The drawings contained in this Recommendation have been done in Autocad.

12 Error recovery

This section specifies the error recovery mechanisms for errors which may occur during the communication between a TLM terminal and a TLMAU. The type of recovery will depend upon whether the error occurred during the transmission of control information or a normal document. The recovery mechanism of Recommendation T.62 shall apply. In addition, the qualifications given in the following paragraphs shall apply.

For both directions of transfer, if an error is detected during the transmission of a document containing control information, the transmitted part shall be discarded and the complete TAPDU retransmitted.

If an error is detected during the transmission of a document containing message body, the recovery mechanism depends upon the direction of transfer. In the TLMAU—to—TLM terminal direction, two possibilities for recovery are available, the first one is to apply the rules of Recommendation T.62 using CDC, the second one is the complete retransmission of the TAPDU. In the TLM terminal—to—TLMAU direction, three possible recovery mechanisms are identified, the choice of which is a national matter:

1) No recovery is provided. The complete TAPDU (control information and message body) must be retransmitted.

2) The TLMAU shall retain the received part up to the last acknowledged checkpoint. The originating TLM terminal is required to resume the sending of the remainder of the TAPDU by using CDC protocol element as described in Recommendation T.62. The TLMAU shall automatically link the retained part with the resumed part prior to sending to the MTS. If the originating TLM terminal does not resume the sending within a predetermined time or if the TLMAU receives new TAPDU within that time, the TLMAU shall discard the received part of the TAPDU.

3) The last acknowledged normal document is considered the end of the TAPDU. The TLMAU will send this part to the MTS as a complete message. If the TLM terminal wishes to resume transmission of the interrupted TAPDU, it must first resend the control information. There is no correlation between the original message and the continued message at a receipient TLMAU or IPM—UA.

Note 1 — For failures occurring between document boundaries, the last acknowledged document shall be regarded as the end of the current TAPDU.

Note 2 — A TAPDU, which may be composed of more than one document, cannot be considered complete before at least two documents were received.

Note 3 — If there is a relationship between the sequence of normal documents (for example a multi—part body message) being transferred, the recovery mechanism 2 should not be used.

13 Control procedures

13.1 *Session control procedure*

The session control procedures shall be in accordance with Recommendation T.62. However, the qualifications listed in the following subsections shall also apply.

13.1.1 *Session element of procedure*

The values of mandatory parameters used during session establishment shall be as given in Table 7/T.330. The format of the TLMAU terminal identifier shall be that specified in Recommendation F.200.

13.1.2 *Session rules*

Change control (CSCC/RSCCP exchange) may occur inside TAPDU boundaries but should occur outside document boundaries.

13.2 *Document control procedures*

The document control procedures shall be in accordance with Recommendation T.62.

Control information is conveyed in either a control document or a normal document. Where encoded information types of the control information is Teletex, the body of the IP—message is conveyed in a sequence (at least one) of normal documents.

Note — The use of the normal document for conveying control information should be restricted to Teletex access to IPMS.

TABLE 7/T.330

Mandatory parameter values during session establishment

T.62 parameters	Call Originator	
	TLM terminal	TLMAU
Terminal identifier of calling terminal (in CSS)	TLM TID	TLMAU's TID
Terminal identifier of called terminal (in RSSP)	TLMAU TID	TLM TID
Date and time	Date and time the TLM terminal originated call	Date and time the TLMAU terminal originated call
Service identifier	Telematic	Telematic
Window size	Note 1	Note 1
Session user data	Basic terminal capabilities and interchange format(s) which are available as receiving capabilities of the TLM terminal (Note 1)	Basic terminal capabilities and interchange format(s) which are available as receiving capabilities of the TLMAU (Note 1)

Note 1 — This parameter is mandatory for the Group 4 facsimile service.

Note 2 — TID means terminal identification.

Note 3 — Negotiation of the capabilities of handling control documents encoded in T.61 characters is not needed.

In a multiple—TAPDU session, the TAPDUs are delimited by at least one of the following methods:

1) implicity, that is, by the first document of the TAPDU being a control document;

2) explicitly, that is, by the first document of the TAPDU containing a parameter indicating the number of subsequence documents in the TAPDU.

Note — The second method is mandatory for multiple—action—unit session, where the first document of the TAPDU is a normal document.

When a single TAPDU is sent on a session connection, the TAPDU is delimited only by the end of the session.

