The drawaing contained in this Recommendation have been done in Autocad

TABLE 2/T.330

Relationship between abstract operation and TAPDU

mhs—doc—xfer Abstract operation	TAPDU		Direction of transfer TLM TLMAU
Operation	TAPDU name	TLMAU status	
MessageSend	(O) Send—TAPDU (R) SendAck—TAPDU (E) Exception—TAPDU	M C M	->\ <- <-
MessageProbe	(O) Probe—TAPDU (R) ProbeAck—TAPDU (E) Exception—TAPDU	C C C	->\ <- <-
ExplicitReceive	(O) ExplicitRN—TAPDU (R) ExplicitRNAck—TAPDU (E) Exception—TAPDU	C C C	->\ <- <-
MessageCancel	(O) Cancel—TAPDU (R) — (E) Exception—TAPDU	C — C	->\ <-
MessageDeliver	(O) Deliver—TAPDU	M	<-
ReceiptStatusNotice	(O) ReceiptStatusNotice—TAPDU	M	<-
DeliveryStatusNotice	(O) DeliveryStatusNotice—TAPDU	M	<-
Register	(O) Register—TAPDU (R) RegisterAck—TAPDU (E) Exception—TAPDU	C C C	->\ <- <-
DSList	(O) DSQuery—TAPDU (R) DSReport—TAPDU (E) Exception—TAPDU	C C C	->\ <-
DSDelete	(O) MessageDelete—TAPDU (R) — (E) Exception—TAPDU	C — C	->\ <- <-
DSFetch	(O) OutputRequest—TAPDU (R) OutputMessage—TAPDU (E) Exception—TAPDU	C (remarque 1) C (remarque 1) C (remarque 1)	->\ <- <-
MessageStatus	(O) StatusQuery—TAPDU (R) StatusReport—TAPDU (E) Exception—TAPDU	C C C	->\ <- <-

O Argument R Result E Error

M Mandatory C Conditional

Note 1 — In cases where TLMAU provides DS, these TAPDU are mandatory.

Note 2 — A message may arrive at a TLM terminal as a result of either a Deliver—TAPDU or OutputMessage—TAPDU. The Deliver—TAPDU is applicable when delivery occurs directly to a TLM erminal. The OutputMessage—TAPDU is only applicable in the case that DS is subscribed.

10.1.1.1 *Send—TAPDU*

The Send—TAPDU comprises following elements:

Send—TAPDU

```
Send—TAPDU ::= SEQUENCE {
        [0] SEQUENCE {
            [0] SendTAPDUId,
            [1] SEQUENCE {
                quantityOfDocs      QuantityOfDocsElementId,
                number—of—docs      NumberOfAssociatedDocuments } OPTIONAL },

        —— See Note 1
Send [1] SET {
            [0] SEQUENCE {
                priority                PriorityElementId,
                    priority—ind        PriorityValue DAFAULT normal } OPTIONAL,
            [1] SEQUENCE {
                perMessageIndicators    PerMessageIndicatorsElementId,
                                SEQUENCE {
                deferred—delivery—time [0] DateandTime OPTIONAL,
                                                [1] SET {
                disclose—recipients                 [0] DiscloseRecipientsValue OPTIONAL,
                alternate—recipient—allowed         [1] AlternateRecipientAllowedValue OPTIONAL,
                recipient—reassignment—prohibited [2] ReassignmentValue OPTIONAL } } } OPTIONAL,
            [2] SEQUENCE {
                conversion              ConversionElementId,
                conversion—info             ConversionInfoValue } OPTIONAL,
            [3] SEQUENCE {
                contentinfo             ContentInfoElementId,
                content—return—request      ContentReturnRequestValue } OPTIONAL,
            [4] SEQUENCE {
                returnAddress           ReturnAddressElementId,
                    postal—address              PostalAddressValue OPTIONAL,
            [5] SEQUENCE {
                latestDelivery          LatestDeliveryElementId,
                    latest—delivery—time  DateandTime } OPTIONAL },
            [6] SEQUENCE {
                to                      ToElementId,
                            SET OF SEQUENCE {
                    primary—recipient   [0] ORDescriptor,
                                        [1] RecOptions } } OPTIONAL,
—— See Note 2
            [7] SEQUENCE {
                cc                      CCElementId,
                            SET OF SEQUENCE {
                    copy—recipient      [0] ORDescriptor,
                                        [1] RecOptions } } OPTIONAL,
—— See Note 2
            [8] SEQUENCE {
                bcc                     BCCElementId,
                            SET OF SEQUENCE {
                    blind—copy—recipient [0] ORDescriptor,
                                        [1] RecOptions } } OPTIONAL },
—— See Note 2
```

