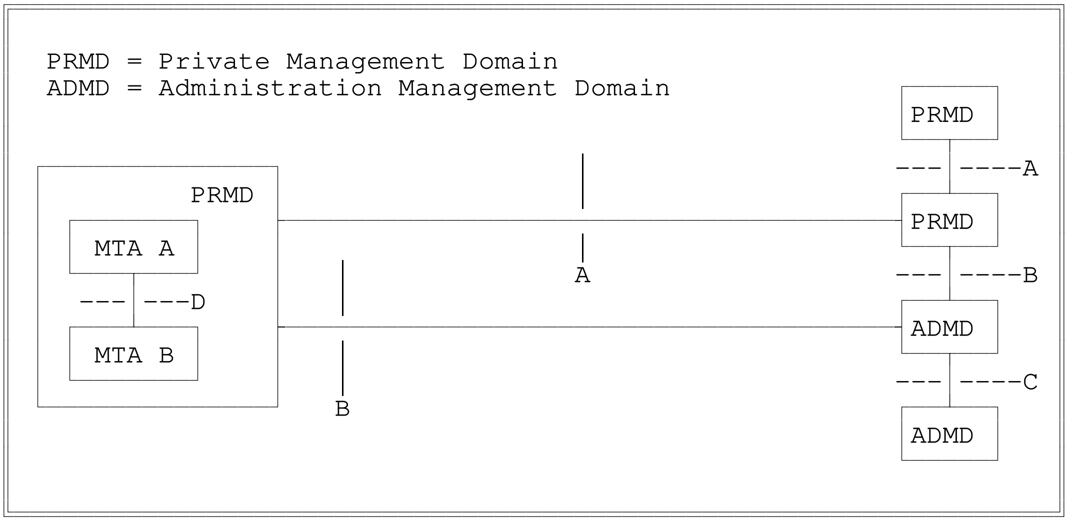


Figure 2 - This agreement applies to the interface between: (A) PRMD and PRMD; (B) PRMD and ADMD; (C) ADMD and ADMD; and (D) MTA and MTA.

2 Normative references

CCITT Recommendation X.121 (1988), International Numbering Plan.

CCITT Recommendation X.400, (Red Book, 1984), Message Handling Systems: System Model-Service Elements.

CCITT Recommendation X.401, (Red Book, 1984), Message Handling Systems: Basic Service Elements and Optional User Facilities.

CCITT Recommendation X.408, (Red Book, 1984), Message Handling Systems: Encoded Information Type Conversion Rules.

CCITT Recommendation X.409, (Red Book, 1984), Message Handling Systems: Presentation Transfer Syntax and Notation.

CCITT Recommendation X.410, (Red Book, 1984), Message Handling Systems: Remote Operations and Reliable Transfer Server.

CCITT Recommendation X.411, (Red Book, 1984), Message Handling Systems: Message Transfer Layer.

CCITT Recommendation X.420, (Red Book, 1984), MessageHandling Systems: Interpersonal Messaging User Agent Layer.

CCITT Recommendation X.430, (Red Book, 1984), Message Handling Systems: Access Protocol for Teletex Terminals.

3 Status

This version of the X.400 based Message Handling System implementation agreements was completed on December 16, 1988. No further enhancements will be made to this version. See the next clause--Errata.

4 Errata

5 PRMD to PRMD

5.1 Introduction

This clause is limited in scope to issues arising from the **direct** connection (interface A in figure 2) of two PRMDs. "Direct" means that no ADMD or relaying PRMD provides MHS services to facilitate message interchange. "Direct" does not exclude those instances for which Administrations happen to be ADMDs but are not providing X.400 services, that is, they are used only to provide lower layer services such as X.25. Figure 3 schematically represents the scope of this clause.

These issues relate to the use of the UAL (User Agent Layer) and MTL (Message Transfer Layer) services, protocol elements, recommended practices and constraints. In particular, this clause addresses the P1 and P2 protocols and their related services in a direct connection environment. This clause describes the minimum level of services provided by a PRMD. Provision for the use of the remaining services defined in the X.400 Series of Recommendations is beyond the scope of this clause.

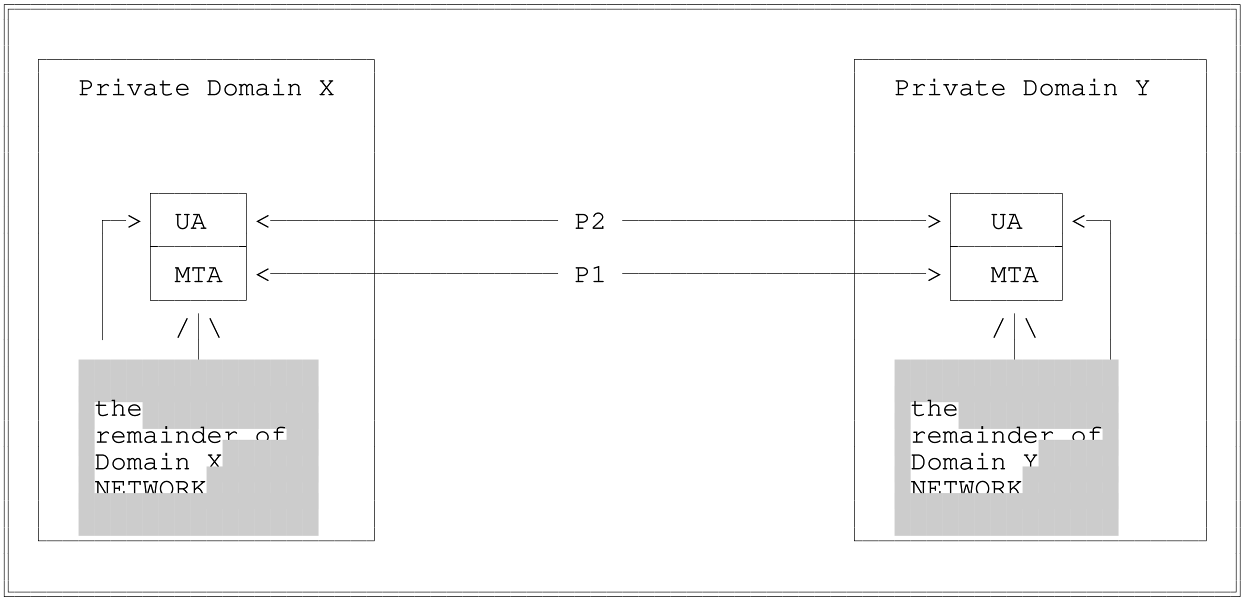


Figure 3 - Interconnection of private domains.

5.2 Service elements and optional user facilities

This clause identifies those service elements and optional user facilities that must be provided in support of P1 and P2.

5.2.1 Classification of support for services

The classification of UA and MT-Service elements is used to define characteristics of equipment. Equipment can claim SUPPORT or NON-SUPPORT of a Service; in the case of UA-service elements, a separate classification is given for Origination and Reception.

The service provider is defined as the entity providing the service, in this case, the MTL or the UAL. The service user is either the MHS user or the UAL. The classification of provider and user relates to the sublayer for which the service element is defined.

5.2.1.1 Support (S)

Support means:

a) The service provider makes the service element available to the service user, and

b) The service user gives adequate support to the MHS to invoke the service element or makes information associated with the service element available.

Support for Origination means that:

a) The service provider makes the service element available to the service user for invocation, and

b) The service user gives adequate support to the end user of the MHS to invoke the service element.

Support for Reception means that:

a) The service provider makes information associated with the service element available to the service user.

NOTE - A UA- or MT-service element can carry information from originator to recipient only if:

b) the service element is available to the originator,

c) the service element is available to the recipient, and

d) all intermediate steps carry the information.

5.2.1.2 Non Support (N)

This means that the service provider is not required to make the service element available to the service user. However, the service provider should not regard the occurrence of the corresponding protocol elements as an error and should be able to relay such elements. Implementations making a profile available should indicate deviations (additions or deletions) with respect to the requirement in the profile.

5.2.1.3 Not Used (N/U)

This means that although the Recommendation allows this service element, this profile does not use it.

5.2.1.4 Not Applicable (N/A)

This means that this service element does not apply in this particular case (for originator or recipient).

5.2.2 Summary of supported services

Within a PRMD, a User Agent must support **all** P2 BASIC IPM Services (X.400) and **all** P2 ESSENTIAL IPM Optional user facilities (X.401) subject to the qualifiers listed in annex A.

Within a PRMD, a MTA must support **all** BASIC MT Services (X.400) and **all** ESSENTIAL MT optional user facilities (X.401) subject to the qualifiers listed in annex A.

No support is required of the additional optional user facilities of X.401.

5.2.3 MT service elements and optional user facilities

Tables 1 through 3 show the Message Transfer (MT) service elements and optional user facilities.

Table 1 - Basic MT service elements

Service Elements	Support (S) or Non-support (N)
Access Management	N/U[1]
Content Type Indication	S
Converted Indication	S
Delivery Time Stamp Indication	S
Message Identification	S
Non-delivery Notification	S
Original Encoded Information Types Indication	S
Registered Encoded Information Types	N/U[1]
Submission Time Stamp Indication	S

Notes
1 Not applicable to co-resident UA and MTA.

Table 2 - MT optional user facilities provided to the UA-selectable on a per-message basis

MT Optional User Facilities	Categorization	Support (S) or Non-support (N)
Alternate Recipient Allowed	E	S
Conversion Prohibition	E	S
Deferred Delivery	E	N[2]
Deferred Delivery Cancellation	E	N[2]
Delivery Notification	E	S
Disclosure of Other Recipients	E	N[3]
Explicit Conversion	A	N
Grade of Delivery Selection	E	S
Multi-destination Delivery	E	S
Prevention of Non-delivery Notification	A	N
Probe	E	N[4]
Return of Contents	A	N

Legend
E: Essential optional user facility.
A: Additional optional user facility.

Notes
2 A local facility subject to qualifiers in annex A.
3 Support not required for an originating MT user; support must be provided for recipient MT users.
4 Subject to qualifiers in annex A.

Table 3 - MT optional user facilities provided to the UA agreed for any contractual period of Time

MT Optional User Facilities	Categorization	Support (S) or Non-Support (N)
Alternate Recipient Assignment	A	N
Hold for Delivery	A	N/U
Implicit Conversion	A	N

Legend
A: Additional optional user facility.

5.2.4 IPM service elements and optional user facilities

Tables 4 through 6 show the IPM service elements and optional user facilities.

Table 4 - Basic IPM service elements

Service Elements	Origination by UAs	Reception by UAs
Access Management	N/U[5]	N/U[5]
Content Type Indication	S	S
Converted Indication	N/A	S
Delivery Time Stamp Indication	N/A	S
Message Identification	S	S
Non-delivery Notification	S	N/A
Original Encoded Information Types Indication	S	S
Registered Encoded Information Types	N/A	N/A[5]
Submission Time Stamp Indication	S	S
IP-message Identification	S	S
Typed Body	S	S

Notes
5 Does not apply to co-resident UA and MTA.

Table 5 - IPM optional facilities agreed for a contractual period of time

Service Elements	Categorization	Support (S) or Non-Support (N)
Alternate Recipient Assignment	A	N
Hold for Delivery	A	N
Implicit Conversion	A	N

Table 6 - IPM optional user facilities selectable on a per-message basis

IPM Optional User Facilities	Origination by UAs	Reception by UAs
Alternate Recipient Allowed	A (N)	A (N)
Authorizing Users Indication	A (N)	E (S)
Auto-forwarded Indication	A (N)	E (S)
Blind Copy Recipient Indication	A (N)	E (S)
Body Part Encryption Indication	A (N)	E (S)
Conversion Prohibition	E (S)	E (S)
Cross-referencing Indication	A (N)	E (S)
Deferred Delivery	E (N)[6]	N/A
Deferred Delivery Cancellation	A (N/U)[6]	N/A
Delivery Notification	E (S)	N/A
Disclosure of Other Recipients	A (N)	E (S)
Expiry Date Indication	A (N)	E (S)
Explicit Conversion	A (N)	N/A
Forwarded IP-message Indication	A (N)	E (S)
Grade of Delivery Selection	E (S)	E (S)
Importance Indication	A (N)	E (S)
Multi-destination Delivery	E (S)	N/A
Multi-part Body	A (N)	E (S)
Non-receipt Notification	A (N)	A (N)
Obsoleting Indication	A (N)	E (S)
Originator Indication	E (S)	E (S)
Prevention of Non-delivery Notification	A (N)	N/A
Primary and Copy Recipients Indication	E (S)	E (S)
Probe	A (N)	N/A
Receipt Notification	A (N)	A (N)
Reply Request Indication	A (N)	E (S)
Replying IP-message Indication	E (S)	E (S)
Return of Contents	A (N)	N/A
Sensitivity Indication	A (N)	E (S)
Subject Indication	E (S)	E (S)

Notes
6 A local facility subject to qualifiers in annex A.

5.3 X.400 protocol definitions

This clause reflects the agreements of the OIW regarding P1 and P2 protocol elements.

5.3.1 Protocol classification

The protocol classifications are defined in table 7.

Table 7 - Protocol classifications

```
1)  UNSUPPORTED = X
    These elements may be generated, but no specific processing should
    be expected in a relaying or delivering domain. A relaying domain
    must at least relay the semantics of the element. The absence of
    these elements should not be assumed, in a relaying or delivering
    domain, to convey any significance.

2)  SUPPORTED = H
    These elements may be generated. However, implementations are not
    required to be able to generate these elements. Appropriate
    actions shall be taken in a relaying or delivering domain.

3)  GENERATABLE = G
    Implementations must be able to generate and handle these protocol
    elements, although they are not necessarily present in all
    messages generated by implementations of this profile.
    Appropriate actions shall be taken in a relaying or delivering
    domain.

4)  REQUIRED = R
    Implementations of this profile must always generate this protocol
    element. However, its absence cannot be regarded as a protocol
    violation as other MHS implementations may not require this
    protocol element. Appropriate actions shall be taken in a
    relaying or delivering domain.

5)  MANDATORY = M
    This must occur in each message as per X.411 or X.420 as
    appropriate; absence is a protocol violation. Appropriate actions
    shall be taken in a relaying or delivering domain.
```

5.3.2 General statements on pragmatic constraints

Where a protocol element is defined as a choice of Numeric String and Printable String (i.e., Administration Domain Name and Private Domain Identifier), then a numeric value encoded as a printable string is equivalent to the same value encoded as a numeric string. This does not apply to the Country Name protocol element.

The maximum number of recipients in a single MPDU is 32K - 1 (that is, 32767). However, no individual limits on the number of occurrences (recipients) are placed on the following protocol elements: Authorizing Users, Primary Recipients, Copy Recipients, Blind Copy Recipients, Obsoletes and Cross References. Additionally, there is no limit on the number of Reply to Users. This is a local matter for the originating system.

Use of strings. A Printable String is defined in terms of the number of characters, which is the same number of octets. For T.61 strings the number of octets is twice the number of characters specified.

The ability to generate maximum size elements is not required, with the exception of the component fields in the Standard Attribute List, in which case it is required.

5.3.3 MPDU size

The following agreements govern the size of MPDUs:

a) All MTAEs must support at least one MPDU of at least 2 megabyte, and

b) The size of the largest MPDU supported by a UAE is a local matter.

5.3.4 P1 protocol elements

Table 8 contains Protocol Elements and their classes.