13.3 *Log—on procedure*

It is conceivable that many subscribers will commonly use a single TLM terminal, some mechanism is required to identify a specific user by intermediate system, i.e. log—on procedure using user name and password. Detailed procedure requires further study.

ANNEX A

(to Recommendation T.330)

Formal definition of TLMA abstract service

This Annex is an integral part of this Recommendation.

This Annex, a supplement to § 7, formally defines for reference purposes the abstract service TLMA (telematic agent). It uses the PORT and ABSTRACT—BIND, —UNBIND, —OPERATION, and —ERROR macro of Recommendation X.407.

```
TLMAAbsService { ccitt—t330 }
DEFINITIONS IMPLICIT TAGS ::=
BEGIN

—— Prologue

EXPORTS

        —— Primary port types

                miscellanea

IMPORTS

        —— Abstract service

                origination,   reception,   management

                    FROM IPMSAbstractService { joint—iso—ccitt
                      mhs(6)   ipms(1)   modules(0)   abstract—service(3) }

        import,   export

                    FROM MTSAbstractService { joint—iso—ccitt
                      mhs(6)   mts(3)   modules(0)   mTS—abstract—service(3) }

        —— Information objects and their aspects

                IPM,   RN,   NRN

                    FROM IPMSInformationObjects { joint—iso—ccitt
                      mhs(6)   ipms(1)   modules(0)   information—objects(2) }

        —— Functional objects

                tlma

                    FROM IPMSFunctionalObjects { joint—iso—ccitt
                      mhs(6)   ipms(1)   modules(0)   functional—objects(1) }

        —— MTAS aspects

                MessageDeliveryEnvelope, ORName, MessageIdentifier, Priority, Time, ReportDeliveryEnvelope,
                PerRecipientReportDeliveryFields
                    FROM MTSAbstractService { joint—iso—ccitt
                      mhs(6)   mts(3)   modules(0)   mTS—abstract—service(3) }

        —— Abstract services macros

                OBJECT,      PORT,      ABSTRACT—BIND,      ABSTRACT—ERROR,      ABSTRACT—
OPERATION

                    FROM AbstractServiceNotation { joint—iso—ccitt
                      mhs(6)   adsdc(2)   modules(0)   notation(1) }

DateandTime ::= Time

—— Object

tlma                            OBJECT
```

```
                    PORTS {
                            origination     [S],
                            reception       [S],
                            management      [S],
                            miscellanea     [S],
                            import          [C],
                            export          [C] }
                    ::= id—ot—tlma
```

—— *Ports*

```
miscellanea PORT
        SUPPLIER PERFORMS {
                ChangeSubscriptionProfile,
                DSList,
                DSDelete,
                DSFetch,
                MessageStatus }
        ::= id—pt—miscellanea
```

TLMAAbsService (*continuation*)

—— *Miscellanea port specific abstract error problems*

```
SubscriptionProfileProblem ::= ENUMERATED {
        not—changed(0) }

DSProblem ::= ENUMERATED {
        no—message—in—ds(0),
        ds—not—supported(1),
        ds—not—subscribed(2),
        retrieval—identifier—invalid(3),
        parameter—invalid(4) }

MessageStatusProblem ::= ENUMERATED {
        query—identifier—invalid(0) }
```

—— *RetrievalIdentifier*

```
RetrievalIdentifier ::= PrintableString
```

—— *CallIdentification*

```
CallIdentification ::= SEQUENCE {
        TLMAU TID                       [0] TerminalIdentifier,
        TLM TID                         [1] TerminalIdentifier,
        date—and—time                           [2] DateAndTime,
        document—ref—num                [3] DocumentReferenceNumber,
        additional—session—ref—num      [4] AdditionalSessionReferenceNumber OPTIONAL }

TerminalIdentifier ::= PrintableString —— defined in Recommendation F.200 ——

DocumentReferenceNumber ::= NumericString

AdditionalSessionReferenceNumber ::= NumericString
```

