8 General message format and information element coding

This section should be read in conjunction with § 4 of Recommendation Q.931 and contains the coding of the information elements specifically used by the procedures described in this Recommendation.

8.1 *Message type*

The following additional codings are defined in Table 8-1/Q.932 for message type.

8.2 *Other information elements*

These information elements are coded according to the general coding rules as defined in § 4.5.1 of Recommendation Q.931.

Note — The value used for Protocol discriminator shall be as defined for messages used in Recommendation Q.931.

Table 8-2/Q.932 contains the codepoints allocated to the information elements defined in this Recommendation.

8.2.1 *Endpoint identifier*

The purpose of the Endpoint identifier information element is:

— to indicate the user service identifier and terminal identifier for the purpose of terminal identification; and

— to indicate a specific terminal for the purpose of terminal selection.

(See Annex A for the associated procedures.)

The Endpoint identifier information element is coded as shown in Figure 8-1/Q.932 and Table 8-3/Q.932.

The default maximum length of the Endpoint identifier information element is four octets.

8.2.2 *Facility*

This section defines only the structure and the coding of the Facility information element. Specific procedures that will be required are subject to further study in relation to future Recommendations on specific supplementary services.

The purpose of the Facility information element is to indicate the invocation and operation of supplementary services, identified by the corresponding operation value within the Facility information element. The Facility information element is defined in Figures 8-2/Q.932 to 8-5/Q.932 and Tables 8-4/Q.932 to 8-20/Q.932.

The Facility information element may be repeated in a given message.

The maximum length of the Facility information element is application dependent consistent with the maximum length of the message.

8.2.2.1 *Component (Octets 4, etc.)*

This specification makes use of and is a subset of Recommendations X.208 [7] (Specification of Abstract Syntax Notation One (ANS.1)), X.209 [8] (Specification of basic encoding rules for Abstract Syntax Notation One (ANS.1)), X.219 [9] (Remote operations: model, notation and service definition) and X.229 [10] (Remote operations: protocol specification). Based on Recommendations X.208 and X.209, the following specific encoding apply.

A component is a sequence of data elements each of which is made up of a tag, a length and a contents. The component type is indicated by the first octet of the Facility information element component. The component types defined for the Facility information element are:

— Invoke

— Return result

— Return error

— Reject.

Note 1 — Recommendation X.229 which defines the Remote Operations Service Element (ROSE) uses the term Application Protocol Data Unit (APDU) in place of component. However since this protocol element may be applied to the support of network layer services and of application layer services, the term ‘‘component’’ is more appropriate in the context of this Recommendation.

Tables 8-5/Q.932 to 8-8/Q.932 show the structure of these component types.

Note 2 — See Appendix III for a general description of the component coding and formatting principles.

H.T. [T12.932]

TABLE 8-1/Q.932

Q.932 message types

{
8 7 6 5 4 3 2 1
0 0 1 - - - - -
(Q.931 call information phase message group)
0 0 1 0 0
HOLD
0 1 0 0 0
HOLD ACKNOWLEDGE
1 0 0 0 0
HOLD REJECT
1 0 0 0 1
RETRIEVE
1 0 0 1 1
RETRIEVE ACKNOWLEDGE
1 0 1 1 1
RETRIEVE REJECT
0 1 1 - - - - -
(Q.931 miscellaneous message group)
0 0 0 1 0
FACILITY
0 0 1 0 0
REGISTER
}	

Tableau 8-1/Q.932 [T12.932], p.1

H.T. [T13.932]

TABLE 8-2/Q.932

Information elements specific to supplementary service control

{
Bits
8 7 6 5 4 3 2 1
}
Maximum length (octets)
(Note 1)
0
:
 	
:
 	
:
 	
:
 	
:
 	
:
Variable length information elements:
0 0 1 1 1 0 0
Facility
8.2.2
Note 3
0 1 1 0 0 1 0
Information request
8.2.5
3
0 1 1 1 0 0 0
Feature activation
8.2.3
4
0 1 1 1 0 0 1
Feature indication
8.2.4
5
0 1 1 1 0 1 0
Service profile identification
8.2.6
32
0 1 1 1 0 1 1
Endpoint identifier
8.2.1
4
}
	Reference §	{	