Send—TAPDU *(continued)*

```
—— Send—TAPDU Definition (continued)
        [2] SET {

            [0] SEQUENCE {
                thisIPM                     ThisIPMElementId,
                this—ipm—id                     IPMIdentifier } OPTIONAL,
—— See Note 3

            [1] SEQUENCE {
                from                        FromElementId,
                    originating—user                ORDescriptor } OPTIONAL,
            [2] SEQUENCE {
                authorizing                 AuthorizingElementId,
                                SET OF
                    authorizing—user            OrDescriptor } OPTIONAL,
            [3] SEQUENCE {
                repliedToIPM                RepliedToIPMElementId,
                    replied—to—ipm—id           IPMIdentifier } OPTIONAL,
            [4] SEQUENCE {
                obsoletedIPMs               ObsoletedIPMsElementId,
                                SEQUENCE OF
                    obsoleted—ipm—id        IPMIdentifier } OPTIONAL,
            [5] SEQUENCE {
                relatedIPMs
                                            RelatedIPMsElementId,
                                SEQUENCE OF
                    related—ipm—id              IPMIdentifier } OPTIONAL,
            [6] SEQUENCE {
                subject                     SubjectElementId,
                    subject—content             SubjectContent } OPTIONAL,
            [7] SEQUENCE {
                contentIndicator            ContentIndicatorElementId,
                                SEQUENCE {
                    expiry—time     [0] DateandTime OPTIONAL,
                                    [1] SET {
                    importance      [0] ImportanceValue DEFAULT normal,
                    sensitivity     [1] SensitivityValue OPTIONAL } } } OPTIONAL,
            [8] SEQUENCE {
                reply
                                            ReplyElementId,
                                SEQUENCE {
                    reply—time
                                        [0] DateandTime,
                                        [1] SET OF {
                    reply—recipient             ORDescriptor } } OPTIONAL,
            [9] SEQUENCE {
                language                    LanguageElementId,
                    language—ind                LanguagueInd } OPTIONAL,
MsgIncomplete [10] MsgIncompleteElementId OPTIONAL },
            —— Body
        [3] SET {
            [0] SEQUENCE {
                BodyType                    BodyTypeElementId,
                        SET OF {

                    Body—part               BodyPartValue } OPTIONAL } }
```

Send—TAPDU *(end)*

```
—— Send—TAPDU Definition (continued)

—— Definition of RecOptions

    RecOptions ::= SET {
            user—report—request                          [1]  UserReportRequestValue
                                                              OPTIONAL,
            explicit—conversion                          [2]  ExplicitConversionValue
                                                              OPTIONAL OPTIONA
            rn—request                                   [3]  RNRequestValue OPTIONAL,
            nrn—request                                  [4]  NRNRequestValue OPTIONAL,
            return—request                               [5]  ReturnRequestValue OPTIONAL,
            reply—request                                [6]  ReplyRequestValue DEFAULT
                                                              noReply,
            requested—delivery—method                    [7]  RequestedDelValue OPTIONAL,
            terminal—type                                [8]  TerminalTypeValue OPTIONAL,
            physical—forwarding—prohibited               [9]  PhyForProhibValue OPTIONAL,
            physical—forwarding—address—request          [10] PhyForAdrValue OPTIONAL,
            physical—delivery—modes                      [11] PhyDelModValue OPTIONAL,
            registered—mail—type                         [12] RegMailType Value OPTIONAL,
            recipient—number—for—advice                  [13] RecNumAdvValue OPTIONAL,
            physical—delivery—report—request             [14] PhyDelRepValue OPTIONAL,
            originator—requested—alternate—recipient     [15] OrgRecAltValue OPTIONAL,
```

Note 1 — This element must be present when ControlInfo is conveyed by a normal document and more than one TAPDU are conveyed in this session.

Note 2 — OR Descriptor must contain an ORAddress and at least one of these addresses must be present.

Note 3 — When this element is omitted, the TLMAU shall construct this component which consists of the following components: originator name, date and time, and if necessary, a sequence number.

10.1.1.2 *SendAck—TAPDU*

The SendAck—TAPDU comprises following elements:

SendAck—TAPDU

```
SendAck—TAPDU ::= SEQUENCE {
    [0] SEQUENCE {
            sendAck [0] SendAckTAPDUId,
                [1] SEQUENCE {
                    correlationInfo          CorrelationInfoElementId,
                        call—id                  CallIdentification } },
    —— See Note
    [1] SET {
            [0] SEQUENCE {
                submissionId                 SubmissionIdElementId,
                    submission—msg—id            MessageIdentifier },
            [1] SEQUENCE {
                submissionTime               SubmissionTimeElementId,
                    submission—time              DateandTime } } }
```

Note — This element is a session connection information that identifies previous Send—TAPDU being reported on.