—— *Change subscription profile abstract operation component types*

```
DSMode ::= SET {
                        [0] Mode,
        frequency       [1] Frequency OPTIONAL,
        time            [2] DateandTime OPTIONAL }

Mode ::= CHOICE {

        [0] auto—output,
        [1] retrieval }

Frequency ::= NumericString     —— in minutes ——
```

```
ErrorRecoveryMode ::= CHOICE {  —— see § 12 in this Recommendation ——
        [0] recovery—1,         —— no—recovery ——
        [1] recovery—2,         —— full—support—of—T.62—recovery—procedures ——
        [2] recovery—3          —— recovery—of—document—basis —— }

MessageDeleteMode ::= CHOICE {

        [0] auto—delete,
        [1] manual—delete }

—— DS List abstract operation component types

ListReport ::= SET {
        retrieval—id            [0] RetrievalIdentifier,
        message—type            [1] MessageType,
        priority                [2] Priority OPTIONAL,
        message—length                  [3] MessageLength OPTIONAL,
        originator—name         [4] ORName OPTIONAL }

MessageType ::= CHOICE {
        [0] ipm,
        [1] receipt—notice,
        [2] non—receipt—notice,
        [3] report }

MessageLength ::= NumericString         —— in octet ——

TLMAAbsService (continuation)

—— DS Fetch abstract operation component types

DeleteAfterOutput ::= CHOICE {

        [0] delete,

        [1] keep }

MessageReport ::= CHOICE {
        ipm—fetch               [0] IPMFetch,
        rn—fetch                [1] RNFetch,
        nrn—fetch                       [2] NRNFetch,
        report—fetch            [3] ReportFetch }

IPMFetch ::= SEQUENCE {
        envelope        [0] MessageDeliveryEnvelope,
        content         [1] IPM }

RNFetch ::= SEQUENCE {
            envelope    [0] MessageDeliveryEnvelope,
            —— not used intended—recipient—name and other recipient—name of deliver—envelope in RNFetch
——

        content         [1] RN } }

NRNFetch ::= SEQUENCE {

        [0] number—of—docs      [0] NumberOfAssociatedDocuments OPTIONAL,
        [1] SET {
            envelope            [1] MessageDeliveryEnvelope,
            —— not used intended—recipient—name and other recipient—name of deliver—envelope in NRNFetch
——

        content         [2] NRN } }

ReportFetch ::= SEQUENCE {
        [0] SET {
            number—of—docs  [0] NumberOfAssociatedDocuments OPTIONAL,
            call—id         [1] CallIdentification },
        [1] SET {
```

```
                envelope            [0] ReportDeliveryEnvelope,
                returned—content    [1] IPM OPTIONAL } } }

NumberOfAssociatedDocuments ::= NumericString

—— Message status abstract service

QueryIdentifier ::= CHOICE {
                submission—id       [0] MessageIdentifier,

                call—id             [1] CallIdentification }

StatusInfo ::= SET {

                                    [0] Status,

                per—recipient—info          [1] PerRecipientReportDeliveryFields OPTIONAL }

Status ::= CHOICE {
        [0] in—process,
        [1] delivered,
        [2] non—delivered }

—— Miscellanea abstract—operations

ChangeSubscriptionProfile ::= ABSTRACT—OPERATION

        ARGUMENT SET {
                                    ds—mode                 [0] DSMode OPTIONAL,
                                    error—recovery—mode     [1] ErrorRecoveryMode OPTIONAL,

                                    message—delete—mode     [2] MessageDeleteMode OPTIONAL }
RESULT { }
ERRORS {
        name—error,
        ds—error,
        subscription—profile—error }

TLMAAbsService (continuation)

DSList      ::= ABSTRACT—OPERATION

        ARGUMENT { }
        RESULT SET {
                        list—reports [0] SET OF ListReport OPTIONAL }
        ERRORS {
                subscription—error,
                name—error,
                ds—error }

DSDelete    ::= ABSTRACT—OPERATION

        ARGUMENT SET {
                selector        [0] SET OF RetrievalIdentifier }

        RESULT
        ERRORS {

                        subscription—error,
                        name—error,
                        ds—error }

DSFetch     ::= ABSTRACT—OPERATION

        ARGUMENT SET {

                                    selector                [0] SET OF RetrievalIdentifier,
                                    delete—after—output     [1] DeleteAfterOutput OPTIONAL }

        RESULT SET {
```

```
                    retrieval—id            [0] RetrievalIdentifier,
                    message—report          [1] MessageReport }

        ERRORS {
                    subscription—error,
                    name—error,
                    ds—error }

MessageStatus ::= ABSTRACT—OPERATION

        ARGUMENT SET {
                                            [0] QueryIdentifier }

        RESULT SET {
                    report—time             [0] DateandTime,
                    reported—message—id     [1] MessageIdentifier,
                    reported—message—id     [2] SET OF StatusInfo }

        ERRORS {
                    subscription—error,
                    name—error,
                    message—status—error }
```