Table 8 - P1 protocol elements

Element	Class	Restrictions and Comments
MPDU		
UserMPDU	G	
DeliveryReportMPDU	G	
ProbeMPDU	H	
UserMDPU		
UMPDUEnvelope	M	
UMPDUContent	M	
UMPDUEnvelope		
MPDUIdentifier	M	
originator ORname	M	
originalEncodedInformationTypes	G	If this field is absent, then the Encoded Information Type is "unspecified."
ContentType	M	
UAContentID	H	Maximum length = 16 characters.
Priority	G	
PerMessageFlag	G	Maximum length = 2 octets.
deferredDelivery	X	
PerDomainBilateralInfo	X	No limit on number of occurrences.
RecipientInfo	M	Maximum number = 32K - 1 occurrences. More severe limitations are by bilateral agreement.
TraceInformation	M	
UMPDUContent	M	
MPDUIdentifier		
GlobalDomainIdentifier	M	
IA5String	M	Maximum length = 32 characters, graphical subset only. Refer to T.50 for clarification of graphical subset.
PerMessageFlag		
discloseRecipients	H	
conversionProhibited	G	
alternateRecipientAllowed	H	
contentReturnRequest	X	

Table 8 - P1 protocol elements (continued)

Element	Class	Restrictions and Comments
PerDomainBilateralInfo		
CountryName	M	Maximum length = 3 characters.
AdministrationDomainName	M	Maximum length = 16 characters.
BilateralInfo	M	Maximum depth = 8; maximum length = 1024 octets (including encoding).
RecipientInfo		
recipient	M	
ExtensionIdentifier	M	Maximum value = 32K - 1 (32767).
perRecipientFlag	M	Maximum length = 2 octets.
ExplicitConversion	X	
perRecipientFlag		
ResponsibilityFlag	M	
ReportRequest	M	
UserReportRequest	M	
TraceInformation		Reference should be made to Version 5 of the X.400 Implementor's Guide for information related to Trace sequencing.
GlobalDomainIdentifier	M	
DomainSuppliedInfo	M	
DomainSuppliedInfo		
arrival	M	
deferred	X	
action	M	
0=relayed (value)	G	
1=rerouted (value)	H	Rerouting is not required.
converted	H	
previous	H	
ORName		See clause 5.3.5
EncodedInformationTypes		
bit string	M	Delivery can only occur if match is made with Registered Encoded Information Types. Individual vendors may impose limits. Maximum length = 4 octets.
G3NonBasicParameters	X	
TeletexNonBasicParameters	X	
PresentationCapabilities	X	

Table 8 - P1 protocol elements (continued)

Element	Class	Restrictions and Comments
DeliveryReportMPDU		
DeliveryReportEnvelope	M	
DeliveryReportContent	M	
DeliveryReportEnvelope		
report	M	
originator	M	
TraceInformation	M	
DeliveryReportContent		
original	M	
intermediate	G	See comment at end of table.
UAContentID	G	
ReportedRecipientInfo	M	Maximum number = 32K - 1 occurrences.
returned	H	Can only be issued if specifically requested in the originating message.
billingInformation	X	Maximum depth = 8; maximum length = 1024 octets (including encoding).
ReportedRecipientInfo		
recipient	M	
ExtensionsIdentifier	M	
PerRecipientFlag	M	
LastTraceInformation	M	
intendedRecipient	H	
SupplementaryInformation	X	Maximum length = 64 characters. Value is pending verification by the CCITT SG VIII or XI.
LastTraceInformation		
arrival	M	
converted	G	
Report	M	

Table 8 - P1 protocol elements (concluded)

Element	Class	Restrictions and Comments
Report		
DeliveredInfo	G	Generated if delivery is reported.
NondeliveredInfo	G	Generated if failure to deliver is reported.
DeliveredInfo		
delivery	M	
typeofUA	R	This element must be generated with a PRIVATE value by PRMDs.
NonDeliveredInfo		
ReasonCode	M	
DiagnosticCode	H	Whenever possible, use a meaningful diagnostic code.
ProbeEnvelope		
probe	M	
originator	M	
ContentType	M	
UAContentID	H	Maximum length = 16 characters.
original	G	If this field is absent, then the Encoded Information Type is "unspecified."
TraceInformation	M	
PerMessageFlag	G	
contentLength	H	
PerDomainBilateralInfo	X	
RecipientInfo	M	Maximum number = 32K - 1 occurrences.
GlobalDomainIdentifier		
CountryName	M	Maximum length = 3 characters.
AdministrationDomainName (4)	M	Maximum length = 16 characters or digits.
PrivateDomainIdentifier	R	Maximum length = 16 characters or digits. This element must be generated by PRMDs.
End of Definitions		

Note: **[**Comment on intermediate TraceInformation in the DeliveryReportContent in table 8: Audit and confirmed reports should not be requested by other than the originating domain for two reasons. First, the return path of the report may be different from the path taken by the original message, and it may exclude the domain that added the request for audit and confirmed to the message. Second, if the return path is different from the path of the original message, the originating domain would receive intermediate trace information that it did not request.**]**

5.3.5 ORName protocol elements

Only form 1 variant 1 O/R names are supported.

Table 9 contains ORName protocol elements.

These agreements interpret 1984 X.400 clause 3.4 to mean that the attributes in the ORName in the MPDU must identify exactly one UA, and that all the values for the attributes specified in the ORName must be identical to the values of the corresponding attributes associated with the recipient UA. Underspecified names that are unique are deliverable.

Overspecified ORNames, ORNames with mismatching fields, and ambiguous names are to be non-delivered or sent to the alternate recipient as appropriate.

Table 9 - ORName protocol elements

Element	Class	Restrictions and Comments
ORName		
StandardAttributeList	M	
DomainDefinedAttributeList	G	
StandardAttributeList (1)		
CountryName	R	As defined in X.411, Maximum length = 3 characters.
AdministrationDomainName (4)	R	Maximum length = 16 characters or digits.
X121Address	X	Maximum length = 15 digits.
TerminalID	X	Maximum length = 24 characters.
PrivateDomainName (2)	G	Maximum length = 16 characters.
OrganizationName (2)	G	Maximum length = 64 characters.
UniqueUAIdentifier	X	Maximum length = 32 digits.
PersonalName	G	Maximum length of values of sub-elements = 64 characters. Note: The possibility that this value may be reduced to 40 characters is for further study by the CCITT.
OrganizationalUnit (3)	G	Maximum length = 32 characters per occurrence. A maximum of four occurrences are allowed.
DomainDefinedAttributeList (5)		Maximum = 4 occurrences.
type	M	Maximum length = 8 characters.
value	M	Maximum length = 128 characters.
PersonalName		
surName	M	Maximum length = 40 characters.
givenName	G	Maximum length = 16 characters.
initials	G	Maximum length = 5 characters; excluding surname initial and punctuation and spaces.
generationQualifier	G	Maximum length = 3 characters.

Table 9 - ORName protocol elements (concluded)

Notes:

1 The following apply for comparison of the Standard Attributes of an O/R Name:
   a) Lower case is interpreted as upper case (for IA5).
   b) Multiple spaces may be interpreted as a single space. Originating domains shall only transmit single significant spaces. If multiple spaces are transmitted, non-delivery may occur.

2 At least one of these must be supplied.

3 These should be sent in descending sequence, from the most significant <Organizational Unit> (highest in organization hierarchy) to the least significant. Only those specified should be sent. (That is, an unspecified <Organizational Unit> should not be sent along as a field of [null] content, nor zero length, etc.)

4 This attribute shall contain one space in all ORNames of messages originated in a PRMD that is not connected to an ADMD, and in ORNames of recipients reachable only through a PRMD; otherwise, this attribute shall contain an appropriate ADMD name.

5 Some existing systems which will be accessed via an X.400 service (whether directly connected using X.400 protocols or otherwise) may require the provision of addressing attributes which are not adequately supported by Standard Attributes as defined in these Agreements. In such cases, Domain Defined Attributes are an acceptable means of carrying additional addressing information. Failure to support the specification and relaying of DDAs may prevent successful interworking with such existing systems until such time as all systems are capable of relaying and delivery using only the Standard Attribute list. Specific recommendations on the use of DDAs for particular applications are in the Recommended Practices, annex B.

5.3.6 P2 protocol profile (based on [X.420])

Tables 10 and 11 classify the support for the P2 protocol elements required by this profile. The tables give restrictions and comments in addition to X.420.

Restriction on length is one of the types of restrictions. The reaction of implementations to a violation of this restriction is not defined by this Profile.

5.3.6.1 P2 protocol - Heading

Table 10 specifies the support for protocol elements in P2 Headings.

Table 10 - P2 Heading protocol elements

Element	Class	Restrictions and Comments
UAPDU		
IM-UAPDU	G	
SR-UAPDU	X	
IM-UAPDU		
Heading	M	
Body	M	
Heading		
IPMessageId	M	
originator	R	
authorizingUsers	H	
primaryRecipients	G	At least one of primaryRecipients, copyRecipients, or blindCopyRecipients must be present.
copyRecipients	G	
blindCopyRecipients	H	
inReplyTo	G	
obsoletes	H	
crossReferences	H	
subject	G	Maximum length = 128 T.61 characters (256 octets);the ability to generate the maximum size subject is not required.
expiryDate	H	
replyBy	H	
replyToUsers	H	
importance	H	Appropriate action is for further study.
sensitivity	H	Appropriate action is for further study.
autoforwarded	H	

Table 10 - P2 Heading protocol elements (continued)

Element	Class	Restrictions and Comments
IPmessageId		
ORName	H	
PrintableString	M	Maximum length = 64 characters.
ORDescriptor		
ORName	H	Specify the ORName whenever it is possible. See annex B.
freeformName	H	Maximum length = 64 characters, graphical subset only (128 octets.)
telephoneNumber	H	Maximum length = 32 characters. This allows for punctuation. It does not take into account possible future use by ISDN.
Recipient		
ORDescriptor	M	
reportRequest	X	
replyRequest	H	
Body		No limit on number of BodyParts.
BodyPart	G	No limit on length of any BodyPart or the depth of ForwardedIPMessage BodyParts nested. Classification is subject to pending CCITT resolution
SR-UAPDU		
nonReceipt	H	
receipt	H	
reported	M	
actualRecipient	R	
intendedRecipient	H	
converted	X	
NonReceiptInformation		
reason	M	
nonReceiptQualifier	H	
comments	H	Maximum length = 256 characters.
returned	H	May only be issued if specifically requested by originator.

Table 10 - P2 Heading protocol elements (concluded)

Element	Class	Restrictions and Comments
ReceiptInformation		
receipt	M	
typeOfReceipt	H	
SupplementaryInformation	X	Maximum length = 64 characters. Note: This value is pending verification by the CCITT SG VIII or IX.
End of Definitions		

5.3.6.2 P2 protocol - BodyParts

5.3.6.2.1 BodyPart identifiers

All BodyParts with identifiers in the range 0 up to and including 16K -1 are legal and should be relayed. BodyPart identifiers corresponding to X.121 Country Codes should be interpreted as described in Note 2 of figure 4.

a) Implementations are required to generate and image IA5Text.

b) Implementations should specify the other BodyPart types supported.

c) If an implementation supports a particular BodyPart type for reception, it should also be able to support that BodyPart type for reception if it is part of a ForwardedIPMessage.

d) For the BodyPart types currently considered, support for the protocol elements is as indicated in table 11.

5.3.6.2.2 Privately defined BodyParts

This clause describes an interim means for identifying privately defined BodyParts. This clause shall be replaced in a future version taking into account CCITT recommendations with equivalent functionality.

```
BodyPart                  :: =  CHOICE {
                                  [0] IMPLICIT IA5Text,
                                  [1] IMPLICIT TLX,
                                  .
                                  .
                                  .
                                  [234] IMPLICIT UKBodyParts,
                                  .
                                  .
                                  .
                                  [310] IMPLICIT USABodyParts,
                                  .
                                  .
                                  .    }
-- Where UKBodyParts and USABodyParts are defined as:
                             SEQUENCE { BodyPartNumber, ANY }
BodyPartNumber            ::=  INTEGER
```

Notes

1 In the EncodedInformationTypes of the P1 Envelope, the undefined bit must be set when a message contains a privately defined BodyPart. Each UA that expects such BodyParts should include undefined in the set of deliverable EncodedInformationTypes it registers with the MTA.

2 All BodyPartNumbers assigned must be interpreted relative to the BodyPart in which they are used, which is that tagged with the value [310] for those defined within the United States. The NIST assigns unique message BodyPartNumbers for privately defined formats within the United States.

3 Refer to clause 12.6 for recommendations regarding the implementaion of USABodyParts.

Figure 4 - X.409 definition of privately defined BodyParts.

5.3.6.3 P2 BodyPart protocol elements

Table 11 - P2 BodyParts

Elements	Class	Restrictions and Comments
BodyPart		
IA5Text	G	
TLX	X	
Voice	X	
G3Fax	X	
TIFO	X	
TTX	X	
Videotex	X	
NationallyDefined	X	
Encrypted	X	
ForwardedIPMessage	H	
SFD	X	
TIF1	X	
unidentified	X	
IA5Text		
repertoire	H	
IA5String	M	For rendition of IA5Text see annex C.
TLX		For further study by CCITT.
Voice		
Set		For further study by CCITT.
BitString	M	
G3Fax		
numberOfPages	X	
P1.G3NonBasicParameters	X	
SEQUENCE (OF BIT STRING)	M	
BIT STRING	H	See Note.
P1.G3NonBasicParameters		Support for individual elements is for further study.
TIFO		
T.73Document	M	
T.73ProtocolElement	H	See Note.

Table 11 - P2 BodyParts (continued)

Elements	Class	Restrictions and Comments
TTX		
numberOfPages	X	
telexCompatible	X	
P1.TeletexNonBasicParams	X	
SEQUENCE	M	
T61String	H	See Note.
P1.TeletexNonBasicParams		
graphicCharacterSets	X	
controlCharacterSets	X	
pageFormats	X	
miscTerminalCapabilities	X	
privateUse	X	
Videotex		
SET		For further study by CCITT.
VideotexString	M	
NationallyDefined		
ANY	M	
Encrypted		
SET		For further study by CCITT.
BIT STRING	M	
ForwardedIPMessage		
delivery	H	
DeliveryInformation	H	
IM-UAPDU	M	
DeliveryInformation		
P1.ContentType	M	
originator	M	
original	M	
P1.Priority	G	
DeliveryFlags	M	
otherRecipients	H	
thisRecipient	M	
intendedRecipient	H	
converted	X	
submission	M	

Table 11 - P2 BodyParts (concluded)

Elements	Class	Restrictions and Comments
SFD		
SFD.Document	M	
TIF1		
T73 Document	M	
T73.ProtocolElement	H	See note.

Note: This element is not an addition to the definition of the BodyPart. It is described here to show that the SEQUENCE may contain zero elements. A Problem Report has been submitted to the CCITT to clarify whether this is permissible. The NIST/OSI Workshop will adopt the CCITT decision.

5.4 Reliable Transfer Server (RTS)

5.4.1 Implementation strategy

Based on X.410 Clause 3 and X.411 Clause 3.5.

5.4.2 RTS option selection

The maximum number of simultaneous associations is not limited in this profile; if the capacity of a system is exceeded, it should not initiate or accept additional associations.

Associations are established by the MTA which has messages to transfer.

Associations are released when they are not needed. Associations may also be ended prematurely due to internal problems of the RTS.

For both monologue and two way alternate associations, the initiator keeps the initial turn.

When establishing an RTS association, the following rules apply to the use of parameters in addition to those in X.410 Clause 3.2.1:

a) Dialogue mode: Monologue must be supported for this profile; two-way alternate is used only if both partners agree.

b) Initial turn: Kept by the initiator of the association.

The "priority-mechanism" and the "transfer-time limit" are regarded as local matters.

5.4.3 RTS protocol options and clarifications

Realization of the RTS protocol is subject to the following rules in addition to those specified in X.410 Clause 4:

a) One RTS association corresponds to one or more consecutive session connections (not concurrent ones). The first is opened with ConnectionData of type OPEN, and subsequent ones are opened with type RECOVER.

b) Recovery of a Session connection is only by RTS initiator.

c) *Checkpoint size*:

1) Checkpointing and No Checkpointing should be supported. Whenever possible, checkpointing should be used.