10.1.1.3 *Exception—TAPDU*

The Exception—TAPDU comprises following elements:

Exception—TAPDU

```
Exception—TAPDU ::= SEQUENCE {
    [0] SEQUENCE {
        exception [0] ExceptionTAPDUId,
                  [1] SEQUENCE {
                      correlationInfo          CorrelationInfoElementId,
                          call—id                  CallIdentification },
    —— See Note
                  [2] SEQUENCE {
                      errors                   ErrorsElementId,
                          error—cause              ErrorCauseValue } } }
```

Note — This element is a session connection information that identifies associated TAPDU being reported on e.g. Send—TAPDU.

10.1.2 *MessageProbe*

A TLM terminal sends a Probe—TAPDU to invoke the MessageProbe abstract operation. The TLMAU returns a ProbeAck—TAPDU to report the result of that operation, or may return an Exception—TAPDU (§ 10.1.1.3) to report an abstract error.

10.1.2.1 *Probe—TAPDU*

The Probe—TAPDU comprises following elements:

Probe—TAPDU

```
Probe—TAPDU ::= SEQUENCE {
          [0] SEQUENCE {
              [0] ProbeTAPDUId,
              [1] SEQUENCE {
                  quantityOfDocs           QuantityOfDocsElementId,
                      number—of—docs               NumberOfAssociatedDocuments } OPTIONAL },

          [1] SET {
              —— Continuation see send—TAPDU.
              —— Note that only few elements of the send—TAPDU are relevant for a Probe—TAPDU.
              —— Not relevant elements will be ignored.
              —— At least one recipient must be present.
```

10.1.2.2 *ProbeAck—TAPDU*

The ProbeAck—TAPDU comprises following elements:

ProbeAck—TAPDU

```
ProbeAck—TAPDU ::= SEQUENCE {
        [0] SEQUENCE {
            probeAck [0] ProbeAckTAPDUId,
                     [1] SEQUENCE {
                         correlationInfo          CorrelationInfoElementId,
                             call—id                  CallIdentification } },

        [1] SET {
                [0] SEQUENCE {
                         probeId                  ProbeElementId,
                             probe—msg—id             MessageIdentifier },
                [1] SEQUENCE {
                         submissionTime           SubmissionTimeElementId,
                             submission—time          DateandTime } } }
```

10.1.3 *ExplicitReceive*

A TLM terminal sends an ExplicitRN—TAPDU to invoke the ExplicitReceive abstract operation. The TLMAU returns an ExplicitRNAck—TAPDU to report the result of that operation, or may return an Exception—TAPDU (see § 10.1.1.3), to report an abstract error.

10.1.3.1 *ExplicitRN—TAPDU*

The ExplicitRN—TAPDU comprises following elements:

ExplicitRN—TAPDU

```
ExplicitRN—TAPDU ::= SEQUENCE {
    [0]

                explicitRN                   ExplicitRNTAPDUId,
    [1] SET {
                [0] SEQUENCE {
                    recipients               RecipientsElementId,
                        recipient—name           ORName },
                [1] SEQUENCE {
                    priority                 PriorityElementId,
                        priority—ind             PriorityValue DEFAULT normal } OPTIONAL,
                [2] SEQUENCE {
                    subjectIPM               SubjectIPMElementId,
                        subject—ipm—id               IPMIdentifier } OPTIONAL,
                [3] SEQUENCE {
                    IPNOriginator            IPNOriginatorElementId,
                        ipn—originating—user     ORDescriptor } OPTIONAL,
                [4] SEQUENCE {
                    timeOfReceipt            TimeOfReceiptElementId,
                        receipt—time             DateandTime } OPTIONAL,
                [5] SEQUENCE {
                    convertedInfoTypes       ConvertedInfoTypesElementId,
                                             SET OF
                    eIT                          EITValue } OPTIONAL } }
```

Note — If receipt—time element defined in Receipt is omitted, TLMAU extracts one from the CES of the session in which this TAPDU was transferred. This may differ from the time of actual receipt of IPM.

10.1.3.2 *ExplicitRNAck—TAPDU*

The ExplicitRNAck—TAPDU comprises following elements:

ExplicitRNAck—TAPDU

```
ExplicitRN—TAPDU ::= SEQUENCE {
    [0] SEQUENCE {
        explicitRNAck [0] ExplicitRNTAPDUId,
                      [1] SEQUENCE {
                          correlationInfo          CorrelationInfoElementId,
                              call—id                  CallIdentification } },

    [1] SET {

                  [0] SEQUENCE {
                          submissionId             SubmissionElementId,
                              submission—msg—id        MessageIdentifier },
                  [1] SEQUENCE {
                          submissionTime           SubmissionTimeElementId,
                              submission—time          DateandTime } } }
```

10.1.4 *MessageCancel*

A TLM terminal sends a Cancel—TAPDU to invoke the MessageCancel abstract operation. The TLMAU returns no TAPDU to report the result of that operation, or may return an Exception—TAPDU (see § 10.1.1.3), to report an abstract error.