—— *Miscellanea port specific abstract errors*

```
subscription—profile—error    ABSTRACT—ERROR

        PARAMETER SET {
                    problem [0] SubscriptionProfileProblem }

        ::= 0

ds—error    ABSTRACT—ERROR

                    problem [0] DSProblem }

        ::= 1

message—status—error    ABSTRACT—ERROR

        PARAMETER SET {
                    problem [0] MessageStatusProblem }

        ::= 2

END —— of TLMAAbsService
```

ANNEX B

(to Recommendation T.330)

Format of TAPDU components

In this Annex the formats of the control information for different examples TAPDUS are shown. The principles of encoding are given in § 11. The formats of elements as defined in § 11 are illustrated by the use of the following four different syntax elements:

1) The elements number field is represented by a sequence of numeric graphic characters.
2) The element name field is represented by a text string giving the CCITT language reference name of the field. The actual value shall be a language—dependent representation of that reference name.
3) Separators are shown as they shall be represented in the actual control information.
4) Element value fields are shown in square brackets ("[" and "]"). The actual parameter values are described in § 11.

Note — The examples shown below are not exhaustive. Not all types (element value of the TAPDU description in § 10 (ASN.1 description) are included.

B.1 *Conventions*

Following symbols are used in this Annex:

1) NL new line function such as (CR LF or LF CR) or LF.
2) space character.
3) [ ] element value field. Actual values are defined in § 11 of this Recommendation.
4) [ ] iteration.

B.2 *Example of TAPDU*

B.2.1 *Send—TAPDU*

3.1: SEND:

62: QUANTITY—OF—DOCS:[number—of—docs]

13: PRIORITY:[priority—ind]

19: FLAGS:
[deferred—delivery—time] `NL`
=[disclose—recipients],[alternate—recipient—allowed]

16: CONVERSION:[conversion—prohibited]

17: CONTENT—INFO:[content—return—request]

27: THIS—IPM:
[this—ipm—id]

20: FROM:
[originating—user]

21: AUTHORIZING:
[[authorizing—user] `NL`]

22: TO:
[[primary—recipient] `NL`
=[user—report—request],[rn—request],[nrn—request],[return—request],[reply—request] `NL`]

23: CC:
[[copy—recipient] `NL`
=[rn—request],[nrn—request],[return—request],[reply—request] `NL`]

24: BCC:
[[blind—copy—recipient] `NL`]

30: REPLIED—TO—IPM:

[replied—to—ipm—id]

29: OBSOLETES:
[[obsoleted—ipm—id] `NL`]

28: RELATED—IPMS:
[[related—ipm—id] `NL`]

26: SUBJECT:
[subject—content]

18: CONTENT—INDICATOR:
[expiry—time] `NL`
=[importance],[sensitivity]

25: REPLY:
[reply—time] `NL`
[[reply—recipient] `NL`]

31: BODY—TYPE: [[body—part],]

B.2.2 *SendAck—TAPDU*

3.14: SEND—ACK:

1: CORRELATION—INFORMATION:
[call—id]

65: SUBMISSION—ID:
[submission—msg—id]

33: SUBMISSION—TIME:
[submission—time]

B.2.3 *Exception—TAPDU*

3.12: EXCEPTION:

1: CORRELATION—INFORMATION:
[call—id]

9: ERRORS: [error—cause]

B.2.4 *Probe—TAPDU*

3.2: PROBE:

B.2.5 *ProbeAck—TAPDU*

3.15: PROBE—ACK:

1: CORRELATION—INFORMATION:
[call—id]

66: PROBE—ID:
[probe—msg—id]

33: SUBMISSION—TIME:
[submission—time]

B.2.6 *ExplicitRN—TAPDU*

3.6: EXPLICIT—RN:

15: RECIPIENTS:
[recipient—name]

13: PRIORITY:[priority—ind]

71: SUBJECT—IPM:
[subject—ipm—id]

69: IPN—ORIGINATOR:
[ipn—originating—user]

47: TIME—OF—RECEIPT:
[receipt—time]

44: CONVERTED—INFORMATION—TYPES:
[[eIT],]