Tableau 8-2/Q.932 [T13.932], p.2

Figure 8-1/Q.932 [T14.932], p.3

H.T. [T15.932]

TABLE 8-3/Q.932

Endpoint identifier information element

{

User service identifier (USID) (octet 3)

The USID is a selection parameter which identifies a group of terminals on an interface which share a common service profile and which may be addressed together. Upon receipt of this element, a terminal will consider itself as being addressed if the value received matches its stored value or if the value received is coded as all ‘‘1’’s (127). When USID is coded as 127, octet 4 is not used.

}

{

Interpreter (octet 4)

Bit 7 of octet 4 indicates how a terminal is to interpret the TID field received. When set to ‘‘0’’, the terminal is being addresses only if the TID matches (see TID definition following). When set to ‘‘1’’, the terminal is being addressed only if the TID received is not 63 and does not match. In the user-to-network direction, this bit is set to ‘‘0’’.

}

{

Terminal identifier (TID) (octet 4)

The TID is a selection parameter which identifies a single terminal within a group designated by a USID value. For USID = 127, the TID does not apply. Upon receipt to this field, a terminal will consider itself addressed if one of the following is true:

—

the interpreter bit = ‘‘0’’ and the value received matches the terminal’s stored value;
— the interpreter bit = ‘‘1’’ and the value received does not match the terminal’s stored value;
— the value received is coded all ‘‘1’’s (63).
}

Tableau 8-3/Q.932 [T15.932], p.4

Figure 8-2/Q.932 [T16.932], p.5

H.T. [T17.932]

TABLE 8-4/Q.932

Facility information element

{
Service discriminator
 	
Bits 5 4 3 2 1
 	
1 0 0 0 1 Discriminator for supplementary service applications
 	
All other values are reserved and their usage is the subject of other Recommendations.
}

Tableau 8-4/Q.932 [T17.932], p.6

Blanc

H.T. [T18.932]
TABLE 8-5/Q.932
Invoke component

Invoke component	Reference §	Mandatory indication	Octet group	
	Component type tag	8.2.2.3	Mandatory	4
	Component length (Note 1)	8.2.2.2		5
{ Invoke identifier tag Invoke identifier length Invoke identifier				
}	8.2.2.4 8.2.2.2	Mandatory	6 7 8	
Linked identifier tag Linked identifier length Linked identifier	8.2.2.4 8.2.2.2	Optional	9 10 11	
Operation value tag Operation value length Operation value	8.2.2.5 8.2.2.2 (Note 3)	Mandatory	12 13 14	
Argument (Note 2)	8.2.2.8 (Note 3)	Optional	15, etc.	

Tableau 8-5/Q.932 [T18.932], p.7

Blanc

H.T. [T19.932]
TABLE 8-6/Q.932
Return result component

Return result component	Reference §	Mandatory indication	Octet group	
	Component type tag	8.2.2.3	Mandatory	4
	Component length (Note 3)	8.2.2.2		5
{				
Invoke identifier tag				
Invoke identifier length				
Invoke identifier				
}	8.2.2.4 8.2.2.2	Mandatory	6 7 8	
Sequence tag	8.2.2.8	Optional	9	
Sequence length (Note 4)	8.2.2.2	(Note 1)	10	
{				
Operation value tag				
Operation value length				
Operation value				
}	8.2.2.5 8.2.2.2 (Note 6)	Optional (Note 2)	11 12 13	
Result (Note 5)	8.2.2.8 (Note 6)	Optional	14, etc.	

Tableau 8-6/Q.932 [T19.932], p.8

Blanc

H.T. [T20.932]
TABLE 8-7/Q.932
Return error component

Return error component	Reference §	Mandatory indication	Octet group	
	Component type tag	8.2.2.3	Mandatory	4
	Component length (Note 1)	8.2.2.2		5
{				
Invoke identifier tag				
Invoke identifier length				
Invoke identifier				
}	8.2.2.4 8.2.2.2	Mandatory	6 7 8	
Error value tag	8.2.2.6	Mandatory	9	
Error value length	8.2.2.2		10	
Error value			11	
Parameter (Note 2)	8.2.2.8 (Note 3)	Optional	12, etc.	