2) The minimum checkpointSize is 1 (that is, 1024 octets).

d) *Window size*:

1) Minimal value of 1 (if checkpointing is supported).

2) WindowSize = 1 means: After an S-SYNCH-MINOR request is sent, wait until the confirmation is received before issuing an S-DATA, S-SYNCH-MINOR, or S-ACTIVITY-END request.

e) APDUs should not be blocked into one activity.

f) Only one SSDU shall be transferred:

1) Between two adjacent minor synch points.

2) Between minor synch points and adjacent S-ACTIVITY-START and S-ACTIVITY-END requests.

3) Between S-ACTIVITY-START and S-ACTIVITY-END without checkpoints.

g) A *monologue association* is defined as follows:

1) The RTS user responsible for establishing the association is called the initiator.

2) The initiator keeps the initial turn.

3) APDUs are transferred in the direction of the initiator to the recipient only.

4) There shall be no token passing.

5) Only the initiator can effect an orderly release of the association.

h) A two-way alternate association is as described in X.410.

i) In the UserData parameter of the S-U-ABORT, the ReflectedParameter will not be used in the AbortInformation element.

j) When the S-ACTIVITY-RESUME is used to resume an activity in the same session connection as the one in which it started, this must happen immediately after the activity has been interrupted (i.e., no intervening activity can occur). Otherwise, X.410 Clause 4.3 paragraph 1 may be violated.

k) When S-ACTIVITY-RESUME is used to resume an activity started in another session connection, the following conditions must be met:

1) The current session connection is of type "recover."

2) The value of OldSessionConnectionIdentifier in S-ACTIVITY-RESUME must match the value of the SessionConnectionIdentifier parameter used in the S-CONNECT of the prior session connection. This value is also identical to the SessionConnectionIdentifier in the ConnectionData (in PConnect, in SS-UserData) for the current session connection.

3) This must occur as the first activity of the next session connection for the same RTS-association. It must be the first, otherwise X.410 Clause 4.5.1 point 1 is violated.

NOTE - It is in the same RTS-ASSOCIATION because the use of S-ACTIVITY-RESUME only makes sense within the scope of one RTS association.

l) If the transfer of an APDU is interrupted before the confirmation of the first checkpoint, the value of the SynchronizationPointSerialNumber in S-ACTIVITY-RESUME should be zero, and the S-ACTIVITY-RESUME must be immediately followed by an S-ACTIVITY-DISCARD.

m) In S-TOKEN-PLEASE, the UserData parameter shall contain an integer conforming to X.409 which conveys the priority.

n) The receiving RTS can use the value of the Reason parameter in the S-U-EXCEPTION-REPORT to suggest to the sending RTS that it should either interrupt or discard the current activity. S-U-Exception Reports are handled as stated in Version 5 of the Implementors Guide pages 12-13, paragraph E-33.

o) In the case of S-P-ABORT, the current activity (if any) is regarded as interrupted, rather than discarded.

p) Table 12 illustrates the legal negotiation possibilities allowed by X.410 Clause 4.2.1 regarding checkpoint size and window size.

q) These agreements include the provisions of Version 6 of the Implementors Guide identical in all respects to Version 5, except that the following points have been added to clause 3.5:

1) for section 4.4.1 of X.410; "If the receiving RTS receives an S-ACTIVITY-DISCARD indication primitive and has already issued a TRANSFER indication primitive, it aborts the connection (S-U-ABORT request) with the 'transfer completed' version code."

2) for section 4.6.2 of X.410 "The 'transfer completed (7)' abort reason is used to indicate to the sending RTS that the receiving RTS could not discard the activity because it has already completed the transfer (issued a TRANSFER indication primitive)." Transfer completed (7) is also added to the table of abort reasons in this clause.

Table 12 - Checkpoint window size of IP

		acceptor answer		
		CS = 0 (or unspecified) WS unspecified	CS = m WS = j (or unspecified)	CS = n WS = j (or unspecified)
initiator proposal	CS = O (or unspecified) WS = i (or unspecified)	legal	legal	legal
	CS = k WS = i (or unspecified)	legal	legal	not allowed

Legend
CS: means CheckpointSize
WS: means WindowSize
i, j, k, m, and n: are integer values with the following relations:
$0 \le m \le k < n$ (values assigned to CS)
$0 < j \le i$ (values assigned to WS)
For unspecified parameters, the default applies. In this case, the numeric relations apply, that is, the default values substitutefor the unspecified integer.

5.4.4 RTS protocol limitations

The RTS Protocol Limitations for this profile are listed in table 13.

Table 13 - RTS protocol elements

Element	Class	Restriction
PConnect	M	
DataTransferSyntax	M	Value = 0.
pUserData	M	
checkpointSize	H	
windowSize	H	
dialogueMode	H	
ConnectionData	M	
applicationProtocol	G	Value = 1.
	H	Value = 8883.
ConnectionData		
open	G	
recover	G	
open		
RTS user data	G	
recover		
SessionConnectionIdentifier	G	
RTS user data		
mTAName	G	Maximum length 32 characters graphic subset of IA5 only.
password	G	Maximum length 64 octets graphic subset of IA5 only.
< null RTS User Data >	G	Generated if other validation methods are used.
SessionConnectionIdentifier		
CallingSSUserReference	M	Maximum length 64 octets including encoding = 62 octets of T.61.
CommonReference	M	
AdditionalReferenceInformation	H	Maximum length 4 octets including encoding = 2 octets of T.61.
PAccept	G	
DataTransferSyntax	M	Value = 0.
pUserData	M	
checkpointSize	H	
windowSize	H	
ConnectionData	M	

Table 13 - RTS protocol elements (concluded)

Element	Class	Restriction
PRefuse	G	
RefuseReason	M	
SS User Data (in S-TOKEN-PLEASE)	G	See Note
AbortInformation (in S-U-ABORT)	G	
AbortReason	H	
reflectedParameter	X	Restricted to 8 bits.
End of Definitions		

Note - Generated if supplied by the RTS-user. The RTS use may specify a priority in the TURN-PLEASE primitive, and if so, it is carried as the SS-User-Data in S-TOKEN-PLEASE.

5.5 Use of session services

The session requirements and use of session are covered in part 5 of this document.

5.6 Data transfer syntax

This clause defines Presentation Transfer Syntax and notation rules applicable to these agreements. Implementations must conform EXACTLY as specified in X.409 with no further restrictions. Annex C defines rendition of IA5 Text and T61 characters.

6 PRMD to ADMD and ADMD to ADMD

6.1 Introduction

This clause defines the implementation agreements that apply to the interface between two management domains when at least one is an ADMD. A message arriving at an ADMD has either no recipient within that domain or one or more recipients within that domain. In the former case, the ADMD serves as a relay between two or more domains and the actions required of that ADMD are independent of the nature (PRMD or ADMD) of the domains. In the latter case, the ADMD is responsible for delivering messages to the proper recipient(s) within its jurisdiction, and may also be responsible for relaying the message.

Given the two roles for an ADMD, this clause describes two distinct sets of functional requirements for an ADMD. The first is the relaying requirement that is needed to provide PRMD and other ADMD interworking. The second is analogous to the PRMD's support to its customers through the integrated UAs. These are distinct functional differences. The services provided to the UAs of an ADMD are independent of the requirement that an ADMD provide a function for interworking with any type of Management Domain (MD). Figure 5 illustrates the two roles played by an ADMD.

This clause is presented in the form of deviations from the agreements applicable to PRMD-to-PRMD (sec. 5). Unless explicitly noted in the remainder of this clause, all of the specifications for PRMD to PRMD apply to PRMD to ADMD and ADMD to ADMD.

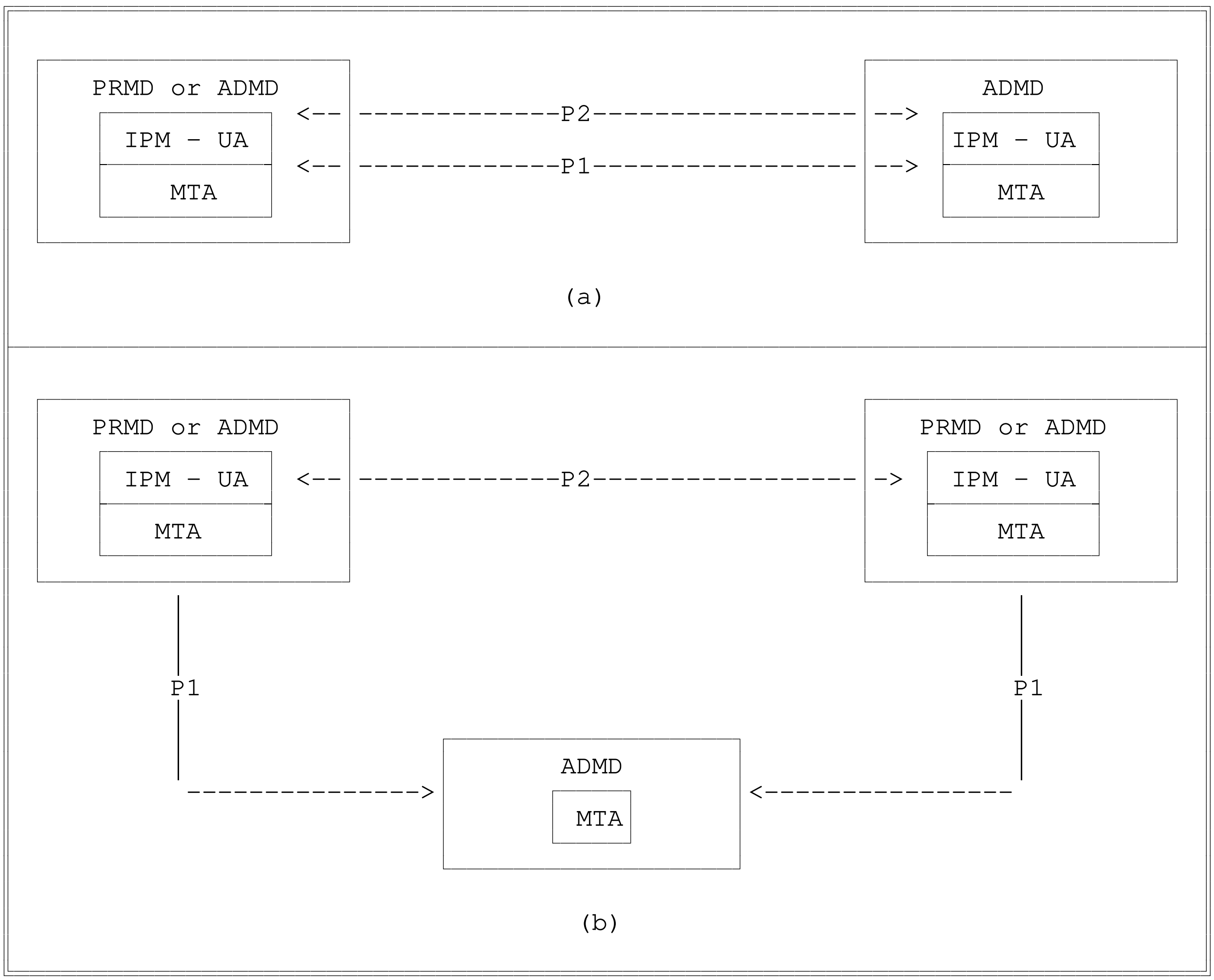


Figure 5 - An ADMD May (b) or May Not (a) Serve as a Relay.

6.2 Additional ADMD functionality

The following defines the additional ADMD specific functionality required over and above that specified in the PRMD clause.

6.2.1 Relay responsibilities of an ADMD

ADMDs will relay all content types (not just P2) unchanged in the absence of a request for conversion.

6.2.2 P1 protocol classification changes

Table 14 describes the changes to the PRMD P1 Protocol classifications required for a delivering Administration domain (with respect to the original message; this means the domain which originates the delivery reports).

Table 14 - P1 protocol classification changes for a delivering ADMD

Protocol Elements	Class	
DeliveredInfo		
typeOfUA	H	
ReportedRecipientInfo		
SupplementaryInformation	H	See Note 1.
GlobalDomainIdentifier		
PrivateDomainIdentifier	H	

For relaying Administration domains, the classifications are all "X"

For originating Administration domains, these are all "NOT APPLICABLE."

Notes

1 Domains providing access to TELEX/TELETEX recipients, whether directly or indirectly as a result of bilateral agreements between domains, must ensure that this information, when present, is accessible by the recipient of the delivery report.

6.2.3 O/R Names

O/R Names shall consist of:

a) CountryName,

b) AdministrationDomainName.

as well as one or more of the following:

a) PrivateDomainName,

b) PersonalName,

c) OrganizationName,

d) OrganizationalUnit,

e) UniqueUAIdentifier,

f) X121Address.

g) DomainDefinedAttributeList. (An implementation may accept or reject an OR Name that only contains country, ADMD, and DDA list.)

NOTE - The destination PrivateDomainName or OrganizationName must be present if destined for a PRMD. The ADMD relaying the message to that destination PRMD requires this element.

6.2.4 P1 ADMD name

Management Domains (MDs) must specify in the ADMD name field of the O/R Name StandardAttributeList in P1, the name of the Administration domain:

a) to which the message is being sent (in recipient names)

b) from which the message originated (in the originator name).

6.2.5 Interworking with integrated UAs

If the message originates at a UA owned by an ADMD, or is delivered to such a UA, the O/R Name follows the same Form 1 Variant 1 constraints as the base specifications; except that the ADMD name is the name of the ADMD that owns the UA and instead of supplying a PRMD Name, one (or more) of the following must be provided:

a) OrganizationName,

b) OrganizationalUnit,

c) PersonalName.

d) DomainDefinedAttributeList. (An implementation may accept or reject an OR Name that only contains country, ADMD, and DDA list.)

6.3 Differences with other profiles

6.3.1 TTC profile

There are no outstanding issues regarding interworking between TTC-conformant systems and NIST-conformant systems with the exception of the number of recipients and the supported MPDU sizes. The ExtensionIdentifier field may contain a maximum value of 32K-1; however, according to the current TTC profile, if a message with more than 256 recipients is received, some TTC-conformant domain may generate a nondelivery notification. This also applies to the ReportedRecipientInfo in a delivery report. Further, a TTC MTA is required to handle an MPDU size of at least 32KB. The NIST required MPDU size is 2MB as covered in clause 5.3.3. Other differences are shown in annex E. TTC is currently based on Version 4 of the Implementor's Guide.

6.3.2 CEPT profile

See annex E.

6.4 Connection of PRMDs to multiple ADMDs

Given that Management Domain names (both PRMD and ADMD) shall be unique within the United States, then when an ADMD is presented a message for transfer from a PRMD, it will accept O/R Names (both originator and recipient) which have an AdministrationDomainName field value different than the Administration's name. "Accept" implies the attempt to route/deliver the message shall be made, as appropriate, based upon the knowledge that MD names are unique.

Whether this functionality is required by an Administration for conformance to this agreement is for further study.

If a PRMD is connected to two or more ADMDs which are not effectively connected (either directly or via a third ADMD), full X.400 functionality shall not be available. Problems occur especially in the areas of:

a) Naming,

b) Routing,

c) Replying.

It should be noted that a single PRMD that is connected to more than one ADMD can be represented by more than one combination of country-name, ADMD-name, and PRMD name. For example, it may occur that the PRMD-name for a particular PRMD may take different values, depending on the ADMD-name. Implementors should be aware of the consequences of these possibilities on routing.

6.5 Connection of an ADMD to a routing PRMD

It is possible for a collection of interconnected private domains to establish one domain as the "gateway" to an ADMD, and hence to the world.

If an ADMD is connected to such a gateway PRMD, the individual private domains shall be registered with the Administration. Administrations need not support such connections.

Note also that upon receipt by the ADMD of a message originating somewhere within the PRMD collection, that the TraceInformation may contain more than one element.