B.2.7 *ExplicitRNAck—TAPDU*

3.16: EXPLICIT—RN—ACK:

1: CORRELATION—INFORMATION:
[call—id]

65: SUBMISSION—ID:
[submission—msg—id]

33: SUBMISSION—TIME:
[submission—time]

B.2.8 *Cancel—TAPDU*

3.13: CANCEL:

65: SUBMISSION—ID:
[submission—msg—id]

1: CORRELATION—INFORMATION:
[call—id]

B.2.9 *Deliver—TAPDU*

3.3: DELIVER:

62: QUANTITY—OF—DOCS:[number—of—docs]

13: PRIORITY:[priority—ind]

14: ORIGINATOR:
[originator—name]

39: THIS—RECIPIENT:
[this—recipient—name]

40: INTENDED—RECIPIENT:
[intended—recipient—name]

41: OTHER—RECIPIENTS:
[[other—recipient—name] `NL`]

33: SUBMISSION—TIME:
[submission—time]

35: DELIVERY—ID:
[delivery—msg—id]

42: CONVERSION—INDICATION:
[[eIT],] [conversion—prohibition]

44: CONVERTED—INFORMATION—TYPES:
[[eIT],]

27: THIS—IPM:
[this—ipm—id]

20: FROM:
[originating—user]

21: AUTHORIZING:
[[authorizing—user] `NL`]

22: TO:
[[primary—recipient] `NL`
=[rn—request],[nrn—request],[return—request],[reply—request] `NL`]

23: CC:
[[copy—recipient] `NL`
=[rn—request],[nrn—request],[return—request],[reply—request] `NL`]

24: BCC:
[[blind—copy—recipient] `NL`
=[rn—request],[nrn—request],[return—request],[reply—request] `NL`]

30: REPLIED—TO—IPM:
[replied—to—ipm—id]

29: OBSOLETES:
[[obsoleted—ipm—id] `NL`]

28: RELATED—IPMS:
[[related—ipm—id] `NL`]

26: SUBJECT:
[subject—content]

18: CONTENT—INDICATOR:
[expiry—time] `NL`
=[importance],[sensitivity],[autoforwarded]

25: REPLY:
[reply—time] `NL`
[[reply—recipient] `NL`]

67: EXTENSIONS:
[[extension—type] `NL`
[extension—value] `NL`]

31: BODY—TYPE:[body—part]

32: FORWARDED—INFO:
[forwarded—time] `NL`

//The delivery—envelope follows.

B.2.10 *ReceiptStatusNotice—TAPDU*

3.5: RECEIPT—STATUS—NOTICE:

62: QUANTITY—OF—DOCS:[number—of—docs]

13: PRIORITY:[priority—ind]

35: DELIVERY—ID:
[delivery—id]

14: ORIGINATOR:
[originator—name]

39: THIS—RECIPIENT:
[this—recipient—name]

33: SUBMISSION—TIME:
[submission—time]

4: TIME—OF—DELIVERY:
[delivery—time]

42: CONVERSION—INDICATION:
[[eIT],] [conversion—prohibition]

44: CONVERTED—INFORMATION—TYPES:
[[eIT],]

43: NOTIFICATION—TYPE:[report—type]

71: SUBJECT—IPM:
[subject—ipm—id]

69: IPN—ORIGINATOR:
[ipn—originating—user]

70: PREFERRED—RECIPIENT:
[preferred—recipient]

//The following three elements appear in case of RN.

47: TIME—OF—RECEIPT:
[receipt—time]

48: TYPE—OF—RECEIPT:
[type—of—receipt]

68: SUPPLEMENTARY—INFORMATION:
[suppl—receipt—info]

//The following three elements appear in case of NRN.

49: NON—RECEIPT—INFO:
[non—receipt—reason],[discard—reason]

50: COMMENTS:
[comments]

51: MESSAGE—RETURNED—INDICATION:

B.2.11 *DeliveryStatusNotice—TAPDU*

3.4: DELIVERY—STATUS—NOTICE:

62: QUANTITY—OF—DOCS:[number—of—docs]

1: CORRELATION—INFORMATION:
[call—id]

65: SUBMISSION—ID:
[submission—msg—id]

66: PROBE—ID:
[probe—msg—id]

[//repeated for each recipient

3: REPORTED—RECIPIENT:
[reported—recipient—name]

43: NOTIFICATION—TYPE:[report—type]

40: INTENDED—RECIPIENT:
[intended—recipient—name]

44: CONVERTED—INFORMATION—TYPES:
[[eIT],]

//The following three elements appear in case of DN.