Tableau 8-7/Q.932 [T20.932], p.9

H.T. [T21.932]
TABLE 8-8/Q.932
Reject component

Reject component	Reference	Mandatory indication	Octet group	
	Component type tag	8.2.2.3	Mandatory	4
	Component length (Note)	8.2.2.2		5
{				
Invoke identifier tag				
Invoke identifier length				
Invoke identifier				
}	8.2.2.4 8.2.2.2	Mandatory	6 7 8	
Problem tag	8.2.2.7	Mandatory	9	
Problem length	8.2.2.2		10	
Problem	8.2.2.7		11	

Tableau 8-8/Q.932 [T21.932], p.10

8.2.2.2 *Length of each component or of their data elements*

Lengths up to 127 octets are coded using the short form of Recommendation X.209: bit 8 is set to zero and the remaining seven bits are a binary encoding of the length, with bit 1 the least significant bit. (This length encoding is identical to that of Recommendation Q.931 for lengths up to 127 octets.) This is illustrated in Figure 8-3/Q.932.

Figure 8-3/Q.932 [T22.932], p.

If the length of the contents is greater than 127 octets, then the long form of the length of the contents is used. The long form length is from 2 to 127 octets long. Bit 8 of the first octet is coded 1, and bits 1 to 7 of the first octet encode a number one less than the size of the length in octets as an unsigned binary number whose MSB and LSB are bits 7 and 1, respectively. The length itself is encoded as an unsigned binary number whose MSB and LSB are bit 8 of the second octet and bit 1 of the last octet, respectively. This binary number should be encoded in the fewest possible octets, with no leading octets having the value 0. This is illustrated in Figure 8-4/Q.932.

Figure 8-4/Q.932 [T23.932], p.

8.2.2.3 *Component type tag*

The coding of the component type tag is shown in Table 8-9/Q.932.

H.T. [T24.932]
TABLE 8-9/Q.932
Component type tag

{								
Invoke	1	0	1	0	0	0	0	1
Return result	1	0	1	0	0	0	1	0
Return error	1	0	1	0	0	0	1	1
Reject	1	0	1	0	0	1	0	0

Table 8-9/Q.932 [T24.932], p.

8.2.2.4 *Component identifier tags*

An invoke identifier is used to identify an operation invocation and is reflected in the return result or return error that responds to it. An invoke may refer to another invoke through the linked identifier. When a protocol error occurs, the invoke identifier is reflected in the reject component, but if it is not available, a null is returned. Invoke and linked identifiers are one octet long. The null has zero length. The coding of the component identifier tags is shown in Table 8-10/Q.932.

H.T. [T25.932]
TABLE 8-10/Q.932
Coding of component identifier tag

Bits 8 7 6 5 4 3 2 1								
Invoke identifier	0	0	0	0	0	0	1	0
Linked identifier	1	0	0	0	0	0	0	0
Null	0	0	0	0	0	1	0	1

Tableau 8-10/Q.932 [T25.932], p.14

Blanc

8.2.2.5 *Operation value tag*

The operation value specifies the facility or supplementary service application and operation being requested. Values are encoded as integers. The value of the operation value is supplementary service specific and will be specified in future Recommendations which contain the protocol for individual supplementary services. The coding for the operation value tag is shown in Table 8-11/Q.932.

H.T. [T26.932]

TABLE 8-11/Q.932

Coding of operation value tag

Bits 8 7 6 5 4 3 2 1								
Operation value tag	0	0	0	0	0	0	1	0

Table 8-11/Q.932 [T26.932], p.

8.2.2.6 *Error value tag*

Operations report errors as specified for each individual operation. Values are encoded as integers. The coding for the error value tag is shown in Table 8-12/Q.932.