The X.400 Recommendations specify that an ADMD should not attempt to relay a message destined for another ADMD through a PRMD, thus an ADMD should ensure that messages destined for another ADMD are not relayed through a PRMD. It should be noted, however, that a relaying PRMD will relay any such message it receives.

6.6 Management domain names

All Management Domain Names (both Private and Administration) shall be unique within the U.S.

A central naming authority shall be established to register domain names.

6.7 Envelope validation errors

For validation errors, a non-delivery notice shall be generated (if possible) with reason code of "unableToTransfer" and diagnostic code of "invalidParameters" (unless specified otherwise).

ADMDs will validate P1 Envelopes in the following areas:

a) The X.409 syntax of all elements should be checked.

b) The pragmatic constraint limits (lengths of fields and number of occurrences of fields) should be checked.

c) Semantic validation of the following elements should be done:

1) originator O/R Name,

2) recipient O/R Name in the RecipientInfo,

3) Priority.

d) Only recipient Names with the responsibility flag set should be validated. The validation of O/R names is defined in 8.3.3; the validation of priority is defined in 8.3.7.1.

e) MPDU Identifier Validation

1) Validation of the GlobalDomainIdentifier component of the MPDU Identifier is performed upon reception of a message (i.e., as a result of a TRANSFER.Indication).

2) The country name should be known to the validating domain, and depending on the country name, validation of the ADMD name may also be possible.

3) Additional validation of the GlobalDomainIdentifier is performed against the corresponding first entry in the TraceInformation. If inconsistencies are found during the comparison, a non-delivery notice with the above defined reason and diagnostic codes is generated.

4) A request will be generated to the CCITT for a more meaningful diagnostic code (such as "InconsistentMPDUIdentifier").

6.8 Quality of service

6.8.1 Domain availability

6.8.1.1 ADMD availability

The goal is to provide 24 hour per day availability. Note that there will be periods of time when an ADMD may be unavailable due to maintenance windows in its supporting network or in an MTA within the domain.

6.8.1.2 PRMD availability

Although the goal of PRMD availability is also 24 hours per day, business reasons are likely to dictate some different level of availability. ADMDs shall require a profile from the PRMD that indicates its schedule of regular availability to the ADMD.

6.8.2 Delivery times

In the absence of standardized quality of service parameters, the following are agreed to. When standardized parameters from CCITT Study Group I become available, they shall be adopted.

a) In table 15 the delivery time targets are established.

b) The interval(s) between retries and the number of retry attempts that an ADMD uses in attempting delivery to a PRMD or integrated UA, will be locally determined domain parameters. However, the total elapsed times after which delivery attempts will be stopped are shown in table 16. This implies that, after these times, a Non-Delivery Notice will be generated.

c) An Administration shall continue to attempt delivery until the forced nondelivery time, even if the recipient domain has scheduled an unavailability window.

Table 15 - Delivery time targets

Delivery Class	95% Delivered Before
Urgent	3/4 hour
Normal	4 hours
Non-Urgent	24 hours

Table 16 - Forced nondelivery times

Delivery Class	NonDelivery Forced After
Urgent	4 hours
Normal	24 hours
Non-Urgent	36 hours

NOTE - Both tables apply to the period between acceptance by the originating MTA in the originating Administration domain to the time of delivery in the destination Administration domain. Transit time within PRMDs is NOT included in the above times.

6.9 Billing information

All aspects relating to billing, charging, tariffs, settlement, and in particular to the use of the billingInformation field in the delivery report, is subject to bilateral agreement, and shall not be addressed in these implementation agreements.

No ADMD shall require a PRMD to supply or process billing information.

6.10 Transparency

No P1 extensions, other than the MOTIS extensions are to be allowed (Reference A/3211). Should an ADMD receive a message containing P1 extensions, it shall generate a non-delivery notice (if possible) with reason code of unableToTransfer and diagnostic code of invalidParameters.

If MOTIS elements are present, a relaying MTA can optionally:

a) Relay the message. If the MTA does relay, it must not drop any of the protocol elements.

b) Non-Deliver the message.

A receiving MTA can optionally:

a) Deliver the message

b) Non-Deliver the message.

The CCITT has been requested to establish a more meaningful diagnostic code (such as protocolError) for this occurrence. Such a code has now been provided in the Implementors Guide.

P2 extensions shall be relayed transparently by ADMDs.

6.11 RTS password management

RTS password management shall be a local matter. This includes:

a) password length

b) frequency of changes

c) exchange of passwords with communicating partners

d) loading passwords (i.e., the timing of password changes with respect to active associations).

6.12 For further study

Issues requiring further study are:

a) Intra-Domain Routing

b) Multi-Vendor Domains

7 Inter and intra PRMD connections

7.1 Introduction

This clause is limited in scope to issues arising from the indirect connection of a PRMD to another PRMD or to an ADMD, and to the interconnection of MTAs to form inter-PRMD connections. Indirect means that the connection is made via a relaying PRMD. The X.400 Recommendations describe the way that a PRMD connects to a ADMD and the way that an ADMD connects to another ADMD. The Recommendations do not, however, describe the way that a PRMD connects indirectly to an ADMD or another PRMD, nor do they describe the way that MTAs are connected within a PRMD. These configurations (shown in figures 6 and 7) are useful, for example, in connecting equipment from different vendors at a single customer site.

The P1 protocol and its related services for both inter and intra PRMD connections are addressed in this clause. In addition, a method for routing within a PRMD is given. It is recognized that uniform methods for Administration, maintenance, and quality of service should be developed for such configurations, and this work is for further study.

This clause describes the minimum that must be provided in order to implement a relaying PRMD and a MTA within a PRMD.

This clause is presented in the form of deviations from agreements applicable to PRMD to PRMD connection (sec. 5). That is, unless specifically noted in the remainder of this clause, the agreements in clause 5 apply to both relaying PRMDs and MTAs within a PRMD.

It should be noted that the comments regarding ORNames in clause 6.5 also apply to this clause.

7.2 The relaying PRMD

A PRMD that has the capability of relaying messages to another PRMD is called a relaying PRMD. A PRMD implementation need not claim to be a relaying PRMD. A PRMD implementation which does claim to be a relaying PRMD must follow the implementation agreements in this clause.

7.2.1 Relay responsibilities of a PRMD

The responsibilities of a relaying PRMD are the same as those of an ADMD (as specified in secs. 6.8 and 6.2.1). In addition, the PRMD will simply deliver messages routed to it from an ADMD, even if this results in routing a message from the ADMD to the PRMD to another ADMD.

7.2.2 Interaction with an ADMD

In order for an ADMD to route a message to ADMD A via ADMD B, it must know that A is reachable through B. Similarly, in order for any MD to route a message to PRMD A via a relaying PRMD B, it must know that A is reachable through B (see figure 8).

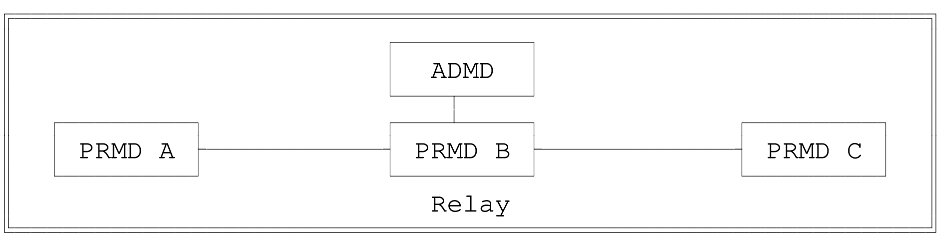


Figure 6 - Relaying PRMD.

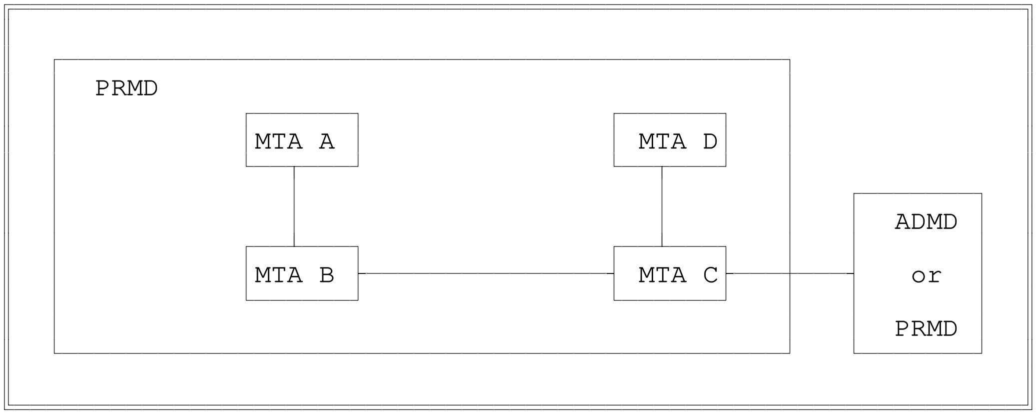


Figure 7 - Intra PRMD connections.

NOTES

1 Clause 6.6 specifies that ADMDs are not required to connect to a relaying PRMD, but they are not precluded from doing so.

2 TraceInformation may have more than one sequence on submission of a message by a relaying PRMD to an ADMD.

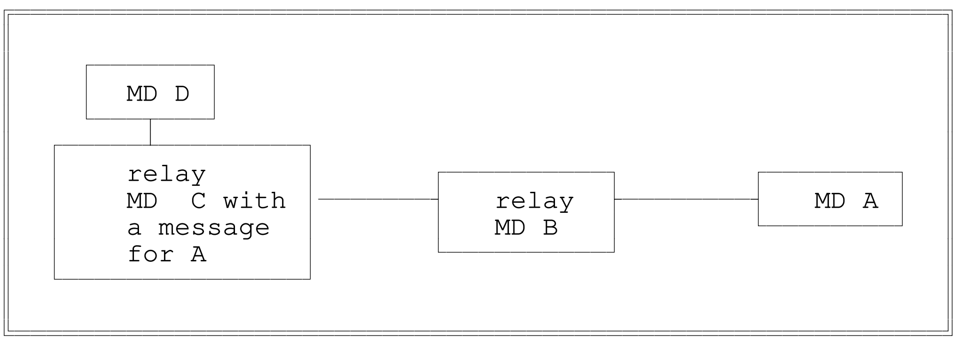


Figure 8 - MD C must know of A to route the message.

7.3 Intra PRMD connections

A PRMD is composed of MTAs which cooperate to perform the functions expected of a domain. An MTA implementation need not claim to follow the implementation agreements of this clause.

7.3.1 Relay responsibilities of an MTA

The relaying responsibilities of an MTA are the same as those of an ADMD (as specified in 6.8 and 6.2.1) with one exception: loop suppression within the domain is done using the MOTIS InternalTraceInfo protocol element. The MTA must validate the InternalTraceInfo (see 8.3.5 for details on validation). In addition, the PRMD will simply deliver messages routed to it from an ADMD, even if this results in routing a message from the ADMD to the PRMD to another ADMD (please see 6.6).

7.3.2 Loop suppression within a PRMD

The only mechanism defined in the X.400 Recommendations for suppressing loops is TraceInformation, which is added on a per domain basis to detect and suppress loops among domains. Loops among MTAs within a domain need to be detected and suppressed. This implies that each MTA must add trace information that is meaningful within the domain. The MOTIS solution of adding InternalTraceInfo to the P1 Envelope of a message was adopted. The definition of InternalTraceInfo is given in figure 9. The InternalTraceInfo is added by each MTA within a PRMD to handle a message, and it is examined in the same way as TraceInformation to detect and suppress loops.

```
InternalTraceInfo    ::= [APPLICATION 30] IMPLICIT SEQUENCE OF SEQUENCE {
                             MTAName,
                             MTASuppliedInfo }

MTAName              ::= PrintableString
```

Figure 9 - Definition of InternalTraceInfo.

If the MTAName and password of X.411 are used for validation, then it is recommended that the MTAName used for validation also be used in the InternalTraceInfo. However, there is a complication: in X.411, MTAName is an IA5String, and the MTAname defined by MOTIS is a PrintableString. Efforts will be made to change the MOTIS definition from PrintableString to IA5String.

Three actions are defined in MTASuppliedInfo: relayed, rerouted, and recipientReassignment as shown in figure 10. The recipientReassignment action is not supported in these agreements. The ability to generate it is not required, and if it is present on an incoming message, the action field will be ignored.

```
MTASuppliedInfo       ::= SET {
  arrival                   [0] IMPLICIT Time,
  deferred                  [1] IMPLICIT Time OPTIONAL,
  action                    [2] IMPLICIT INTEGER {
                                  relayed               (0),
                                  rerouted              (1),
                                  recipientReassignment (2) }
  previous                  MTAName OPTIONAL }
```

Figure 10 - Defined actions in MTASuppliedInfo.

7.3.3 Routing within a PRMD

Routing within a PRMD is complicated by the lack of a directory standard. In particular, it constrains intra-domain routing decisions to be based on some combination of the intra-domain attributes of the O/R Name, Organization Name Organizational Units, and Personal Name. In order to enhance interworking and to reduce the difficulty of configuring intra-domain connections, it is useful to restrict the ways in which these may be used in making routing decisions.

However, it is recognized that vendors may wish to provide MTAs with varying degrees of routing capability within a PRMD as a temporary expedient until appropriate standards for automated construction of directories and routing tables are available. This clause assigns class numbers to certain levels of routing capability and discusses the consequences of using MTAs which fall into each class. The classification scheme will allow some diversity in allocating O/R Name space and in configuring intra-domain routes.

When other methods are recommended by standards bodies, the classification scheme described here will become obsolete. Large-scale, multi-vendor PRMDs may not be practical in the absence of standardized methods.

7.3.3.1 Class designations

When it is clear that a message is to be delivered within a domain, the Country Name, ADMD Name, and PRMD Name have already served their purpose in determining the next MTA in the route to the recipient. The remaining fields that might be used on making routing decisions within the PRMD are the Organization Name, Organizational Units, and Personal Name.

MTAs are classified by their ability to discriminate between O/R names when making routing decisions within a PRMD. Conformant MTAs will be classified as shown in table 17.

Table 17 - Conformant MTA classifications

	Class 1	Class 2	Class 3
Organization Name	H	H	H
SEQUENCE OF Organizational Unit	X	H	H
Personal Name	X	X	H

An "H" means that the MTA must be able to base its intra-domain routing decisions on the given component of the O/R Name. In particular, both Class 2 and Class 3 MTAs must be able to discriminate on all the members in a supplied sequence of OrganizationalUnits. A Class 3 MTA must be able to discriminate on all of the elements in a PersonalName.

An "X" means that the MTA need not have the ability to discriminate on the given component.

There is a hierarchy in support of components. The ability to discriminate on a given component does not imply the requirement to do so: e.g., a Class 3 MTA is not required to have tables for every PersonalName in the domain. Equally, an MTA which can discriminate on OrganizationalUnits to make routing decisions need not always use the full sequence in an O/R Name if a partial sequence provides enough information.

The above classifications only apply to routing decisions in selecting a next hop within a domain. All MTAs are entitled to examine the full O/R Name when identifying their own directly served UAs.

The routing table of a Class 1 MTA will be relatively small, because intra-domain routing decisions are based solely on OrganizationName. The routing table of a Class 2 MTA may be substantially larger and more complex because of its ability to discriminate on OrganizationalUnits as well as OrganizationName to make routing decisions. The routing table of a Class 3 MTA may be larger still, because its use of the components of PersonalName in addition to the other information.

7.3.3.2 Specification of MTA classes

If an MTA implementation claims to follow the implementation agreements, it must be either a Class 1, Class 2, or a Class 3 MTA. The class of an MTA implementation should be specified so that PRMD administrators can choose equipment properly.

7.3.3.3 Consequences of using certain classes of MTAs

Definition: An MTA which accepts submission requests and furnishes delivery indications to a UA is said to "directly serve" the UA.