4: TIME—OF—DELIVERY:
[delivery—time]

45: TYPE—OF—UA:
[type—of—ua]

68: SUPPLEMENTARY—INFORMATION:
[suppl—info]

//The following two elements appear in case of NDN.

46: NON—DELIVERY—REASON:
[reason—code],[diagnostic—code]

]

72: CONTENT—RETURNED—INDICATION:

B.2.12 *Register—TAPDU*

3.11: REGISTER:

73: EXPIRED—DISCARD:[discard—ipm]

74: OBSOLETED—DISCARD:[discard—ipm]

76: AUTO—FWD—IPMS:
[auto—fwd—ipms]

77: AUTO—FWD—RECIPIENTS:
[[auto—fwd—recipient],]

78: AUTO—FWD—HEADING:
[auto—fwd—heading]

79: AUTO—FWD—COMMENT:
[auto—fwd—comment]

58: DS—MODE:[ds—mode]

59: TLMAU—OPERATION:[error—recovery—mode],[auto—acknowledgement]

60: AUTO—OUTPUT:
[frequency] `NL`
[output—time]

81: MESSAGE—DELETE—MODE:[message—delete—mode]

//and other components require further study.

B.2.13 *RegisterAck—TPDU*

3.11: REGISTER—ACK:

B.2.14 *DSQuery—TAPDU*

3.7: DS—QUERY:

B.2.15 *DSReport—TAPDU*

3.8: DS—REPORT:

[//repeated for each message

38: RETRIEVAL—ID:
[retrieval—id]

52: MESSAGE—TYPE:[message—type]

14: ORIGINATOR:
[originator—name]

13: PRIORITY:[priority—ind]

37: MESSAGE—LENGTH:[message—length]

]

B.2.16 *MessageDelete—TAPDU*

3.18: MESSAGE—DELETE:

82: MESSAGE—SELECTOR:
[[retrieval—id] `NL`]

B.2.17 *OutputRequest—TAPDU*

3.9: OUTPUT—REQUEST:

[//repeated for each retrieval identifier

38: RETRIEVAL—ID:
[retrieval—id]

80: DELETE—AFTER—OUTPUT:[delete—after—output]

]

B.2.18 *OutputMessage—TAPDU*

3.10: OUTPUT—MESSAGE:

[//repeated for each message

62: QUANTITY—OF—DOCS:[number—of—docs]

38: RETRIEVAL—ID:
[retrieval—id]

52: MESSAGE—TYPE:[message—type]

4: TIME—OF—DELIVERY:
[delivery—time]

]

The remaining Components of this TAPDU are identical to the components in the Delivery. DeliveryStatusNotice and ReceiptStatusNotice—TAPDU. The actual components to be used depend upon the Message Type parameter value specified in the Message Type component.

B.2.19 *StatusQuery—TAPDU*

3.19: STATUS—QUERY:

14: ORIGINATOR:
[originator—name]

65: SUBMISSION—ID:
[submission—msg—id]

1: CORRELATION—INFORMATION:
[call—id]

B.2.20 *StatusReport—TAPDU*

3.20: STATUS—REPORT:

1: CORRELATION—INFORMATION:

[call—id]

84: TIME—OF—REPORT:

[report—time]

75: REPORTED—MESSAGE—ID:

[reported—message—id]

[//repeated for each message

83: MESSAGE—STATUS:

[status]

4: TIME—OF—DELIVERY:

[delivery—time]

45: TYPE—OF—UA:

[type—of—ua]

46: NON—DELIVERY—REASON:

[reason—code],[diagnostic—code]

]

ANNEX C

(to Recommendation T.330)

Element ID list

This Annex provides a table which maps the coding of TAPDU and the corresponding element ID. The element ID list Table C—1/T.330 comprises:

1) The first column contains the element number and element name, as defined in § 11 of this Recommendation.

2) The second column contains the TAPDUId or ElementId name defined in § 10 of this Recommendation.

3) The third column contains the type of the TAPDU element, defined in § 10 of this Recommendation.

4) The fourth column contains the Remarks.