H.T. [T27.932]

TABLE 8-12/Q.932

Coding of error value tag

Bits 8 7 6 5 4 3 2 1								
Error value tag	0	0	0	0	0	0	1	0

Tableau 8-12/Q.932 [T27.932], p.16

Blanc

8.2.2.7 *Problem tag*

Protocol problems are indicated in groups. Table 8-13/Q.932 indicates the tags for these groups. The contents for each of these tags is indicated in Tables 8-14/Q.932 to 8-17/Q.932. The contents of these tags are defined in Table 8-18/Q.932.

H.T. [T28.932]
TABLE 8-13/Q.932
Coding of problem tags

{								
General problem	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Invoke problem	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**1**
Return result problem	**1**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
Return error problem	**1**	**0**	**0**	**0**	**0**	**0**	**1**	**1**

Tableau 8-13/Q.932 [T28.932], p.17

H.T. [T29.932]
TABLE 8-14/Q.932
Coding of general problem

Bits 8 7 6 5 4 3 2 1								
Unrecognized component	0	0	0	0	0	0	0	0
Mistyped component	0	0	0	0	0	0	0	1
Badly structured component	0	0	0	0	0	0	1	{
0								
Note								
— ROSE uses the term application protocol data unit (APDU) in place of component.								
}								

Tableau 8-14/Q.932 [T29.932], p.18

Blanc

H.T. [T30.932]

TABLE 8-15/Q.932

Coding of invoke problem

Bits 8 7 6 5 4 3 2 1								
Duplicate invocation	0	0	0	0	0	0	0	0
Unrecognized operation	0	0	0	0	0	0	0	1
Mistyped argument	0	0	0	0	0	0	1	0
Resource limitation	0	0	0	0	0	0	1	1
Initiator releasing	0	0	0	0	0	1	0	0
{ Unrecognized linked identifier }	0	0	0	0	0	1	0	1
Linked response unexpected	0	0	0	0	0	1	1	0
Unexpected child operation	0	0	0	0	0	1	1	1

Tableau 8-15/Q.932 [T30.932], p.19

H.T. [T31.932]

TABLE 8-16/Q.932

Coding of return result problem

Bits 8 7 6 5 4 3 2 1								
Unrecognized invocation	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Result response unexpected	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**
Mistyped result	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**

Tableau 8-16/Q.932 [T31.932], p.20

H.T. [T32.932]

TABLE 8-17/Q.932

Coding of return error problem

Bits 8 7 6 5 4 3 2 1								
Unrecognized invocation	0	0	0	0	0	0	0	0
Error response unexpected	0	0	0	0	0	0	0	1
Unrecognized error	0	0	0	0	0	0	1	0
Unexpected error	0	0	0	0	0	0	1	1
Mistyped parameter	0	0	0	0	0	1	0	0

Tableau 8-17/Q.932 [T32.932], p.21

H.T. [T33.932]

TABLE 8-18/Q.932

Problem code definitions

General-problem
{
—
unrecognized-component
} {
signifies that the type of the component, as evidenced by its type identifier, is not one of the four defined by Recommendation X.229 [10]
}
— mistyped-component {
signifies that the structure of the component does not conform to Recommendation X.229
}
{
—
badly-structured-component
} {
signifies that the struture fo the component does not conform to the standard notation and encoding, defined in Recommendations X.208 [7] and X.209 [8]
}
Invoke-problem
— duplicate-invocation {
signifies that the invoke-identifier parameter violates the assignment rules of Recommendation X.219 [9]
}
{
—
unrecognized-operation
} {
signifies that the operation is not one of those agreed between the user and the network
}
— mistyped-argument {
signifies that the type of the operation argument supplied is not that agreed between the user and the network
}
— resource-limitation {
the performing user or network is not able to perform the invoked operation due to resource limitation
}
— initiator-releasing {
the association-initiator is not willing to perform the invoked operation because it is about to attempt to release the application-association
}
{
—
unrecognized-linked-identifier
} {
signifies that there is no operation in progress with an invoke-identifier equal to the specified linked-identifier
}
{
—
linked-response-unexpected
} {
signifies that the invoked operation referred to by linked-identifier is not a parent-operation
}
{
—