The presence in a domain of an MTA acting as a Class 1 or Class 2 MTA imposes administrative restrictions on the assignment of O/R Names to UAs and in the configuration of routes within that domain.

A Class 1 MTA may directly serve UAs from several OrganizationNames. However, if a Class 1 MTA directly serves a UA with a given OrganizationName, no other MTA in the domain may directly serve a user with the same OrganizationName. This means that if all MTAs in a domain are Class 1, then all UAs with a given OrganizationName must be assigned to the same MTA.

A Class 2 MTA may directly serve UAs from any combination of OrganizationName and sequence of OrganizationalUnits. However, if a Class 2 MTA directly serves a UA with a given combination, no other MTA in the domain may directly serve a user with the same combination. This means that if all MTAs in a domain are Class 2, then all UAs with a given OrganizationName and sequence of OrganizationalUnits must be assigned to the same MTA.

A domain consisting entirely of Class 3 MTAs is free of all the above restrictions.

If Class 1 or Class 2 MTAs are used to perform relaying within a PRMD containing MTAs of other classes, care must be exercised in determining the topology of the domain to avoid leaving certain UAs inacessible from certain MTAs within the domain. The example below shows one of the configurations that should be avoided. The example is intended to stimulate careful examination of the relationship between network and organizational topologies.

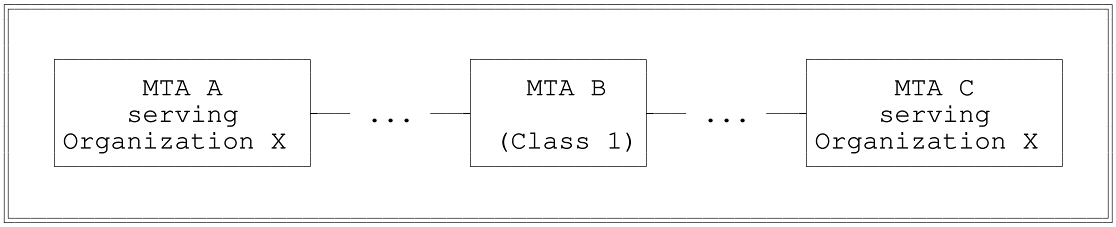


Figure 11 - Example of a configuration to be avoided.

In figure 11, B will route all messages for Organization X to either A or C because B is a Class 1 MTA. The administrator who created this configuration probably wanted B to route some messages for Organization X to A, and some to C. However, B does not have the capability for this because it is only a Class 1 MTA. The configuration in figure 11 can be corrected by replacing B with a Class 2 or Class 3 MTA.

7.3.4 Uniqueness of MPDUIdentifiers within a PRMD

When generating an IA5String in an MPDUIdentifier, each MTA in a domain must ensure that the string is unique within the domain. This shall be done by providing an MTA designator with a length of 12 octets which is unique within the domain, to be concatenated to a per message string with maximum length of 20 octets.

Two pieces of information, the MTA name and MTA designator, need to be registered within a PRMD to guarantee uniqueness. This registration facility need not be automated. If the MTA name is less than or equal to 12 characters, it is recommended that it also be used as the MTA designator.

7.4 Service elements and optional user facilities

A PRMD made up of MTAs which support varying sets of service elements in addition to those required in these agreements may appear to provide inconsistent service for these elements. For example, if one MTA supports deferred delivery and another MTA does not, then deferred delivery can be used by some, but not all, users in the PRMD. Similarly, if one MTA supports return of contents and another does not, then a user outside of the PRMD will receive returned contents for messages sent to one user, but not for messages sent to another user. Note that this same inconsistency occurs when sending to two PRMDs which support different additional optional elements.

7.5 X.400 protocol definitions

This clause describes additions and modifications to clause 5.3 which are required for implementation of a relaying PRMD or an MTA within a PRMD.

7.5.1 Protocol classification

The classification scheme given in clause 5.3.1 applies to elements passing from one PRMD to another. For both relaying PRMDs, and MTAs in a PRMD, the same classification scheme will be used, but within a PRMD the classification applies to elements passing from one MTA to another.

In addition to the classifications given in clause 5.3.1, a classification of Prohibited has been used.
PROHIBITED = P

This element shall not be used. Presence of this element is a protocol violation.

7.5.2 P1 protocol elements

Table 18 contains protocol elements and their classes. An * signifies that the classification of the protocol element has not changed from table 8.

Table 18 - P1 protocol elements

Element	Class	Restrictions and Comments
UMPDUEnvelope		
MPDUIdentifier	M*	This field needs to be unique within a PRMD. See clause 7.3.4 for the method of ensuring uniqueness.
originator	M*	It is recommended that all components of the originator's ORName be included to help ensure that reports can be returned.
TraceInformation	M*	The first MTA in the domain to receive the message adds the TraceInformation. Subsequent MTAs can update the TraceInformation in the event of conversion or deferred delivery. When a message is generated, the originating MTA adds the TraceInformation.
InternalTraceInfo	M/P	This element is mandatory for envelopes transferred between MTAs within a PRMD, and prohibited in messages transferred outside the domain. Elements are always added to the end of the sequence. (See Note 1)
InternalTraceInfo MTAName	M	MTANames within a PRMD must be unique. See clause 7.3.4 for the method of assuring uniqueness Maximum length = 32 characters.
MTASuppliedInfo	M	

Table 18 - P1 protocol elements (continued)

Element	Class	Restrictions and Comments
MTASuppliedInfo		
arrival	M	
deferred	X	This field must be generated by MTAs which perform deferred delivery.
action	M	See clause 7.3.2 for restrictions on values of this field.
previous	X	This field must be generated by MTAs which perform rerouting.
DeliveryReportEnvelope		
TraceInformation	M*	The first MTA in the domain to receive the report adds the TraceInformation. When a report is generated, the originating MTA adds the TraceInformation.
InternalTraceInfo	M/P	This field is mandatory for envelopes transferred between MTAs within a PRMD, and prohibited in messages transferred outside the domain. (See Note 1)
DeliveryReportContent		
intermediate InternalTraceInfo	P	If it were possible to include this field in the delivery report content, an audit and confirmed report could be provided to detect problems within a PRMD. Efforts are being made to add this field to the MOTIS definition.
DeliveredInfo		
typeOFUA	R*	It is the responsibility of the MTA generating the report to generate this element.

Table 18 - P1 protocol elements (concluded)

Element	Class	Restrictions and Comments
ProbeEnvelope		
TraceInformation	M*	The first MTA in the domain to receive the probe adds the TraceInformation. When a probe is generated, the originating MTA adds the TraceInformation.
InternalTraceInfo	M/P	This field is mandatory for envelopes transferred between MTAs within a PRMD, and prohibited in messages transferred outside the domain. (See Note 1)

Notes

1 The M classification is only applicable if an implementation is claiming conformance according to clause 10.2.

7.5.3 Reliable Transfer Server (RTS)

In the pUserData of PConnect, the value of applicationProtocol should be 1. This value was chosen because the agreements on intra-domain connections are not strictly P1, nor are they MOTIS. Philosophically, it would be good to choose a new application protocol identifier for these agreements, but this introduces too many practical problems. Since these agreements are closer to P1 than to MOTIS, the value of 1 will be used. This will not cause interworking problems between domains, because the only deviation from P1 is the InternalTraceInfo, which will not be present in messages transferred outside of a domain.

8 Error handling

This clause describes appropriate actions to be taken upon receipt of protocol elements which are not supported in this profile, malformed MPDUs, unrecognized O/R Name forms, content errors, errors in reports, and unexpected values for protocol elements.

8.1 MPDU encoding

The MPDU should have a context-specific tag of 0, 1, or 2. If it does not have one of these tags, it is not possible to figure out who originated the message. Therefore, the way this error is reported is a local matter.

8.2 Contents

Once delivery to the UA has occurred, it is not possible to report errors in P2 information to the originator. In addition, it seems unreasonable to insist that the MTA that delivers a message ensure that the P2 content of the message is acceptable. As a result, the handling of content errors is a local matter.

8.3 Envelope

This clause describes the handling of errors in message envelopes. Some of the error conditions described below may be detected in a recipient's O/R Name. This may limit the reporting MTA's ability to generate a nondelivery notification that accurately reflects the erroneous O/R Name in the ReportedRecipientInfo. This handling of this situation is a local matter.

8.3.1 Pragmatic constraint violations

In all cases of pragmatic constraint violation, a nondelivery report should be generated with a ReasonCode of unableToTransfer, and a DiagnosticCode of pragmaticConstraintViolation.

8.3.2 Protocol violations

If all required protocol elements are not present, a nondelivery report with a ReasonCode of unableToTransfer and a DiagnosticCode of protocolViolation should be generated.

If a protocol element is expected to be of one type, but is encoded as another, then a nondelivery report with a ReasonCode of unableToTransfer and a DiagnosticCode of invalidParameters should be generated.

8.3.3 O/R Names

The domain that has responsibility for delivering a message should also have the responsibility to send the nondelivery notification if the message cannot be delivered. Therefore, each MTA should only validate the O/R Names of recipients with responsibility flags set to TRUE. In addition, a nondelivery notification can only be sent if the originator's O/R Name is valid.

If any element in the O/R Name is unrecognized or if the CountryName, AdministrationDomainName, and one of PrivateDomainName and OrganizationName (and, for ADMDs, PersonalName and OrganizationalUnit) are not all present, then a nondelivery report should be generated with a ReasonCode of unableToTransfer, and a DiagnosticCode of unrecognizedORName. If the message can be delivered even though the ORName is invalid, delivery is a local matter. Note, however, that if the message is delivered, the invalid ORName might be propagated through the X.400 system (e.g., by forwarding).

If the O/R Name has all of the appropriate protocol elements and the message still cannot be delivered to the recipient, the following DiagnosticCodes may appear in the nondelivery report: unrecognizedORName,ambiguousORName,and uaUnavailable.

8.3.4 TraceInformation

Since non-relaying domains need not do loop suppression, domains with responsibility for delivering the message need not be concerned about the semantics of the TraceInformation, that is, arrival time and converted EncodedInformationTypes can be provided to the UA without inspection by the MTAs of the domain as long as the TraceInformation is properly encoded according to X.409.

When a message is accepted for relay, the relaying domain must check that a TraceInformation SEQUENCE has been added by the domain that last handled the message. If the appropriate TraceInformation was not added, this should be treated as a protocolViolation (sec. 8.3.2).

In addition, the relaying domain must check that the information was added in the sequence defined by the rules for adding TraceInformation in the CCITT X.400 Implementor's Guide. If the sequence is invalid,then a nondelivery report should be generated with a ReasonCode of unableToTransfer and a diagnosticCode of invalidParameters.

> **NOTE -** It would be desirable for the CCITT to add a diagnostic code of invalidTraceInformation to allow a more meaningful description of this problem. A request for this new diagnostic code will be submitted.

8.3.5 InternalTraceInfo

This clause applies only to MTAs which follow the agreements of clause 7.

When a message is accepted for relay from another MTA in the domain, the relaying MTA must check that an InternalTraceInfo SEQUENCE has been added by the MTA that last handled the message. If the appropriate InternalTraceInfo was not added, this should be treated as a protocolViolation (sec. 8.3.2).

In addition, the relaying MTA must check that the information was added in the sequence defined by the rules for adding TraceInformation in the CCITT X.400 Implementor's Guide. If the sequence is invalid, then a nondelivery report should be generated with a ReasonCode of unableToTransfer and a diagnosticCode of invalidParameters.

> **NOTE -** It would be desirable for the CCITT to add a diagnostic code of invalidTraceInformation to allow for a more meaningful description of this problem. A request for this new diagnostic code will be submitted.

8.3.6 Unsupported X.400 protocol elements

The protocol elements defined in X.400 but unsupported by this profile are: the deferredDelivery and PerDomainBilateralInfo parameters of the UMPDUEnvelope, the ExplicitConversion parameter of RecipientInfo, and the alternateRecipientAllowed and contentReturnRequest bits of the PerMessageFlag. Appropriate actions are described below for domains that do not support the protocol elements.

8.3.6.1 deferredDelivery

The delivering domain shall do one of the following:

a) deliver at once,

b) hold for deferred delivery,

c) return a nondelivery notification with a ReasonCode of unableToTransfer and a DiagnosticCode of noBilateralAgreement.

8.3.6.2 PerDomainBilateralInfo

If a delivering domain receives this element, the element can be ignored.

8.3.6.3 ExplicitConversion

If ExplicitConversion is requested the message should be delivered if possible. That is, if the UA is registered to accept the EncodedInformationTypes of the message, then the message should be delivered even though the requested conversion could not be performed along the route. If delivery is not possible, then a nondelivery report should be generated with a ReasonCode of conversionNotPerformed with no DiagnosticCode.

8.3.6.4 alternateRecipientAllowed

If a delivering domain receives this element the element can be ignored.

8.3.6.5 contentReturnRequest

If a delivering domain receives this element, the element can be ignored.

8.3.7 Unexpected values for INTEGER protocol elements

There are three INTEGERs in the P1 Envelope. Appropriate actions are described below for domains receiving unexpected values for Priority, ExplicitConversion, and ContentType.

8.3.7.1 Priority

Additional values for Priority have been suggested by at least one group of implementors as upward compatible changes to the X.400 Recommendations. Therefore, if a PRMD receives an unexpected value for Priority, and this value is greater than one byte in length, a nondelivery report should be generated with a ReasonCode of unableToTransfer and DiagnosticCode of invalidParameters. If the value is less than or equal to one byte, the PRMD can either generate a nondelivery report as previously specified or interpret the Priority as normal and deliver or relay the message.

8.3.7.2 ExplicitConversion

When an unexpected value is received for ExplicitConversion, it should be handled as in clause 8.3.6.3.

8.3.7.3 ContentType

If the ContentType is not supported by the delivering MTA, then a nondelivery report should be generated with a ReasonCode of unableToTransfer, and a DiagnosticCode of contentTypeNotSupported.

8.3.8 Additional elements

In the absence of multilateral agreements to the contrary, receipt of privately tagged elements and protocol elements in addition to those defined in X.400 will result in a nondelivery report with a ReasonCode of unableToTransfer and a DiagnosticCode of invalidParameters.

The exceptions to this are the MOTIS elements. The treatment of MPDU's containing these MOTIS extensions is described in clause 6.11.

8.4 Reports

There is no mechanism for returning a delivery or status report due to errors in the report itself. Therefore the handling of errors in reports is a local matter.

9 MHS use of Directory Services

9.1 Directory service elements

Recommendation X.400 recognizes the need of MHS users for a number of directory service elements. Directory service elements are intended to assist users and their UAs in obtaining information to be used in submitting messages for delivery by the MTS. The MTS may also use directory service elements to obtain information to be used in routing messages. Some functional requirements of directories have been identified and are listed below:

a) Verify the existence of an O/R name.

b) Return the O/R address that corresponds to the O/R name presented.

c) Determine whether the O/R name presented denotes a user or a distribution list.

d) Return a list of the members of a distribution list.

e) When given a partial name, return a list of O/R name possibilities.

f) Allow users to scan directory entries.

g) Allow users to scan directory entries selectively.

h) Return the capabilities of the entity referred to by the O/R name.

i) Provide maintenance functions to keep the directory up-to-date.

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

Currently, these aspects of directory service elements and procedures are under study by both the CCITT and the ISO. Both organizations are committed to the development of a single Directory Service specification for use by MHS and all other OSI based applications.

Given the incomplete nature of the ongoing activities within the CCITT and the ISO, **no implementation details will be provided now for MHS use of Directory Services**. Implementation agreements for MHS Use of Directory Services will be issued when current activities within the CCITT and the ISO are stable.

9.2 Use of names and addresses

It is recognized that these agreements enable a wide variety of naming and addressing attributes (see sec. 5.3.5 ORName Protocol Elements) wherein each PRMD may adopt particular routing schemes within its domain.