TABLE C—1/T.330

Element ID list

TAPDUId/element ID	Name	Type	Remarks
3.1:SEND:	send	—	
3.2:PROBE:	probe	—	
3.3:DELIVER:	deliver	—	
3.4:DELIVERY—STATUS—NOTICE:	deliveryStatusNotice	—	
3.5:RECEIPT—STATUS—NOTICE:	receiptStatusNotice	—	
3.6:EXPLICIT—RN:	explicitRN	—	
3.7:DS—QUERY:	dsQuery	—	
3.8:DS—REPORT:	dsReport	—	
3.9:OUTPUT—REQUEST:	outputRequest	—	
3.10:OUTPUT—MESSAGE:	outputMessage	—	
3.11:REGISTER:	register	—	
3.12:EXCEPTION:	exception	—	
3.13:CANCEL:	cancel	—	
3.14:SEND—ACK:	sendAck	—	
3.15:PROBE—ACK:	probeAck	—	
3.16:EXPLICIT—RN—ACK:	explicitRNAck	—	
3.17:REGISTER—ACK:	registerAck	—	
3.18:MESSAGE—DELETE:	messageDelete	—	
3.19:STATUS—QUERY:	statusQuery	—	
3.20:STATUS—REPORT:	statusReport	—	
1:CORRELATION—INFORMATION:	correlationInfo	Primitive	
2:			Reserved for CF
3:REPORTED—RECIPIENT:	reportedRecipient	Primitive	
4:TIME—OF—DELIVERY:	timeOfDelivery	Primitive	
5:	Reserved for CF		
6:			Reserved for CF
7:			Reserved for CF
8:			Reserved for CF
9:ERRORS:	errors	Primitive	
10:			Reserved for CF
11:			Reserved for CF
12:RECEIVED—TID:			Reserved for CF
13:PRIORITY:	priority	Primitive	
14:ORIGINATOR:	originator	Constructor	
15:RECIPIENTS:	recipients	Constructor	
16:CONVERSION:	conversion	Primitive	
17:CONTENT—INFO:	contentInfo	Constructor	
18:CONTENT—INDICATOR:	contentIndicator	Constructor	
19:FLAG:	per Message Indicator	Constructor	

TABLE C—1/T.330 *(cont.)*

TAPDUId/element ID	Name	Type	Remarks
20:FROM:	from	Primitive	
21:AUTHORIZING:	authorizing	Constructor	
22:TO:	to	Constructor	
23:CC:	cc	Constructor	
24:BCC:	bcc	Constructor	
25:REPLY:	reply	Constructor	
26:SUBJECT:	subject	Primitive	
27:THIS—IPM:	thisIPM	Primitive	
28:RELATED—IPMS:	relatedIPMS	Constructor	
29:OBSOLETES:	obsoletedIPMS	Constructor	
30:REPLIED—TO—IPM:	repliedToIPM	Primitive	
31:BODY—TYPE:	bodyType	Constructor	
32:FORWARDED—INFO:	forwardedInfo	Constructeur	For further study
33:SUBMISSION—TIME:	submissionTime	Primitive	
34:LATEST—DELIVERY:	date and time	Primitive	
35:DELIVERY—ID:	deliveryId	Primitive	
36:RETURN—ADDRESS:	return address	Primitive	
37:MESSAGE—LENGTH:	messageLength	Primitive	
38:RETRIEVAL—ID:	retrievalId	Primitive	
39:THIS—RECIPIENT:	thisRecipient	Primitive	
40:INTENDED—RECIPIENT:	intendedRecipient	Primitive	
41:OTHER—RECIPIENTS:	otherRecipients	Constructor	
42:CONVERSION—INDICATION:	conversionIndication	Constructor	
43:NOTIFICATION—TYPE:	notificationType	Primitive	
44:CONVERTED—INFORMATION—TYPES:	convertedInfoTypes	Primitive	
45:TYPE—OF—UA:	typeOfUA	Primitive	
46:NON—DELIVERY—REASON:	nonDeliveryReason	Primitive	
47:TIME—OF—RECEIPT:	timeOfReceipt	Primitive	
48:TYPE—OF—RECEIPT:	typeOfReceipt	Primitive	
49:NON—RECEIPT—INFO:	nonReceiptInfo	Primitive	
50:COMMENT:	comment	Primitive	
51:MESSAGE—RETURNED—INDICATION:	messageReturnedInd	—	
52:MESSAGE—TYPE:	messageType	Primitive	
53:LANGUAGE:	languageInd	Primitive	
54:REDIRECTED—FROM:	redirected from	Constructor	