10.1.4.1 *Cancel—TAPDU*

The Cancel—TAPDU comprises following elements:

Cancel—TAPDU

```
Cancel—TAPDU ::= SEQUENCE {
          cancel [0] CancelTAPDUId,
                     [1] SEQUENCE {
                         submissionId             SubmissionIdElementId,
                           submission—msg—id      MessageIdentifier } OPTIONAL,
                     [2] SEQUENCE {
                         correlation—Info         Correlation—InfoElementId,
                           call—id                CallIdentification } OPTIONAL }

          —— one of these must be present.
```

10.1.5 *MessageDeliver*

A TLMAU sends a Deliver—TAPDU to invoke the MessageDeliver abstract operation.

10.1.5.1 *Deliver—TAPDU*

The Deliver—TAPDU comprises following elements:

Deliver—TAPDU

```
Deliver—TAPDU ::= SEQUENCE {
        [0] SEQUENCE {
            deliver [0] DeliverTAPDUId,
                    [1] SEQUENCE {
                        quantityOfDocs                  QuantityOfDocsElementId,
                            number—of—docs                  NumberOfAssociatedDocuments } OPTIONAL
},
            —— MTS parameters
        [1] SET {
                    [0] SEQUENCE {
                        priority                        PriorityElementId,
                            priority—ind                    PriorityValue DEFAULT normal } OPTIONAL,
                    [1] SEQUENCE {
                        originator                      OriginatorElementId,
                            originator—name                 ORName } OPTIONAL,
                    [2] SEQUENCE {
                        thisRecipient                   ThisRecipientElementId,
                            this—recipient—name             ORName },
                    [3] SEQUENCE {
                        orgIntendedRecipient                OrgIntendedRecipientElementId,
                            org—intended—recipient—name         ORName } OPTIONAL,
                    [4] SEQUENCE {
                        otherRecipients                     OtherRecipientsElementId,
                                    SET OF
                            otherRecipient—name             ORName } OPTIONAL,
                    [5] SEQUENCE {
                        redirectedfrom                      RedirectedFromElementId,
                                    SEQUENCE OF
                            redirected—from                     ORName } OPTIONAL,
                    [6] SEQUENCE {
                        submissionTime                  SubmissionTimeElementId,
                            submission—time                 DateandTime },
                    [7] SEQUENCE {
                        deliveryId                      DeliveryElementId,
                            delivery—msg—id                 MessageIdentifier } OPTIONAL,
                    [8] SEQUENCE {
                        conversionIndication                ConversionIndicationElementId,
                                    SET {
                                        [0] SET OF
                            eIT                                 EITValue OPTIONAL } }
                        conversion-prohibited [1] ConversionProhibitedValue OPTIONAL } } OPTIONAL,
                    [9] SEQUENCE {
                        convertedInfoTypes                  ConvertedInfoTypesElementId,
                                    SET OF
                        eIT                                 EITValue } } ,
```

Deliver—TAPDU *(continued)*

```
—— Deliver—TAPDU Definition (continued)

    —— IPMS parameters

  [2] SET {

      [0] SEQUENCE {
          thisIPM                 ThisIPMElementId,
              this—ipm—id             IPMIdentifier },
      [1] SEQUENCE {
          from                    FromElementId,
              originating—user        OrDescriptor } OPTIONAL,
      [2] SEQUENCE {
          authorizing             AuthorizingElementId,
                      SET OF
              authorizing—user        ORDescriptor } OPTIONAL,
      [3] SEQUENCE {
          to                      ToElementId,
                      SET OF SEQUENCE {
              primary—recipient [0] ORDescriptor,
                                [1] NotificationSpecification } } OPTIONAL,
      [4] SEQUENCE {
          cc                      CCElementId,
                      SET OF SEQUENCE {
              copy—recipient [0] ORDescriptor,
                             [1] NotificationSpecification } } OPTIONAL,
      [5] SEQUENCE {
          bcc                     BCCElementId,
                      SET OF SEQUENCE {
              blind—copy—recipient [0] ORDescriptor,
                                   [1] NotificationSpecification } } OPTIONAL,
      [6] SEQUENCE {
              repliedToIPM        RepliedToIPMElementId,
                  replied—to—ipm—id   IPMIdentifier } OPTIONAL,
      [7] SEQUENCE {
              obsoletedIPMs       ObsoletedIPMsElementId,
                          SET OF
                  obsoleted—ipm—id    IPMIdentifier } OPTIONAL,
      [8] SEQUENCE {
              relatedIPMs         RelatedIPMsElementId,
                          SET OF
                  related—ipm—id      IPMIdentifier } OPTIONAL,
      [9] SEQUENCE {
              subject             SubjectElementId,
                  subject—content     SubjectContent } OPTIONAL,
      [10] SEQUENCE {
              contentIndicator    ContentIndicatorElementId,
                          SEQUENCE {
              expiry—time         [0] DateandTime OPTIONAL,
                                  [1] SET {
              importance          [0] ImportanceValue DEFAULT normal,
              sensitivity         [1] SensitivityValue OPTIONAL,
              auto—forwarded      [2] AutoForwardedValue DEFAULT
                                      notAutoForward } } } OPTIONAL,
```