With the exception of the intra-domain connection agreements:

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

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

a) manual directories

b) on-line directories

c) ORName address specifications

d) ORName address translation.

Further, each PRMD may adopt naming and addressing schemes wherein the user view may take a form entirely different from the attributes reflected in table 9. And, each PRMD may have one user view for the originator form and another for the recipient form, and perhaps other forms of user addressing. In some cases (e.g., receipt notification) these user forms must be preserved within the constraints of these implementation agreements. However, mapping between one PRMD user form to another PRMD user form, via the X.400 ORName attributes of these agreements, is outside the scope of these agreements.

10 Conformance

10.1 Introduction

In order to ensure that products conform to these implementation agreements, it is necessary to define the types and degrees of conformance testing that products must pass before they may be classified as **conformant**. This clause defines the conformance requirements and provides guidelines for the interpretation of the results from this type of testing.

This clause is incomplete and will be enhanced in future versions of this Agreement. Later versions will reflect the problems of conformance testing and will outline specific practices and recommendations to aid the development of conformance tests and procedures.

10.2 Definition of conformance

For this clause, the term **conformance** is defined by the following:

a) The tests indicated for this clause are intended to establish a high degree of confidence in a statement that the implementation under test (IUT) **conforms** (or does not **conform**) to the agreements of this clause.

b) **Conformance** to a service element means that the information associated with the service element is made accessible to the user (person or process) whenever this agreement says that this information should be available. Accessible means that information must be provided describing how a user (person or process):

1) causes appropriate information to be displayed, or

2) causes appropriate information to be obtained.

c) **Conformance** to P1, P2, and RTS as part of an X.400 OSI application requires that only the external behavior of that OSI system adheres to the relevant protocol standards. In order to achieve **conformance** to this clause, it is not required that the inter-layer interfaces be available for testing purposes.

d) **Conformance** to the protocols requires:

1) that MPDUs correspond to instances of syntactically correct data units,

2) MPDUs in which the data present in the fields and the presence (or absence) of those fields is valid in type and semantics as defined in X.400, as qualified by this profile,

3) correct sequences of protocol data units in responses (resulting from protocol procedures).

e) Statements regarding the **conformance** of any one implementation to this profile are not complete unless a Protocol Implementation Conformance Statement (PICS) is supplied.

f) The term "Implementation Under Test" (IUT) is interchangeable with the term "system" in the definition of **conformance**, and may refer to:

1) a domain, which may be one or more MTA's with co-located or remote UA's,

2) a single instance of an MTA and co-located UA with X.400 (P1, P2, RTS and session) software,

3) a relaying product with P1, RTS and session software,

4) a gateway product.

g) Claiming Implementation Conformance

1) An implementation which claims to be conformant as an ADMD must adhere to the agreements in clauses 5 and 6.

2) An implementation which claims to be conformant as a PRMD must adhere to the agreements in clause 5.

3) An implementation which claims to be conformant as a relaying PRMD must adhere to the agreements in clause 5 and the appropriate clauses of 7.

4) An implementation which claims to be conformant to the intra-domain connection agreements must adhere to the agreements in clause 5 and the appropriate clauses of 7.

10.3 Conformance requirements

10.3.1 Introduction

Conformance to this specification requires that all the services listed as supported in clauses 5, 6, and if appropriate, 7 of these agreements are supported in the manner defined, in either the CCITT X.400 Recommendations or these agreements. It is not necessary to implement the recommended practices of annex B, in order to conform to these agreements.

It is the intention to adopt, where and when appropriate the testing methodology and/or the abstract test scenarios currently being defined by the CCITT X.400 Conformance Group. However, it is recognized that formal CCITT Recommendations relating to X.400 Conformance Testing will not be available until 1988. It is also recognized that aspects of these agreements are outside the scope of the CCITT, and that other organizations will have to provide conformance tests in these cases.

10.3.2 Initial conformance

This clause is intended to provide guidelines to vendors who envisage having X.400 products available prior to any formal mechanism, or "Conformance Test Center" being made accessible that would allow for conformance to this product specification to be tested.

It is feasible that vendors and carriers will want to enter bilateral test agreements that will allow for initial trials to be carried out for the purposes of testing initial interworking capabilities. It is equally feasible that for the purposes of testing interoperability, only a subset of this specification will initially be tested.

NOTE - By claiming conformance to this subset of information the vendor or carrier CANNOT claim conformance to this entire specification.

There are two aspects to the requirements, interworking and service, as described in the following clauses.

10.3.2.1 Interworking

The interworking requirements for conformance implies that tests be done to check for the syntax and semantics of protocol data elements for a system as defined by the classification scheme of clauses 5.2.1.1 and 7.5.2. For a relay system, the correct protocol elements should be relayed as appropriate. For a recipient system, a message with correct protocol elements must not be rejected where appropriate.

10.3.2.2 Service

For information available to the recipients via the IPMessage Heading and Body, the following should be made accessible:

a) IPMessage ID - only the PrintableString portion of the IPMessageId needs to be accessible.

b) subject,

c) primaryRecipients,

d) copyRecipients,

e) blindcopyRecipients,

f) authorizingUsers,

g) originator,

h) inReplyTo,

i) replyToUsers,

j) importance,

k) sensitivity,

l) IA5Text Bodypart.

Annex A (normative)

Interpretation of X.400 service elements

The work on service element definitions is limited to those that are defined as "supported" in clause 5 of this specification. Furthermore it is not the intent of this clause to define how information should be made available or presented to a MHS user, nor is it intended to define how individual vendors should design their products. In addition, statements on conformance to a specific service element and the allocation of error codes that are generated as a result of violations of the service should be defined in the clauses on conformance and errors as part of the main product specification. The main objective is to provide clarification, where required, on the functions of a service element, and in particular what the original intent of the Recommendations were.

A.1 Service elements

The following Service Elements defined in X.400 have been examined and require further text to be added to their definitions to represent the proposed implementation of these service elements by the X.400 SIG.

The service element clarifications are to be taken in the context of this profile.

Service elements not referenced in this clause are as defined in X.400.

A.2 Probe

A PRMD need **not** generate probes.

If a probe is addressed to and received by a PRMD, the PRMD **must** respond with a **Delivery Report** as appropriate at the time the probe was processed.

A.3 Deferred delivery

In the absence of bilateral agreements to the contrary, Deferred Delivery and Deferred Delivery Cancellation are local matters (i.e., confined to the originating domain) and need **not** be provided.

The extension of Deferred Delivery beyond the boundaries of the initiating domain is via bilateral agreement as specified in section 3.4.2.1 of X.411.

A.4 Content type indication

It is required that both an originating and recipient domain be able to support P2 content type. The ability for domains to be able to exchange content types other than P2 will depend on the existence of bilateral or multi-lateral agreements.

A.5 Original encoded information types indication

It is required that both an originating and recipient domain be able to support IA5 text. Support for other encoded information types, for the purposes of message transfer between domains, will depend on the existence of bilateral or multi-lateral agreements.

The use of the "unspecified" form of encoded information type should only be used when the UMPDU content represents an SR-UAPDU or contains an auto-forwarded IM-UAPDU.

The original encoded information type of a message is not meaningful unless a message is converted en route to the recipient. These agreements support only IA5 text, which should not undergo conversion. The original encoded information types should be made accessible to the recipient for upward compatibility with the use of non-IA5 text message body parts.

A.6 Registered encoded information types

A UMPDU with an "unspecified" value for Original Encoded Information Type shall be delivered to the UA.

A.7 Delivery notification

The UAContentID may be used by the recipient of the delivery notification for correlation purposes.

A.8 Disclosure of other recipients

This service is not made available by originating MTAE's to UAE's, but must be supported by relaying and recipient MTAE's.

By supporting the disclosure of other recipients the message recipient can be informed of the O/R names of the other recipient(s) of the message, as defined in the P1 envelope, in addition to the O/R Descriptors within the P2 header.

These agreements do not support initiation of disclosure of other recipients, but the information associated with it should be made accessible to the recipient for upward compatibility with support for the initiation of this service element.

A.9 Typed body

As defined in X.400 with the addition of the Private Body Types that are to be supported. At present there is no mechanism provided within X.420 that would allow you to respond to reception of an unsupported body type.

Action taken in this situation is a local matter.

A.10 Blind copy recipient indication

It should be considered that the recipient's UA acts on behalf of the recipient, and therefore may choose to disclose all BCC recipients to each other. Therefore it is the responsibility of the originating domain to submit two or more messages, depending on whether or not each BCC should be disclosed to each other BCC.

A.11 Auto forwarded indication

A UA may choose not to forward a message that was previously auto-forwarded. In addition there is no requirement for an IPM UA that does not support non-receipt or receipt notification to respond with a non-receipt notification when a message is auto-forwarded.

A.12 Primary and copy recipients indication

It is required that at least one primary recipient be specified; however, for a forwarded message this need not be present. The recipient UA should be prepared to accept no primary and copy recipients to enable future interworking with Teletex, Fax, etc.

A.13 Sensitivity indication

A message originator should make no assumptions as to the semantic interpretation by the recipients UA regarding classifications of sensitivity. For example, a personal message may be printed on a shared printer.

A.14 Reply request indication

In requesting this service an originator may additionally supply a date by which the reply should be sent and a list of the intended recipients of the reply. If no such list is provided then the initiator of the reply sends the reply to the originator of the message and any recipients the reply initiator wishes to include. The replytoUsers and the replyBy date may be specified without any explicit reply being requested. This may be interpreted by the recipient as an implicit reply request. Note that for an auto-forwarded message an explicit or implicit reply request may not be meaningful.

A.15 Body part encryption

The original encoded information type indication includes the encoded information type(s) of message body parts prior to encryption by the originating domain. The ability for the recipient domain to decode an encrypted body part is a local matter. Successful use of this facility can only be guaranteed if there exists bilateral agreements to support the exchange of encrypted body parts.

A.16 Forwarded IP message indication

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

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

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

c) See ammex B on recommended practices for the use of the delivery information as part of Forwarded IP-message.

A.17 Multipart body

It is the intent of multipart bodies to allow for the useful and meaningful structuring of a message that is constructed using differing body part types. For example, it is not recommended that a message made up of only IA5 text should be represented as a number of IA5 body parts, each one representing a paragraph of text.

Annex B (informative)

Recommended X.400 practices

It is not necessary to follow the recommended practices when claiming conformance to these agreements.

B.1 Recommended practices in P2

B.1.1 ORDescriptor

Vendors following the OSI Implementors' Workshop guidelines shall, whenever possible, generate the ORName portion of an ORDescriptor in ALL IPM heading fields.

B.1.2 ForwardedIPMessage BodyParts

ForwardedIPMessage BodyParts should be nested no deeper than eight. There is no restriction on the number of ForwardedIPMessage BodyParts at any given depth.

B.1.3 DeliveryInformation

It is strongly recommended that DeliveryInformation be supplied in both forwarded and autoforwarded message body parts. DeliveryInformation is useful when a message has multiple forwarded message body parts because without it, the EncodedInformationType(s) of the component forwarded messages cannot be deduced easily. DeliveryInformation is useful for autoforwarded messages because the EncodedInformationType of an autoforwarded message is "unspecified" and the EncodedInformationType(s) of the message cannot be determined easily without it. Absence of the EncodedInformationType(s) makes it difficult for a UA to easily determine whether the message can be rendered.

B.2 Recommended practices in RTS

B.2.1 S-U-ABORT

In the case where S-U-ABORT indicates a temporaryProblem, reestablishment of the session should not be attempted for a "sensible" time period (typically not less than 5 min). In instances where this delay is not required or necessary, report a localSystemProblem.

B.2.2 S-U-EXCEPTION-REPORT

S-U-EXCEPTION-REPORT reason codes can be interpreted as follows:

B.2.2.1 receiving ability jeopardized (value 1)

Possible meaning: The receiving RTS knows of an impending system shutdown.

B.2.2.2 local ss-User error (value 5)

Possible meaning: The receiving RTS needs to resynchronize the session dialogue.

B.2.2.3 irrecoverable procedure error (value 6)

Possible meaning: The receiving RTS has had to delete a partially received APDU, even though some minor synchronization points have been confirmed.

B.2.2.4 non specific error (value 0)

Possible meaning: The receiving RTS cannot handle the APDU (for example, because it was too large) and wishes to inform the sending RTS not to try again.

B.2.2.5 sequence error (value 3):

Possible meaning: The S-ACTIVITY-RESUME request specified a minor synchronization point serial number which does not match the checkpoint data.

B.2.3 OSI addressing information

For purposes of identifying an MTA during an RTS Open, OSI addressing information should be used. This addressing information is conveyed by lower layer protocols and is reflected by the calling and called SSAP parameters of the S-CONNECT primitives.

MTA validation and identification are related, but separate, functions. The mTAName and password protocol elements of the RTS user data should be used for validation, rather that identification, of an MTA. The RTS initiator and responder may independently require each other to supply mTAName and password.

The CallingSSUserReference parameter of the S-CONNECT primitives should only have meaning to the entity that encoded it and should not be used to identify an MTA.

B.3 Recommended practices for ORName

Table 9 stipulates that the StandardAttributeList must contain either PrivateDomainName or OrganizationName. It is recommended that, for both originator and recipients in a private domain, the PrivateDomainName field be used.

It is recommended that there should be a DomainDefinedAttribute to be used in addressing UAs in existing mail systems, in order to curtail the proliferation of different types of DomainDefinedAttributes used for the same purpose. The syntax of this DomainDefinedAttribute conforms to the CCITT Pragmatic Constraints, and thus has a maximum value length of 128 octets and a type length of 8 octets, each of type Printable String. Only one occurrence is allowed.

This DomainDefinedAttribute has the type name "ID" (in uppercase). It contains the unique identifier of the UA used in addressing within the domain. This DomainDefinedAttribute is to be exclusively used for routing within the destination domain (i.e., once routed to that domain via the mandatory components of the StandardAttributeList); any other components of the StandardAttributeList may be provided. If they conflict delivery is not made.

The contents of this parameter need not be validated in the originating domain or any relaying domain, but simply transferred intact to the next MTA or domain.

Class 2 and class 3 MTAs in a PRMD should allow administrators to decide the number of OrganizationalUnits that should appear in user names, instead of imposing a software controlled limit which is less than four. This is desirable because when two different vendors impose different limits on the number of OrganizationalUnits in a name, it becomes difficult for the administrator to choose a sensible naming scheme.

There are existing mail systems that include a small set of non-Printable String characters in their identifiers. For these systems to communicate with X.400 messaging systems, either for pass-through service or delivery to X.400 users, gateways will be employed to encode these special characters into a sequence of Printable String characters. This conversion should be performed by the gateway according to a common scheme and before insertion in the ID DDA, which is intended to carry electronic mail identifiers. X.400 User Agents may also wish to perform such conversions.

It is recommended that the following symmetrical encoding and decoding algorithm for non-Printable String characters be employed by gateways. The encoding algorithm maps an ID from an ASCII representation to a PrintableString representation. Any non-printable string characters not specified in the table are covered by the category "other" in the table below.

The principal conversion table for the mapping is as follows:

Table B.1 - Printable String to ASCII mapping

ASCII Character	Printable String Character
% (percent)	(p)
@ (at sign)	(a)
! (exclamation)	(b)
" (quote mark)	(q)
_ (underline)	(u)
((left paren.)	(l)
) (right paren.)	(r)
other	(3DIGIT)

where 3DIGIT has the range 000 to 377 and is interpreted as the octal encoding of an ASCII character.