unexpected-child-operation
}
signifies that the invoked child-operation is not one that the invoked parent-operation referred to by the linked-identifier allows
}
{
Return-result—-roblem
}
{
—
unrecognized-invocation
}
signifies that no operation with the specified invoke-identifier is in progress
}
{
—
result-response-unexpected
}
signifies that the invoke operation does not report a result
}
— mistyped-result
signifies that the type of the result parameter supplied is not that agreed between the user and the network
}
{
Return-error-problem
}
{
—
unrecognized-invocation
}
signifies that no operation with the specified invoke-identifier is in progress
}
{
—
error-response-unexpected
}
signifies that the invoked operation does not report failure
}
— unrecognized-error
signifies that the reported error is not one of those agreed between the user and the network
}
— unexpected-error
signifies that the reported error is not one that the invoked operation may report
}
— mistyped-parameter
signifies that the type of the error parameters supplied is not that agreed between the user and the network
Note
— The former definitions are adapted from §§ 7.4.4.2 and 7.5.4.2 of Recommendation X.229 (Remote operations: protocol specification).
}
	{
	{
	{
	{
	{
	{

Tableau 8-18/Q.932 [T33.932], p.22

Blanc

8.2.2.8 *Parameters*

The parameters included with a component (i.e., the argument with an invoke, the result with a return result or the parameter with a return error) are indicated in the specification of the operation. They may include optional and default parameters. Parameters shall be one of the following:

— a sequence of parameters

— a set of parameters

— a specific parameter with its own tag

— nothing at all (i.e., absent).

When more than one parameter is required, they shall follow a sequence or set tag as specified in the specification of the operation. (The usage of the sequence and set tags is defined in Recommendations X.208/X.209.)

Sequences and sets of parameters may contain further sequences and sets as specified for the operation to be performed. Table 8-19/Q.932 indicates the coding of the sequence and set tags.

H.T. [T34.932]

TABLE 8-19/Q.932

Coding of sequence and set tags

Bits 8 7 6 5 4 3 2 1								
Sequence tag	0	0	1	1	0	0	0	0
Set tag	0	0	1	1	0	0	0	1

Table 8-19/Q.932 [T34.932], p.

8.2.2.9 *Treatment of existing Recommendation Q.931 information elements as parameters*

Supplementary service protocol specifications are expected to require new parameters to be defined and to require existing Recommendation Q.931 information elements (Note 1).

New parameters shall be defined using Recommendation X.209 coding if they do not appear elsewhere in Q.931 messages.

Supplementary service protocol specifiers may elect to encapsulate one or more existing Recommendation Q.931 information elements within a Recommendation X.209 data element, thereby retaining the Recommendation Q.931 coding for these information elements. When this option is chosen, all the Recommendation Q.931 information elements should be grouped together as the content following the Recommendation Q.931 information elements tag. This is illustrated in Figure 8-5/Q.932. The tag is defined in Table 8-20/Q.932. This data element may appear by itself or as a member of a sequence or set as indicated in § 8.2.2.8.

Note 1 — Encapsulation of the Facility information element within Facility information elements shall not be used.

Blanc

Figure 8-5/Q.932, p.24

H.T. [T35.932]
TABLE 8-20/Q.932
Q.931 information elements tag

Bits 8 7 6 5 4 3 2 1								
Q.931 information elements	0	1	0	0	0	0	0	0

Note — All other values are reserved but this approach may also be applied in the future to coding structures from other Recommendations by defining other tags as required.

Tableau 8-20/Q.932 [T35.932], p.25

8.2.3 *Feature activation*

The purpose of the Feature activation information element is to invoke a supplementary service as identified by the feature identifier number. The service associated with the feature identifier number is dependent on that particular user's service profile.

The maximum length of this information element is 4 octets.

The Feature activation information element is coded as shown in Figure 8-6/Q.932 and Table 8-21/Q.932.

Figure 8-6/Q.932 [T36.932], p.

H.T. [T37.932]

TABLE 8-21/Q.932

Feature activation information element

{
Feature identifier number (octets 3 and 3a)
 	
The feature identifier number is a unique number assigned to a feature in a customer account that is coded as part of both the Feature activation and Feature indication information elements. This number identifies the feature that is being requested or updated. The association of a particular number to a particular feature may be different for each user.
 	