Deliver—TAPDU *(end)*

```
—— Deliver—TAPDU Definition (continued)
        [11] SEQUENCE {
                reply
                                                ReplyElementId,
                                SEQUENCE {
                    reply—time  [0] DateandTime,
                                [1] SET OF
                    reply—recipient             ORDescriptor } } OPTIONAL,
        [12] SEQUENCE {
                language                        LanguageElementId,
                    language—ind                    LanguageInd } OPTIONAL,
MsgIncomplete [13] MsgIncompleteElementId OPTIONAL },
        —— Body
    [3] SEQUENCE {
        [0] SEQUENCE {
                bodyType                        BodyTypeElementId,
                    body—part                       BodyPartValue } OPTIONAL,
        [1] SEQUENCE {
            forwardedInfo                       ForwardedInfoElementId,
                                SEQUENCE {
                    forwarded—time [0] DateandTime,
                                [1] DeliveryEnvelope } } OPTIONAL,
  —— Delivery Envelope contains same set of MTS parameters of Deliver—TAPDU } }
  —— Definition of Notification Specification
    Notification Specification ::= SET {
                                        rn—request      [0] RNRequestValue OPTIONAL,
                                        nrn—request     [1] NRNRequestValue OPTIONAL,
                                        return—request  [2] ReturnRequestValue OPTIONAL,
                                        reply—request       [3] ReplyRequestValue DEFAULT
                                                            noReply }
```

10.1.6 *ReceiptStatusNotice*

A TLMAU terminal sends a ReceiptStatusNotice—TAPDU to invoke the ReceiptStatusNotice abstract operation.

10.1.6.1 *ReceiptStatusNotice—TAPDU*

The ReceiptStatuNotice—TAPDU comprises following elements:

ReceiptStatusNotice—TAPDU

```
ReceiptStatusNotice—TAPDU ::= SEQUENCE {
    [0] SEQUENCE {
        receiptStatusNotice [0] ReceiptStatusNoticeTAPDUId,
                            [1] SEQUENCE {
                quantityOfDocs                  QuantityOfDocsElementId,
                    number—of—docs                  NumberOfAssociatedDocuments } OPTIONAL },
        —— MTS parameters
    [1] SET {
                [0] SEQUENCE {
                    priority                    PriorityElementId,
                        priority—ind                PriorityValue },
                [1] SEQUENCE {
                    deliveryId                  DeliveryIdElementId,
                        delivery—id                 MessageIdentifier } OPTIONAL,
                [2] SEQUENCE {
                    originator                  OriginatorElementId,
                        originator—name                 ORName } OPTIONAL,
                [3] SEQUENCE {
                    thisRecipient               ThisRecipientElementId,
                        this—recipient—name         ORName },
                [4] SEQUENCE {
                    submissionTime              SubmissionTimeElementId,
                        submissionTime              DateandTime },
                [5] SEQUENCE {
                    timeOfDelivery              TimeOfDeliveryElementId,
                        delivery—time               DateandTime },
                [6] SEQUENCE {
                    conversionIndication        ConversionIndicationElementId,
                                SET {
                                                [0] SET OF
                        eIT                             EITValue } OPTIONAL } }
                    conversion—prohibited [1] ConversionProhibitedValue OPTIONAL } } OPTIONAL,
                [7] SEQUENCE {
                    convertedInfoTypes          ConvertedInfoTypesElementId,
                                SET OF
                    eIT                             EITValue } } ,
        —— IPMS parameters
    [2] SET {
                [0] SEQUENCE {
                    notificationType            NotificationTypeElementId,
                        report—type                 ReportTypeValue },
                [1] SEQUENCE {
                    subjectIPM                  SubjectIPMElementId,
                        subject—ipm—id                  IPMIdentifier },
                [2] SEQUENCE {
                    IPNOriginator               IPNOriginatorElementId,
                        ipn—originating—user            ORDescriptor } OPTIONAL,
                [3] SEQUENCE {
                    preferredRecipient          PreferredRecipientElementId,
                        preferred—recipient         ORDescriptor } OPTIONAL,
```