To encode an ASCII representation to a PrintableString, the table and the following algorithm should be used:

```
IF current character is in the encoding set THEN
encode the character according to the table above
ELSE
    write the current character;
continue reading;
```

To decode a PrintableString representation to an ASCII representation, the table and the following algorithm should be used:

```
IF current character is not "(" THEN
      write character
ELSE
      {
      look ahead appropriate characters;
      IF composite characters are in the above table THEN
         decode per above table
      ELSE
         write current character;
      }
continue reading;
```

Class 2 and class 3 MTAs in a PRMD should allow administrators to decide the number of OrganizationalUnits that should appear in user names, instead of imposing a software controlled limit which is less than four. This is desirable because when two different vendors impose different limits on the number of OrganizationalUnits in a name, it becomes difficult for the administrator to choose a sensible naming scheme.

B.4 Postal addressing

For domains wishing to support postal (or physical) delivery options, the following interim set of "nationally-defined" domain defined attributes are recommended. The CCITT will define Standard Attributes in support of physical delivery in its 1988 Recommendations; this is only an interim solution.

CCITT will also be addressing the services associated with physical delivery. This interim solution does not address the end-to-end service aspects of physical delivery; in particular, the following IPM service elements do not currently extend outside of the X.400 environment:

a) alternate Recipient Assignment

b) PROBE

c) Receipt Notification / Non-Receipt Notifications

d) Grade of delivery

"Delivery" means passing a message from the MTS to the physical delivery system (PDS), and not to the user (or user agent).

The following three DDAs are recommended to be used to specify a postal (or physical) address:

CNTRPC encodes the country and postal code for postal delivery. The DDA value is of the form "Country?Postalcode" (for example, "USA?22096"). The country field is optional, the postal code is optional; the separator ("?") is not. If both country and postal code are missing, this DDA should not be specified.

PDA1 The country and postal code fields are free-form text.

PDA2 These two DDA (signifying Postal Delivery Address strings 1 and 2) form a 256 character free-form postal address. Fields are separated by a question mark ("?"). There is no implied separator between PDA1 and PDA2. The meaning of the fields are defined by each domain supporting the physical delivery interface. PDA1 contains the first 128 characters, PDA2 the next 128 characters. If the PDA string is less than 128 characters, PDA2 is not used.

For example, if the domain interprets the PDA fields as lines, the address

```
Mr. John Smith
Conway Steel
123 Main Street
Reston VA 22096
```

would be encoded as follows:

```
type   = "PDA1"
value  = "Mr. John Smith?Conway Steel?123 Main Street?Reston VA"
CNTRPC = "?22096"
```

B.5 EDI use of X.400

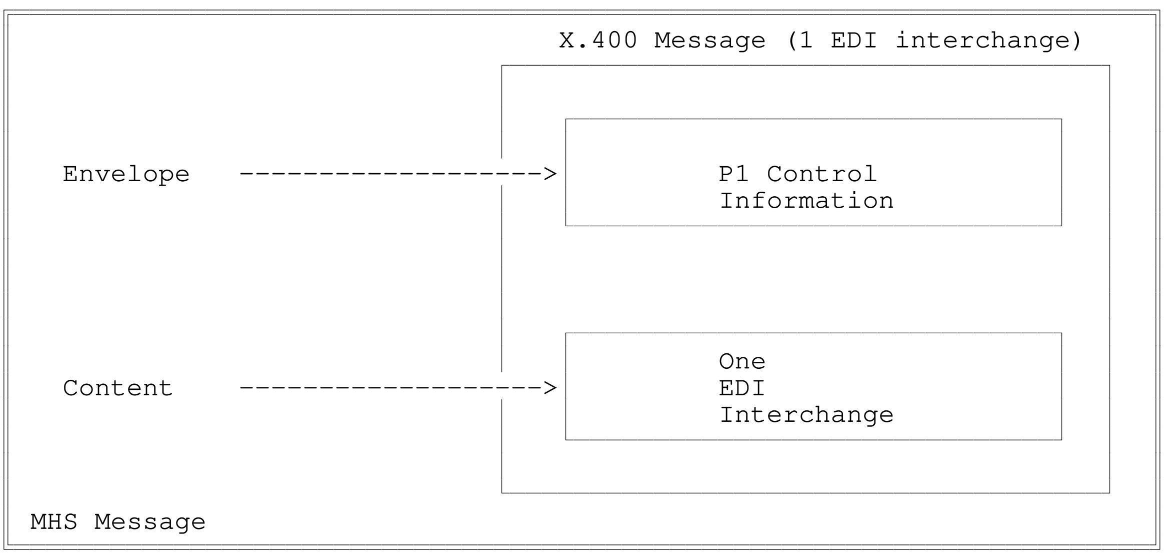
B.5.1 Introduction and scope

This is a guideline for EDI data transfer in an X.400 environment conforming to the NIST agreements. These recommended practices outline procedures for use in transferring EDI transactions between trading partner applications in an attempt to facilitate actual X.400 implementation by EDI users.

The scope of this guideline is to describe specific recommendations for adopting X.400 as the data transfer mechanism between EDI applications.

B.5.2 Model

The MHS recommendations can accommodate EDI through the approach illustrated below. Many Message Transfer (MT) service elements defined in the X.400 recommendations are particularly useful to the EDI application.



This diagram depicts an EDI content (1 EDI interchange) enveloped by the P1 MHS envelope. All the MT Services defined in the X.400 Recommendations may be used for EDI. However, it is not required to support optional or non-essential services to exchange EDI data between EDI users. When an EDI user submits an EDI Trade Document to the EDI User Agent, the EDI UA will submit the EDI content plus P1 envelope to the Message Transfer System (MTS).

```
                                  EDI                           EDI
   MTS      <----------->         UA      <------------>        User
```

The EDI UA must support the essential MT Services as defined in these Agreements; for example, as a minimum, to provide default values for services not elected by the EDI user, such as Grade of Delivery.

NOTE - MT Services are not necessarily made available by the EDI UA to the EDI user.

B.5.3 Protocol elements supported for EDI

The following P1 protocol elements will be used to support EDI applications:

B.5.3.1 Content type

For EDI applications, the content type will be 0 (undefined content).

B.5.3.2 Original encoded information types

Any EIT defined in the X.400 Recommendations may be used to specify the encoding of EDI content. However, for ANSI X12 EDI applications in particular, it is expected that the "undefined" and "Ia5Text" EIT's will normally be used, with "undefined" used to signify the EBCDIC character set.

B.5.4 Addressing and routing

It is anticipated that connection of some existing systems to an X.400 service for EDI purposes will be by other than X.400 protocols, at least in the short term.

EDI messages entering the X.400 environment will therefore need to have X.400 O/R Names added to identify the origination and recipient trading partners, typically by means of local directory services in the origination domain which will map EDI identifiers/addresses into O/R Names. Such O/R Names will contain Standard Attributes as defined in table 9 and for recipient trading partners will at least identify the destination domain.

In the case of trading partners outside the X.400 environment, it is expected, however, that there will be cases where message delivery will require the provision of addressing information beyond that which can be carried in Standard Attributes. In such cases, Domain Defined Attributes are recommended to be used.

The syntax of this DDA is as defined in table 9, with a single occurrence having the type name "EDI" (uppercase) and a value containing the identifier/address of the trading partner. For ASC X12 purposes, specifically, this value will comprise the 2 digit interchange ID qualifier followed by the interchange ID (max 15 characters). Routing on this DDA shall only occur, if at all, in the destination domain.

B.6 USA body parts

It is recommended that UAs can generate any USA Body Part, as defined in clause 5.3.6.2, and that they can receive such body parts as well. reception of USA Body Parts does not imply further processing by the UA, but merely that the body part is made available, with a indication of its registered body part identifier, to another process or deposition in a file. Generation implies the reverse of this process.

B.7 Recommended practices for binary data transfer

The capability to transfer binary data, such as those generated by word/document processing, spreadsheets, or graphics applications among X.400 system is a useful and desirable feature. Many messaging systems provide such capability today.

It is recommended that transfer of binary data through 1984-based systems be achieved using the unidentified BodyPart in P2 with the ASN1 definition recaptured as follows:

```
BodyPart                ::=  CHOICE {
                               [0] IMPLICIT IA5Text
                               ...
                               [14] IMPLICIT Unidentified
                               ... }
Unidentified            ::=  OCTET STRING
```

NOTE - the Unidentified BodyPart is included in 1984 X.400 Implementor's Guide, Version 6, and is renamed as BilaterallyDefinedBodyPart in 1988 X.400 Series with the same tag and definition.

Additionally the binary data can be identified by a text string in the subject heading or in an IA5Text body part preceding the Unidentified BodyPart.

When the Unidentified BodyPart is present in a P2 message, the undefined(0) bit of the P1 EncodedInformationTypes will be set. If the IA5Text bodypart is also present, the IA5text(2) bit will also be set.

The binary data is the raw data as generated by user applications. Besides encapsulating it for transfer purposes, X.400 systems do not encode or interprete the binary data in any way further. How the data is encoded or decoded is defined by the cooperating user applications. How the data is injected into X.400 systems or transferred out of X.400 systems to the user applications, or how the user applications are invoked to process the data is a local implementation issue and not defined.

B.8 Recommended practice for Office Document Architecture (ODA) transfer

It is recommended that the conveyance of ODA documents through 1984-based X.400 systems be achieved using the following schemes:

B.8.1 ODA In P2

An ODA document will be transferred as a single body part with tag 12, recaptured as follows:

```
BodyPart              ::=  CHOICE  {
                             [0]  IMPLICIT IA5Text
                             ...
  oda                        [12] IMPLICIT OCTET STRING
                             ... }
```

The content of the Octet String will contain a value of type OdaBodyPart as follows:

```
OdaBodyPart           ::=  SEQUENCE  {
                             OdaBodyPartParameters,
                             OdaData  }
```

The Parameters and Data components are defined in Annex E of CCITT Recommendation T.411 (1988) (ISO 8613-1).

B.8.2 ODA In P1

The undefined bit (bit 0) of the EncodedInformationTypes must be set and the ODA bit (bit 10) of the EncodedInformationTypes should be set when an ODA document is present in P2. However, MTAs should be tolerant of messages containing ODA documents being received with just the undefined bit (bit 0) set, and should still deliver the message.

B.8.3 Interworking with later versions of X.400

The 1988 X.419 Recommendation acknowledges that a 1984 system may receive messages containing new distinguished [integer] values that it is not expecting, and that this may result in service irregularities. It is implied that it would be optimal for 1984 systems to accept these unexpected integer values if at all possible. (Note clause 8.3.7 of this section describes appropriate actions for unexpected values of the INTEGER fields in the P1 envelope.) No downgrading should be done for these values when passing affected messages from newer systems to older systems.

Annex C (normative)

Rendition of IA5Text and T61String characters

C.1 Generating and imaging IA5Text

The characters that may be used in an IA5String are the graphic characters (including Space), control characters and Delete of the IA5 character repertoire ISO 646.

The graphic characters that may be used with a guaranteed rendition are those related with positions 2/0 to 2/2, 2/5 to 3/15, 4/1 to 5/10, 5/15 and 6/1 to 7/10 in the basic 7-bit code table.

The other graphic characters may be used but have no guaranteed rendition.

The control characters that may be used but have no guaranteed effect are a subset consisting of the format effectors 0/10 (LF), 0/12 (FF) and 0/13 (CR) provided they are used in one of the following combinations:

```
CR LF        to start a new line
CR FF        to start a new page (and line)
LF .. LF     to show empty lines (always after one of the
             preceding combinations).
```

The other control characters or the above control characters in different combinations may be used but have no guaranteed effect.

The character Delete may occur but has no guaranteed effect. The IA5String in a P2 IA5Text BodyPart represents a series of lines which may be divided into pages. Each line should contain from 0 to 80 graphic characters for guaranteed rendition. Longer lines may be arbitrarily broken for rendition. Note that X.408 states that for conversion from IA5Text to Teletex, the maximum line length is 77 characters.

C.2 Generating and imaging T61String

For further study.

Annex D (informative)

Differences in interpretation discovered through testing of the MHS for the CeBit 1987 demonstration

Several interworking problems were discovered through multi-vendor testing. These problems, and recommendations for solutions to them are discussed in this annex.

D.1 Encoding of RTS user data

The password is defined as an ANY in the X.400 Recommendations, and implementor's groups have decided to use an IA5String for this field. There was some confusion about what the X.409 encoding for this IA5String would be, and the correct encoding is:

```
class:          context specific
form:           constructor
id code:        1
length:         length of contents
contents:       (primitive encoding)
IA5String:      16
length:         length of contents
contents:       the password string
class:          context specific
form:           constructor
id code:        1
length:         length of contents
contents:       (constructor encoding) left as an exercise for the reader
```

Implementations should be prepared to receive any X.409 type for the password because of its definition as an ANY.

D.2 Extra session functional units

One vendor proposed more than the required set of functional units on opening the session connection, and the receiver rejected the connection. All debate aside about whether the initiator should have proposed units outside of the required set, or whether the receiver should have rejected the connection, the set of functional units can be negotiated in a straightforward way. The following is recommended.

If the initiator proposes using more than the required set of functional units, the responder should specify the set of functional units that it would like to use (which should include the required set) in the open response. The session implementations will automatically use the intersection of the units proposed by both sides.

If the initiator proposes using less than the required set of functional units, the responder should reject the connection. Unfortunately, there is not an appropriate RefuseReason for rejecting the connection, so instead of refusing the connection in the response to the S-CONNECT, the receiver should issue an S-U-ABORT with an AbortReason of protocolError. Note that it is valid to issue an S-U-ABORT instead of responding to the S-CONNECT. A problem report has been submitted to the CCITT requesting the addition of a RefuseReason for this situation.

If the responder proposes using less than the required set of functional units, the session connection is established before the initiator can check for this. If too few functional units have been proposed, the initiator should abort the connection using S-U-ABORT, with an abort reason of protocolError.

D.3 Mixed case in the MTA name

The MTA name is frequently exchanged over the telephone when two systems are being configured to communicate with one another. In one such telephone exchange, the casing of the MTA name was not specified, the MTA name consisted of both upper and lower case letters, and one of the implementations compared MTA names for equality in a case sensitive manner. Consequently, connections failed until the problem was detected and repaired. It is recommended that the MTA name be compared for equality in a case insensitive manner, and that the password be compared for equality in a case sensitive manner.

D.4 X.410 activity identifier

The X.400 Implementor's Guide recommends that the activity identifier be X.409 encoded, but this is only a recommendation and not a requirement. Consequently, receiving systems cannot assume that the activity identifier will be X.409 encoded.
Encoding of per recipient flag and per message fFlag
In the definition of the PerRecipientFlag in X.411, there is a statement that the last three bits are reserved, and should be set to zero. It is unclear whether those bits are unused in the X.409 encoding. Receivers should accept encodings with either zero or three unused bits. A problem report has been submitted to the CCITT asking for clarification.

Though there is not any statement in X.411 about the last four bits of the PerMessageFlag, some vendors have encoded this with zero unused bits, and some have encoded it with four unused bits. The PerMessageFlag should be encoded with at least four unused bits.

D.5 Encoding of empty bitstrings

There are three valid encodings for an empty bitstring: a constructor of length zero, a constructor of indefinite length followed by the end-of-contents terminator, and a primitive of length one with a zero octet as the value.

D.6 Additional octets for bitstrings

Nothing in X.409 constrains an implementation from sending two, three, four, or even more octets for a bitstring that fits into one octet, with the undefined bits set to zero. Note that the number of excess octets is bounded by the pragmatic constraints guidelines of the CCITT X.400 Implementor's Guide for all of the bitstrings in P1.

D.7 Application protocol identifier

If a value other that 1 is received in the applicationProtocol of the pUserData in the PConnect, NIST implementations will reject the connection. If CEN/CENELEC implementations receive a value other than 8883 for this field, they will reject the connection. This is an unfortunate state of affairs, because if NIST implementations accept the value of 8883 without supporting the MOTIS service elements, they would be misrepresenting themselves. To make matters worse, CEPT uses a value of 1, but relays MOTIS elements, which means that MOTIS elements will be relayed to implementations using a value of 1 to demonstrate that they do not support MOTIS. Work is continuing to try to find a solution that will allow European implementations to interwork with U.S implementations.