Bit 8 in octet 3 is used to extend the feature identifier field. If bit 8 is 0, then another octet follows; if bit 8 is 1, then octet 3 is the last octet. The identifier numbers for a one octet field range from 1 to 127. For a multi-octet field, the order of bit values progressively decrases as the octet number increases.
}

Table 8-21/Q.932 [T37.932], p.

8.2.4 *Feature indication*

The purpose of the Feature indication information element is to allow the network to convey feature indications to the user regarding the status of a supplementary service.

The maximum length of this information element is 5 octets.

The coding of the Feature indication information element is shown in Figure 8-7/Q.932 and Table 8-22/Q.932.

Figure 8-7/Q.932 [T38.932], p.

H.T. [T39.932]

TABLE 8-22/Q.932
Feature indication information element

{
Feature identifier number (octets 3 and 3a)
}
{
These fields are coded as described in Table 8-21/Q.932.
}
{
Status indicator (octet 4)
}
{
The status indicator field identifies the current status of a supplementary service.
}

Bits .line 4 3 2 1 *Examples of possible user equipment implementation* }	*Status*	*Meaning*	{
0 0 0 0 Feature is in the deactivated state }	Deactivated Lamp off	{	
0 0 0 1 Feature is in the active state }	Activated Lamp steady on	{	
0 0 1 0 Feature prompt (waiting for user input) }	Prompt Lamp steady flash	{	
0 0 1 1	Pending	Feature is pending	Lamp steady wink

{
All other values are reserved.
}

Table 8-22/Q.932 [T39.932], p.

8.2.5 *Information request*

The purpose of the Information request information element is to provide the capability for requesting additional information and signalling completion of the information request (see Annex B).

The Information request information element is coded as shown in Figure 8-8/Q.932 and Table 8-23/Q.932.

The default maximum length of the Information request information element is three octets.

Figure 8-8/Q.932 [T40.932], p.

H.T. [T41.932]

TABLE 8-23/Q.932

Information request information element

{

Information request indicator (octet 3, bit 7)

Bit 7

0 Information request completed

1 Prompt for additional information

}

{ *Type of information (octet 3, bits 1-6)*

Bits 6 5 4 3 2 1

0 0 0 0 0 0 undefined

0 0 0 0 0 1 authorization code

0 0 0 0 1 0 address digits

0 0 0 0 1 1 terminal identification

}	All other values are reserved.

Table 8-23/Q.932 [T41.932], p.

8.2.6 *Service profile identification*

The purpose of the Service profile identification information element is to allow the user to initiate automatic assignment of the user service identifier and terminal identifier (see Annex A).

The Service profile identification information element is defined in Figure 8-9/Q.932 and Table 8-24/Q.932.

The default maximum length of the Service profile identification information element is 32 octets.

Figure 8-9/Q.932 [T42.932], p.

H.T. [T43.932]

TABLE 8-24/Q.932

Service profile identification information element

{
SPID (octet 3, etc.)
 	
The service profile identifier parameter is coded in IA5 characters, according to the format specified by the network.
}

Table 8-24/Q.932 [T43.932], p.

Blanc

ANNEX A
(to Recommendation Q.932)

User service profiles and terminal identification

A.1 *Introduction*

These optional procedures allow an ISDN to support identification and selection of specific terminals on a multi-point user-network interface to support multiple user service profiles in those cases in which Recommendation Q.931 information elements are not sufficient for such purposes.

A terminal or network which desires to support such multiple profiles for terminals which could not otherwise be distinguished, must support this additional identification procedure. Otherwise, it is completely optional.