ReceiptStatusNotice—TAPDU *(end)*

```
—— ReceiptStatusNotice—TAPDU Definition (continued)
        [4] SET {
                [0] SEQUENCE {
            timeOfReceipt                   TimeOfReceiptElementId,
                receipt—time                    DateandTime },
                [1] SEQUENCE {
            typeOfReceipt                   TypeOfReceiptValue DEFAULT manual } OPTIONAL,
                [2] SEQUENCE {
            supplReceiptInfo                SupplReceiptInfoElementId,
                suppl—receipt—info              SuplementaryInformation     }     OPTIONAL     }
OPTIONAL,
        [5] SET {
                [0] SEQUENCE {
            nonReceiptInfo                  NonReceiptInfoElementId,
                    SET {
            non—receipt—reason [0] NonReceiptReasonValue,
            discard—reason     [1] DiscardReasonValue OPTIONAL } } OPTIONAL,
                [1] SEQUENCE {
        comments                            CommentElementId,
            comments                            Comment },
messageReturnedInd [2] MessageReturnedIndElementId OPTIONAL } } }
```

10.1.7 *DeliveryStatusNotice*

A TLMAU terminal sends a DeliveryStatusNotice—TAPDU to invoke the DeliveryStatusNotice abstract operation.

10.1.7.1 *DeliveryStatusNotice—TAPDU*

The DeliveryStatuNotice—TAPDU comprises following elements:

DeliveryStatusNotice—TAPDU

```
DeliveryStatusNotice—TAPDU ::= SEQUENCE {
        [0] SEQUENCE {
            deliveryStatusNotice [0] DeliveryStatusNoticeTAPDUId,
                                    [1] SEQUENCE {
                        quantityOfDocs              QuantityOfDocsElementId,
                            number—of—docs                NumberOfAssociatedDocuments } OPTIONAL },

                                    [2] SEQUENCE {
                        correlationInfo             CorrelationInfoElementId,
                            call—id                       CallIdentification } },
        [1] SET {
                [0] SEQUENCE {
                        submissionId                SubmissionIdElementId,
                            submission—msg—id             MessageIdentifier } OPTIONAL,
                [1] SEQUENCE {
                        probeId                     ProbeIdElementId,
                            submission—msg—id             MessageIdentifier } OPTIONAL,
                [2] SET OF {
                        [0] SEQUENCE {
                    reportedRecipient               ReportedRecipientElementId,
                        reported—recipient—name       ORName },
                        [1] SEQUENCE {
                    notificationType                NotificationTypeElementId,
                        report—type                   ReportTypeValue },
                        [2] SEQUENCE {
                    intendedRecipient               IntendedRecipientElementId,
                        intended—recipient—name       ORName },
                        [3] SEQUENCE {
                    convertedInfoTypes              ConvertedInfoTypesElementId,
                                SET OF
                    eIT                             EITValue },
                            [4] SET {                   —— In case of Delivery Notification,
                                                        —— this set of element shall be present.

                                    [0] SEQUENCE {
                    timeOfDelivery                  TimeOfDeliveryElementId,
                        delivery—time                 DateandTime },
                                    [1] SEQUENCE {
                    typeOfUA                        TypeOfUAElementId,
                        type—of—ua                    TypeOfUA DEFAULT public } OPTIONAL,
                                    [2] SEQUENCE {
                    supplInfo                       SupplInfoElementId,
                        suppl—info                    SuplementaryInformation } OPTIONAL,
                            [5] SET {

                                                        —— In case of Non Delivery Notification,
                                                        —— this set of element shall be present.

                                    [0] SEQUENCE {
                    nonDeliveryReason               NonDeliveryReasonElementId,
                                        SET {
                    reason—code                     [0] ReasonCodeValue,
                    diagostic—code                  [1] DiagnosticCodeValue OPTIONAL } } } OPTIONAL },
contentReturned [3] ContenReturnedElementId OPTIONAL } }
```

10.1.8 *Register*

A TLM terminal sends a Register—TAPDU to invoke the register abstract operation. The TLMAU returns a RegisterAck—TAPDU, if necessary, to report the result of that operation, or may return an Exception—TAPDU (see § 10.1.1.3) to report an abstract error.