D.8 Initial serial number in S-CONNECT

This should be implemented in accordance with section 3.5.1 E4 of the Implementor's Guide.

D.9 Connection data on RTS recovery

It is clarified that the ConnectionData is identical in both the S-CONNECT.request and the S-CONNECT.response. The value of the ConnectionData is the old Session Connection Identifier.

D.10 Activity resume

If an activity is being resumed on a new session connection, it is not clear from X.410 and X.225 whether all four of the called-ss-user reference, the calling-ss-user reference, the common reference, and the additional reference information should be specified in the S-ACTIVITY-RESUME, or whether one of the ss-user-references should be absent. It is also unclear whether the called-ss-user reference should be identical to the calling-ss-user reference if both are present. Consequently, receivers should be tolerant of this situation. Appropriate problem reports will be submitted to the CCITT asking for clarification.

D.11 Old activity identifier

The Old Activity Identifier in S-ACTIVITY-RESUME refers to the original activity identifier.

D.12 Negotiation down to transport class 0

For European implementations, X.410 specifies that class 0 transport must be supported. However, it is permissible for an initiator to propose a higher class as the preferred class, provided that class 0 appears as the alternate class in the T-Connect PDU. A responding implementation can choose to use either the preferred or alternate class, but again, must be able to use class 0. In other words, for private to private connections in Europe, class 0 transport is required.

This conflicts with the OIW agreements, since class 0 is only required if one of the partners in a connection is an ADMD.

Annex E (informative)

Worldwide X.400 conformance profile matrix

Y CONFORMANCE (E) implies a conformance problem for European products in the United States.

Y CONFORMANCE (US) implies a conformance problem for U.S. products in Europe.

The A/311 profile is specified in Env 41 202, the A/3211 profile in Env 41 201

No TTC protocol classification for RTS exists.

The notation X/Y indicates "X" for PRMDs and "Y" for ADMDs, i.e., "M/G" would be **Mandatory** for PRMDs and **Generatable** for ADMDs.

Table E.1 - Protocol element comparison of RTS

RTS element	NIST	A/311	A/3211	PROBLEM Y/N
PConnect	M	M	M	N
DataTransferSyntax	M 0	M 0	M 0	N
PUserData	M	M	M	N
checkpointSize	H	H	H	N
windowSize	H	H	H	N
dialogueMode	H	H	H	N
connectdata	M	M	M	N
applicationProtocol	G 1 H 8883	H 1	R 8883	N
ConnectionData				
Open	G	G	G	N
Recover	G	H	G	N
Open				
RTSUserData	G	G	G	N
Recover				
SessionConnectionID	G	G	G	N
RTSUserData				
MTAName	G	G	G	N
Password	G	G	G	N
null	G	G	G	N
SessionConnectionID				
CallingUserReference	M	M	M	N
CommonReference	M	M	M	N
AdditionalRefInfo	H	H	H	N

Table E.1 - Protocol element comparison of RTS (concluded)

RTS element	NIST	A/311	A/3211	PROBLEM (Y/N)
PAccept	G	G	G	N
DataTransferSyntax	M 0	M 0	M 0	N
PUserData	M	M	M	N
CheckpointSize	H	H	H	N
WindowSize	H	H	H	N
ConnectionData	M	M	M	N
PRefuse	G	G	G	N
RefuseReason	M	M	M	N
SSUserData (in S-TOKEN-PLEASE)	G	G	G	N
AbortInformation (in S-U-ABORT)	G	G	G	N
AbortReason	H	H	H	N
reflectedParameter	X	X	X	N

Table E.2 - Protocol element comparison of P1

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
ORname					
StandardAttributeList	M	M	M	M	N See Note 4
DomainDefAttributeList	X	X	X	G	Y See Note 5
StandardAttributeList					
CountryName	R	R	R	M	N
		SO R	R		N
		.121	H		Y Conformance (E)
		ther	X		Y Prot Vio
AdministrationDomainName	R	R	G	M	N
... if PrintableString		R	G		N
... if numericString		H	H		Y Conformance (E)
X.121 Address	X	X/R	X		Conf(US)See Note 1
Terminal ID	X	X/G	X		Conf(US)See Note 1
PrivateDomainName	G	G	G	G	N
OrganizationName	G	G	G	G	N
UniqueUAidentifier	X	X/G	X		Conf(US)See Note 1
PersonalName	G	G	G	G	N
OrganizationalUnit	G	G	G	G	N
DomainDefinedAttribute	X	X	X	G	N
Type	M	M	M	M	N
Value	M	M	M	M	N
PersonalName					
Surname	M	M	M	M	N
GivenName	G	G	G	G	N
Initials	G	G	G	G	N
GenerationQualifier	G	X	X	X	Y Conformance (E)
GlobalDomainIdentifier					
CountryName	M	M	M	M	N
AdministrationDomainName	M	M	G	M	Y Proto Vio
PrivateDomainIdentifier	R/H	H	R	M/X	N
MPDU					
UserMPDU	G	G	G	G	Y TTC required MPDU size is 32K
DeliveryReportMPDU	G	G	G	G	N
ProbeMPDU	H	H	H	H	N

Table E.2 - Protocol element comparison of P1 (continued)

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
UserMPDU					
UMPDUenvelope	M	M	M	M	N
UMPDUcontent	M	M	M	M	N
UMPDUenvelope					
MPDUidentifier	M	M	M	M	N
originatorORname	M	M	M	M	N
originalEncodedTypes	G	H	H	G	Y Conformance (E)
ContentType	M	M	M	M	N
UAcontentID	H	H	H	H	N
Priority	G	G	G	G	N
PerMessageFlag	G	G	G	G	N
DeferredDelivery	X	X	X	X	N
PerDomainBilatInfo	X	X	X	X	N
RecipientInfo	M	M	M	M	Y TTC MPDU 32K
TraceInformation	M	M	M	M	N
MOTIS-> LatestDelivery			X		N
MOTIS-> InternalTraceInfo	M/P		P		N
UMPDUcontent	M	M	M	M	N
MPDUidentifier					
GlobalDomainIdent	M	M	M	M	N
IA5string	M	M	M	M	N
PerMessageFlag					
DiscloseRecipients	H	@ MT	H	H	Y Conformance (US)
		at U	?		Y Conformance (US)
ConversionProhibited	G	G	G	G	N
AlternatRecipAllowed	H	@ MT	H	X	Y Conformance (US)
		at U	?		Y Conformance (US)
ContentReturnRequest	X	X	X	X	
MOTIS-> redirectionProhibited			X		N
PerDomainBilateralInfo					
CountryName	M	M	M	M	N
AdminDomainName	M	M	G	M	Y Prot Vio
MOTIS-> PrivateDomainName			G		N
BilateralInfo	M	M	M	M	N

Table E.2 - Protocol element comparison of P1 (continued)

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
DeliveryReportContent					
original MPDUident	M	M	M	M	N
intermediate Trace	X/G	X	X	X	Y Conformance (E)
UAcontentID	G	G	G	G	N
ReportedRecipientInfo	M	M	M	M	Y TTC 256 max
returned	H	H	X	X	Y Conformance (E)
billing information	X	X	X	X	N
ReportedRecipientInfo					
recipient ORname	M	M	M	M	N
extensionsIdentifier	M	M	M	M	N
PerRecipientFlag	M	M	M	M	N
LastTraceInformation	M	M	M	M	N
intendedRecipient	H	H	H	H	N
SupplementaryInfo	X/H	X	X	X	Y Conformance (E)
MOTIS-> ReassignmentInfo			X		N
MOTIS-> ReassignmentInfo					
MOTIS-> intendedRecipient			M		N
MOTIS-> reasonForReassignment			H		N
LastTraceInformation					
arrival	M	M	M	M	N
convertedEncInfoTypes	G	G	H	G	Y Conformance (E)
Report	M	M	M	M	N
Report					
DeliveredInfo	G	G	G	M	N See Note 6
NonDeliveredInfo	G	G	G		N
DeliveredInfo					
delivery	M	M	M	M	N
TypeofUA	R/H	H	R	M/G	N
NonDeliveredInfo					
ReasonCode	M	M	M	M	N
DiagnosticCode	H	H	H	H	N
MOTIS-> UAprofileIdentifier			X		N
MOTIS-> UAprofileIdentifier					
MOTIS-> ContentType			M		N
MOTIS-> EncodedInfoTypes			M		N

Table E.2 - Protocol element comparison of P1 (continued)

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
ProbeEnvelope					
probe	M	M	M	M	N
originator	M	M	M	M	N
ContentType	M	M	M	M	N
UAcontentID	H	H	H	H	N
originalEncInfoTypes	G	H	H	G	Y Conformance (E)
TraceInformation	M	M	M	M	N
PerMessageFlag	G	G	G	G	N
ContentLength	H	H	H	H	N
PerDomainBilatInfo	X	X	X	X	N
RecipientInfo	M	M	M	M	Y TTC 256 max
MOTIS-> InternalTraceInfo	M/P		P		N
RecipientInfo					
RecipientORname	M	M	M	M	N
ExtensionIdentifier	M	M	M	M	N
PerRecipientFlag	M	M	M	M	N
ExplicitConversion	X	X	X	X	N
MOTIS-> OriginReqAlternatRecip			X		N
MOTIS-> ReassignmentInfo			X		N
PerRecipientFlag					
ResponsibilityFlag	M	M	M	M	N
ReportRequest	M	M	M	M	N
UserReportRequest	M	M	M	M	N
TraceInformation					
GlobalDomainIdent	M	M	M	M	N
DomainSuppliedInfo	M	M	M	M	N

Table E.2 - Protocol element comparison of P1 (concluded)

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
DomainSuppliedInfo					
arrival	M	M	M	M	N
deferred	X	X	X	X	N
action	M	M	M	M	N
(0=relayed)	G	G	G		N Note: Re-routing not required.
(1=rerouted)	H	H	H		N
MOTIS-> (2=recipientReassigned)			H		N
converted	H	G	H	H	Y Conformance(US)
previous	H	G	G	X	Y Conformance(US) (Note: G is inconsistent with action (relayed) being "H.")
ORname					
EncodedInformationTypes					
BitString	M	M	M	M	N See Note 3
G3NonBasicParameters	X	X	X	X	N
TeletexNonBasicParams	X	R	X	X	Y Conformance(US)
PresentationAbilities	X	X	X	X	N
DeliveryReportMPDU	G	G	M	G	N
DeliveryReportEnvelop	M	M	M	M	N
DeliveryReportContent	M	M	M	M	N
DeliveryReportEnvelope					
report	M	M	M	M	N
originator ORname	M	M	M	M	N
TraceInformation	M	M	M	M	N
InternalTraceInfo	M/P		P		N

Table E.3 - Protocol element comparison of P2

P2 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
UAPDU					
IM_UAPDU	G	G	G	G	N
SR_UAPDU	X	X	X	X	N
IM_UAPDU					
Heading	M	M	M	M	N
Body	M	M	M	M	N
Heading					
IPmessageID	M	M	M	M	N
Originator ORname	R	R	R	M/G	N
AuthorizingUsers	H	H	H	H	Y TTC 16 max
PrimaryRecipients	G	G	G	G	Y TTC 256 max
CopyRecipients	G	G	G	G	Y TTC 256 max
BlindCopyRecipient	H	H	H	H	Y TTC 256 max
InReplyTo	G	G	G	G	N
Obsoletes	H	H	H	H	Y TTC 8 max
CrossReferences	H	H	H	H	Y TTC 8 max
Subject	G	G	G	G	N
ExpiryDate	H	H	H	H	N
ReplyBy	H	H	H	H	N
ReplyToUsers	H	H	H	H	Y TTC 32 max
Importance	H	H	H	H	N
Sensitivity	H	H	H	H	N
Autoforwarded	H	H	H	H	N
MOTIS-> CirculationList			X		N
MOTIS-> ObsoletingTime			X		N
IPmessageID					
ORname	H	H	H	H	N
PrintableString	M	M	M	M	N
ORdescriptor					
ORname	H	H	H		N See Note 6
				M	
FreeFormName	H	H	H		N
TelephoneNumber	H	H	H	G	N
Recipient					
ORdescriptor	M	M	M	M	N
ReportRequest	X	X	X	X	N
ReplyRequest	H	H	H	H	N

Table E.3 - Protocol element comparison of P2 (continued)

P2 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
MOTIS-> CirculationList					
MOTIS-> CirculationMember			X		N
MOTIS-> checkmark			M		N
MOTIS-> membername			M		N
MOTIS-> OBsoletingTime					
MOTIS-> Time			H		N
MOTIS-> IP_MessageID			H		N
Body					
BodyPart	G	M	M	G	Y Conformance (US)
SR_UAPDU					
NonReceipt	H	H	H		N
				M	
Receipt	H	H	H		N
Reported	M	M	M	M	N
ActualRecipient	R	R	R	G	N
IntendedRecipient	H	H	H	H	N
Converted	X	X	X	G	N
MOTIS-> CirculationStatus			X		N
NonReceiptInformation					
Reason	M	M	M	M	N
NonReceiptQualifier	H	H	H	H	N
=expired (value)	0	0	0	0	N
=obsoleted (value)	1	1	1	1	N
=subscriptionTerminated	2	2	2	2	N
MOTIS-> =timeobsoleted (value)			X		N
Comments	H	H	H	X	N
returned	H	X	X	X	Y Conformance (E)
ReceiptInformation					
Receipt	M	M	M	M	N
TypeOfReceipt	H	H	H	G	N
SupplementaryInfo	X	X	X	X	N

Table E.3 - Protocol element comparison of P2 (concluded)

P2 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
BODYPART SUPPORT					
o IA5 Text	G	G	G		N See Note 7
o TLX	X	X	X		N
o Voice	X	X	X		N
o G3FAX	X	X	X		N
o TIFO	X	X	X		N
o TTX	X	X/H	X		Y Conf(US) See Note 2
o VideoTex	X	X	X		N
o NationallyDefined	X	X	X		N
o Encrypted	X	X	X		N
o ForwardedIPmessage	H	H	H		N
o SFD	X	X	X		N
o TIFI	X	X	X		N
MOTIS-> o ODA			X		N
MOTIS-> o ISO6937 Text			H		N

NOTES

1 It should be noted that the A/311 profile states: For routing all ADMDs should support all Form 1 Variants of O/R Name. All PRMDs should support at least Form 1, Variant 1 form of OR Name.

2 It should also be noted that the A/311 profile requires that all ADMDs should support the reception of Teletex body parts for delivery to their own UAEs.

3 An A/3211 implementation may generate MOTIS encoded information types. See 6.11.

4 Only Form 1 Variant 1 of O/Rname shown for TTC, but TTC defines other forms and variants. Form 1 Variant 1 recommended for PRMDs and ADMDs, Form 1 Variant 2 also recommended for ADMDs.

5 DDA's can be used to specify recipients in any Japanese domains other than TTC. Assignment of DDAs for UAs within TTC domains is not recommended.

6 One of [DeliveredInfo/NonDeliveredInfo] must be present. TTC encodes this as shown. Other profiles represent this by classifying both protocol elements as generatable. A similar situation exists with the P2 ORdescriptor.

7 TTC is expected to support IA5 for some international MHS communications.

Annex F (informative)

Interworking warnings

ADMD name is to be encoded as a single space when configurations with no ADMD's are present. It should be noted that this may change in January 1988 so that the ADMD name is encoded as a zero length element in such cases.

The OSI Implementors' agreements allow implementation to generate MPDUs with no body parts. Such MPDUs will be rejected by European-conformant systems. (Note this situation may change in January 1988)

In order to optimize the number of recipients you can read and reply to, it is advisable to be able to generate all standard O/R name attributes.