H.T. [T44.932]

TABLE A-1/Q.932

Terminology

Service profile **Service profile refers to the information that the network maintains for a given user to characterize the service offered by the network to that user. As an example, this may contain the association of feature identifiers to specific supplementary services. A service profile may be allocated to an access interface or to a particular user equipment or a group of user equipments.** **}**	**{**
SPID **The service profile identifier is a parameter carried in a service profile identification information element that is sent from the user to network to allow network assignment of a USID and TID. A user's SPID should uniquely identifiy a specific profile of service characteristics stored within the network.** **The SPID will allow the network to distinguish between different terminals that would otherwise be indistinguishable (e.g., same ISDN number). The SPID value is provided to the user at subscription time.** **}**	**{**
USID **User service identifier. A USID uniquely identifies a service profile on an access interface.** **}**	**{**
TID **Terminal identifier. A TID value is unique within a given USID. If two terminals on an interface subscribe to the same service profile, then the two terminals will be assigned the same service USID.** **However, two different TIDs are required to uniquely identify each of the two terminals.** **}**	**{**
EID **Endpoint identifier. The endpoint identifier information element is used for terminal identification. The endpoint identifier parameters contain a USID and TID and additional information used to interpret them.** **}**	**{**

Table A-1/Q.932 [T44.932], p.

Figure A-1/Q.932 shows examples of the relationships of terminals, SPIDs, USIDs, and TIDs and their dynamic relationship to TEIs. In this example, terminals 1, 3, 4 and 5 support the automatic endpoint identifier parameter assignment procedure and terminal 2 does not, but has the endpoint identifier parameters locally entered. Terminal 6 does not support terminal identification, therefore it utilizes the specified default service profile.

Note — Items in parentheses indicate values or relationships which are dynamically established by initialization procedures (see § A.4). Others are established via administrative actions and stored as a result of manual entry.

Figure A-1/Q.932, p.

A user or network that does not recognize the information elements used by this Annex shall, if these elements are received, apply the error procedures defined in § 5.8 of Recommendation Q.931.

A.2 *User service profiles*

The support of user service profiles requires that the service requests from a terminal are associated by the network with a specific profile. A USID is used to identify the profile on an access. The service profile is assigned to a data link connection so that the network can associate all of the service requests from the corresponding Connection Endpoint Suffix (CES) with the required profile (see Note). The assignment of a service profile to a data link connection minimizes the per-service request overhead of profile identification.

The procedures for assigning service profile to a data link connection are incorporated into the initialization procedures described in § A.4.

Note — CES along with SAPI constitute the CEI (Connection Endpoint Identifier) that is used to identify message units passed between the data link layer (as represented by the TEI) and Layer 3.

A.3 *Terminal identification*

The support of terminal identification requires that a call sent by the network can be addressed to:

— all of the terminals of a user service profile;

— one terminal of a user service profile; or

— all but one terminal of a user service profile.

A USID is used to identify the user service profile with a (set of) terminals on an access interface and a TID is used to identify individual terminals within a user service profile on an access.

The USID and TID may be entered into the terminal by the user as arranged at subscription time, or dynamically downloaded to the terminal from the network with an automatic assignment procedure.

The USID and TID parameters are used by the terminal to check the compatibility of a call offered by the network. The inclusion of a USID and TID with only access uniqueness minimizes the per-call overhead of supporting terminal addressing.

The procedures for downloading the USID and TID to a terminal are incorporated into the automatic endpoint identifier allocation and initialization procedures described in § A.4. The procedures for using a USID and TID for terminal identification in an offered call sent by the network are described in § A.5.

A.4 *Initialization*

The initialization procedure provides for the association by the network of the service requests from a terminal on a particular data link connection (as represented by the TEI) with a user service profile. A user requested automatic assignment procedure is described to also support automatic assignment of USID and TID parameters and their downloading by the network to a terminal.

Since initialization provides the basis for subsequent association of a service profile with a data link connection, normally, user equipment that supports initialization is expected to request the initialization procedure (e.g., on the first Layer 3 message after dynamic assignment of a TEI). However, a request for initialization is allowed at any time. The data link connection is always associated with the most recently identified service profile. Under some circumstances, the network may solicit terminal initialization.

A.4.1 *Terminal requested initialization*

a) Terminals may initialize by sending an Endpoint identifier information element (containing a USID and TID) in an INFORMATION message at any time to the network. Subsequent to this, the network may associate the service profile with the data link over which the message was sent.

b) For terminals which support automatic assignment of USID and TID parameters, initialization (that is, association of a service profile with a data link connection) is provided as part of the automatic assignment procedure described here.