10.1.8.1 *Register—TAPDU*

The Register—TAPDU comprises following elements:

Register—TAPDU

```
Register—TAPDU ::= SEQUENCE {
        [0]
                register                        TAPDUIdValue,
        [1] SET {
            [0] SET {
                [0] SEQUENCE {
                        expiredDiscard          ExpiredDiscardElementId,
                            discard—ipm             DiscardValue DEFAULT discard } OPTIONAL,
                [1] SEQUENCE {
                        obsoleteDiscard         ObsoleteDiscardElementId,
                            discard—ipm             DiscardValue DEFAULT discard } OPTIONAL },
            [1] SET {
                [0] SEQUENCE {
                        autoFWDIPMs             AutoFWDIPMsElementId,
                            auto—fwd—ipms               AutoFWDIPMsValue DEFAULT not—auto—forward }
                                                    OPTIONAL,
                [1] SEQUENCE {
                        autoFWDRecipients       AutoFWDRecipientsElementId,
                    SET OF {
                            auto—fwd—recipient—name  ORName } } OPTIONAL,
                [2] SEQUENCE {
                        autoFWDHeading          AutoFWDHeadingElementId,
                            auto—fwd—heading            AutoFWDHeading } OPTIONAL,
            —— For further study
                [3] SEQUENCE {
                        autoFWDComment          AutoFWDCommentElementId,
                            auto—fwd—comment            AutoFWDComment } OPTIONAL },
        [2] SET {
                [0] SEQUENCE {
                        dsMode                  DSModeElementId,
                            ds—mode                 DSModeValue } OPTIONAL,
                [1] SEQUENCE {
                        tLMAUOperation          TLMAUOperationElementId,
                    SET {
                        error—recovery—mode     [0] ErrorRecoveryModeValue OPTIONAL,
                            auto—acknowledgment [1] AutoAcknowledgment DEFAULT manual } } OPTIONAL,
                [2] SEQUENCE {
                        supplRecipientInfo      SupplRecipientInfoElementId,
                            suppl—recipient—info        SuplementaryInformation } OPTIONAL,
                [3] SEQUENCE {
                        autoOutput              AutoOutputElementId,
                    SET {
                        frequency       [0] Frequency OPTIONAL,
                        output—time [1] DateandTime OPTIONAL } } OPTIONAL,
                [4] SEQUENCE {
                        messageDeleteMode       MessageDeleteModeElementId,
                            message—delete—mode         MessageDeleteModeValue DEFAULT
                                                    auto—delete } OPTIONAL } } }
```

10.1.8.2 *RegisterAck—TAPDU*

The RegisterAck—TAPDU comprises following elements:

RegisterAck—TAPDU

```
RegisterAck—TAPDU ::=
        registerAck                 RegisterAckTAPDUId
```

10.1.9 *DSList*

A TLM terminal sends a DSQuery—TAPDU to invoke the DSList abstract operation. The TLMAU returns a DSReport—TAPDU to report the result of that operation, or may return an Exception—TAPDU (see § 10.1.1.3) to report an abstract error.

10.1.9.1 *DSQuery—TAPDU*

The DSQuery—TAPDU comprises following elements:

DSQuery—TAPDU

```
DSQuery—TAPDU ::=
        dsQuery                     DSQueryTAPDUId
```

10.1.9.2 *DSReport—TAPDU*

The DSReport—TAPDU comprises following elements:

DSReport—TAPDU

```
DSReport—TAPDU ::= SEQUENCE {
    [0]
        dsReport                    DSReportTAPDUId,
    [1] SET OF {
            [0] SEQUENCE {
            retrievalId             RetrievalIdElementId,
                retrieval—id            RetrievalIdentifier },
            [1] SEQUENCE {
            messageType             MessageTypeElementId,
                message—type            MessageTypeValue },
            [2] SEQUENCE {
            originator              OriginatorElementId,
                originator—name             ORName } OPTIONAL,
            [3] SEQUENCE {
            priority                PriorityElementId,
                priority—ind            PriorityValue DEFAULT normal } OPTIONAL,
            [4] SEQUENCE {
            messageLength           MessageLengthElementId,
                message—length          MessageLength } OPTIONAL } }
```

10.1.10 *DSDelete*

A TLM terminal sends a MessageDelete—TAPDU to invoke the DSDelete abstract operation. The TLMAU returns no TAPDU to report the result of that operation, or may return an Exception—TAPDU (see § 10.1.1.3) to report an abstract error.

10.1.10.1 *MessageDelete—TAPDU*

The MessageDelete—TAPDU comprises following elements:

MessageDelete—TAPDU

```
MessageDelete—TAPDU ::= SEQUENCE {
                messageDelete [0] MessageDeleteTAPDUId,
                              [1] SEQUENCE {
                    messageSelector          MessageSelectorElementId,
        SET OF {
                    retrieval—id             RetrievalIdentifier } } }
```

10.1.11 *DSFetch*

A TLM terminal sends an OutputRequest—TAPDU to invoke the DSFetch abstract operation. The TLMAU returns an OutputMessage—TAPDU to report the result of that operation, or may return an Exception—TAPDU (see § 10.1.1.3) to report an abstract error.