TABLE C—1/T.330 *(end)*

TAPDUId/element ID	Name	Type	Remarks
55:	—	—	Not used
56:	—	—	Not used
57:	—	—	Not used
58:DS—MODE:	dsMode	Primitive	
59:TLMAU—OPERATION:	tLMAUOperation	Constructor	
60:AUTO—OUTPUT:	autoOutput	Constructor	
61:	—	—	Not used
62:QUANTITY—OF—DOCS:	quantityOfDocs	Primitive	
63:	Not assigned		
64:	Not assigned		
65:SUBMISSION—ID:	submissionId	Primitive	
66:PROBE—ID:	probeId	Primitive	
67:MSG—INCOMPLETE:			
68:SUPPLEMENTARY—INFORMATION:	supplReceiptInfo		
SupplInfo	Primitive		
69:IPN—ORIGINATOR:	IPNOriginator	Primitive	
70:PREFERRED—RECIPIENT:	preferredRecipient	Primitive	
71:SUBJECT—IPM:	subjectIPM	Primitive	
72:CONTENT—RETURNED—INDICATION:	contentReturned	—	
73:EXPIRED—DISCARD:	expiredDiscard	Primitive	
74:OBSOLETED—DISCARD:	obsoletedDiscard	Primitive	
75:REPORTED—MESSAGE—ID:	reportedMessageId	Primitive	
76:AUTO—FWD—IPMS:	autoFWDIPMs	Primitive	
77:AUTO—FWD—RECIPIENTS:	autoFWDRecipient	Primitive	
78:AUTO—FWD—HEADING:	autoFWDHeading	Constructor	For further study
79:AUTO—FWD—COMMENT:	autoFWDComment	Primitive	
80:DELETE—AFTER—OUTPUT:	deleteAfterOutput	Primitive	
81:MESSAGE—DELETE—MODE:	messageDeleteMode	Primitive	
82:MESSAGE—SELECTOR:	messageSelector	Constructor	
83:MESSAGE—STATUS:	messageStatus	Primitive	
84:TIME—OF—REPORT:	timeOfReport	Primitive	

Conventions:

1) Primitive: element contains only one component

2) Constructor: element contains more than one component

3) : space character

ANNEX D

(to Recommendation T.330)

Element of service for TTX/IPM service intercommunication

This PTXAU provides only the services listed in Table D—1/T.330 to telematic users. These services may be implemented using only the mhs—doc—xfer abstract operations: message send, message delivery, receipt status notice and delivery status notice.

When a non—registered user attempts to access a service not within the PTTXAU set of services, then a subscription error will be returned.

TABLE D—1/T.330

Element of service

Reference Rec. F.400 Annex B	F.400 Elements of service	Message submission from TTX to PTTXAU	Message delivery to TTX from PTTXAU	Information generated by PTTXAU
B.5	Authorizing users indication		X	
B.6	Auto—forwarded indication		X	
B.8	Blind copy recipient indication		X	
B.9	Body part encryption indication		X	
B.12	Content type indication		X	X
B.13	Conversion prohibition	X	X	
B.15	Converted indication		X	
B.18	Cross—referencing indication		X	
B.21	Delivery notification	X	NA	X
B.22	Delivery time stamp indication		X	X
B.25	Disclosure of other recipients		X	
B.26	DL expansion history indication		X	X
B.29	Expiry date indication		X	
B.31	Forwarded IP—message indication		X	
B.32	Grade of delivery selection	X	X	
B.34	Implicit conversion		X	
B.35	Importance indication		X	
B.37	IP—message identification		X	
B.38	Language indication		X	
B.39	Latest delivery designation		X	
B.41	Message indentification		X	
B.45	Multi—destination delivery	X	NA	X
B.46	Multi—part body		X	
B.47	Non—delivery notification		NA	
B.48	Non—receipt notification request indication	X	NA	
B.52	Obsoleting indication		X	
B.54	Original encoded information types indication		X	X
B.55	Originator indication		X	
B.56	Originator requested alternate recipient	X	X	
B.62	Primary and copy recipients indication		X	
B.72	Reply request indication		X	
B.73	Replying IP—message indication		X	
B.80	Sensitivity indication		X	
B.88	Subject indication	X	X	
B.89	Submission time stamp indication		X	

NA No applicable

X Applicable