The OutputMessage—TAPDU is sent by TLMAU to be output the message from DS. This TAPDU is triggered by one of the following events:

1) some rule (not defined in this Recommendation) which causes TLMAU to establish a connection to the TLM terminal and to send a message at a specific time, for example, the TLM terminal has registered its times of availability with TLMAU;

2) the TLM terminal establishes a connection to TLMAU and initiates a CSCC which is taken as an implicit request for output by TLMAU;

3) receipt of an OutputRequest—TAPDU.

10.1.11.1 *OutputRequest—TAPDU*

The OutputRequest—TAPDU comprises following elements:

OutputRequest—TAPDU

```
OutputRequest—TAPDU ::= SEQUENCE {
    [0]
            outputRequest              OutputRequestTAPDUId,
    [1] SET OF SEQUENCE {
        [0] SEQUENCE {
            retrievalId                RetrievalIdElementId,
                retrieval—id               RetrievalIdentifier },
        [1] SEQUENCE {
            deleteAfterOutput          DeleteAfterOutputElementId,
                delete—after—output            DeleteAfterOutputValue } OPTIONAL } }
```

10.1.11.2 *OutputMessage—TAPDU*

The OutputMessage—TAPDU comprises following elements:

OutputMessage—TAPDU

```
OutputMessage—TAPDU ::= SEQUENCE {
    [0] SEQUENCE {
            outputMessage    [0] OutputMessageTAPDUId,
                             [1] SEQUENCE {
            quantityOfDocs          QuantityOfDocsElementId,
                number—of—docs          NumberOfAssociatedDocuments } OPTIONAL },
    [1] SET OF SEQUENCE {
        [0] SEQUENCE {
            retrievalId             RetrievalIdElementId,
                retrieval—id            RetrievalIdentifier },
        [1] SEQUENCE {
            messageType             MessageTypeElementId,
                message—type            MessageTypeValue }
        [2] SEQUENCE {
            timeOfDelivery          TimeOfDeliveryElementId,
                delivery—time           DateandTime }
——      The remaining Components of this TAPDU are identical to the components in the Deliver,
        DeliveryStatusNotice and ReceiptStatusNotice—TAPDU. The actual components to be useddepend
        upon the MessageType parameter value specified in the MessageType component. } }
```

Note — The RetrievalIdentifier is an identifier which identifies a message in DS.

10.1.12 *MessageState*

A TLM terminal sends a StatusQuery—TAPDU to invoke the MessageState abstract operation. The TLMAU returns a StatusReport—TAPDU to report the result of that operation, or returns an Exception—TAPDU to report an abstract error.

10.1.12.1 *StarusQuery—TAPDU*

The StatusQuery—TAPDU comprises following elements:

StatusQuery—TAPDU

```
StatusQuery—TAPDU ::= SEQUENCE {
    [0]
            statusQuery             StatusQueryTAPDUId,
    [1] SET {
        [0] SEQUENCE {
            submissionId            SubmissionIdElementId,
                submission—msg—id       MessageIdentifier } OPTIONAL,
——— See Note
        [1] SEQUENCE {
            correlationInfo         CorrelationInfoElementId,
                call—id                 CallIdentification } OPTIONAL } }
——— See Note
```

Note — If none of these are present all outstanding (in operation), operations will be reported.

10.1.12.2 *StatusReport—TAPDU*

The StatusReport—TAPDU comprises following elements:

StatusReport—TAPDU

```
StatusReport—TAPDU ::= SEQUENCE {
    [0] SEQUENCE {
                statusReport [0] StatusReportTAPDUId
                            [1] SEQUENCE {
                correlationInfo          CorrelationInfoElementId,
                    call—id                  CallIdentification } },
    [1] SET {
            [0] SEQUENCE {
                timeOfReport             TimeOfReportElementId,
                    report—time              DateandTime },
            [1] SEQUENCE {
                reportedMessageId        ReportedMessageIdElementId,
                    reported—message—id      MessageTypeValue }
            [2] SET OF SEQUENCE {
                [0] SEQUENCE {
                actualRecipient          ActualRecipientElementId,
                    actual—recipient—name    ORName },
                [1] SEQUENCE {
                messageStatus       [1] MessageStatusElementId,
                    status               StatusValue },
                            [2] SET {
                                                    —— In case of DN, this set
                                                    —— of element shall be present.
                                [0] SEQUENCE {
                timeOfDelivery           TimeOfDeliveryElementId,
                    delivery—time            DateandTime },
                                [1] SEQUENCE {
                typeOfUA                 TypeOfUAElementId,
                    type—of—ua               TypeOfUA DEFAULT public } OPTIONAL } OPTIONAL },
                            [3] SEQUENCE {
                                                    —— In case of DN, this set
                                                    —— of element shall be present.
                nonDeliveryReason        NonDeliveryReasonElementId,
                                SET {
                    reason—code     [0] ReasonCodeValue,
                diagnostic—code         [1] DiagnosticCodeValue } OPTIONAL } } OPTIONAL } },
```