A user may initiate automatic assignment of the endpoint identifier by sending a Service profile identification information element in an INFORMATION message with the dummy call reference. The Service profile identification information element should contain the SPID parameter allocated at the time of subscription. The initialization is acknowledged with an INFORMATION message with the Endpoint identifier information element containing a USID and TID, the values of which are determined by the network. It results in an association of the data link over which it was received with the identified service profile.

When a terminal determines that the initialization procedure has failed, it assumes that the network cannot support the procedure and does not repeatedly attempt initialization.

A.4.2 *Network solicited initialization*

The network may solicit a request for initialization on a data link connection by sending an Information request information element with codepoint ‘‘terminal identification’’ in an INFORMATION message with the dummy call reference. Upon receiving the request, the terminal may respond as described in the previous § A.4.1 a) or b).

When a network determines that the initialization procedure has failed, it assumes that the terminal cannot support the procedures and does not repeatedly request initialization.

A.4.3 *Collision*

When terminal initialization and network solicitation procedures collide, the terminal ignores the solicitation from the network and the network proceeds as normal upon receipt of the initialization request from the terminal.

A.5 *Identification procedures*

When the network offers a call using terminal addressing, the Endpoint identifier information element is included in the SETUP message.

When a terminal receives a SETUP message containing the Endpoint identifier information element, it shall:

— if it is not supported, handle the Endpoint identifier information element in accordance with § 5.8.7 of Recommendation Q.931 and complete normal compatibility checking procedures; or,

— test for an address compatibility with the Endpoint identifier information element if it is supported in addition to completing the normal compatibility checking procedures.

ANNEX B
(to Recommendation Q.932)

Information request procedures

B.1 *Introduction*

This Annex specifies optional procedures to allow a network to request additional information from a user. These procedures do not impact the Recommendation Q.931 call state. This capability shall only be allowed during the Null, Overlap Sending, Outgoing Call Proceeding, Call Delivered, and Activate call states.

The capability is intended for use with the Keypad and Feature key management protocols.

A user or network that does not recognize the information elements used by this Annex shall, if these information elements are received, apply the error recovery procedures defined in § 5.8 of Recommendation Q.931.

B.2 *Procedures*

B.2.1 *Normal procedures*

The network will send an INFORMATION message to the user to request additional information. The INFORMATION message will contain the Information request information element (see § 8), with the information request indicator set to ‘‘prompt for additional information’’ and type of information set to the appropriate value. After sending the INFORMATION message, the network will start timer T302. The network will restart timer T302 on the receipt of every INFORMATION message if the requested information is not complete.

No Recommendation Q.931 call state changes should occur when the INFORMATION message is sent or received.

The user may always send the requested information in keypad facility information elements contained in one or more INFORMATION messages. In addition, if the information requested was a called party number, then the user may also send the requested information in the called party number information element in one or more INFORMATION messages.

In both the call associated and non-call associated cases, when the network has determined that sufficient information has been received to proceed, it may send an INFORMATION message to the user, containing an

Information request information element, with the information request indicator set to ''information request completed'' to signal the required information has been received correctly. If the additional information was requested during Overlap Sending state, and no additional information is required before the network can proceed with processing of the call, a CALL PROCEEDING message may suffice to signal the end of information sending.

In the call associated case, the network may also indicate that sufficient information has been received by initiating call clearing according to § 5.3 of Recommendation Q.931.

B.2.2 *Abnormal procedures*

If no response is received from the user, or if the information received is incomplete upon expiry of timer T302, or if the information provided by the user is invalid, then:

— in the call associated case, the network shall initiate call clearing according to § 5.3 of Recommendation Q.931;

— in the non-call associated case, the network shall return an INFORMATION message containing a Cause information element with an appropriate cause value.

In the non-call associated case, if the user responds with a RELEASE COMPLETE message to an INFORMATION message containing an Information request information element, then the procedure shall be considered as terminated.

APPENDIX I
(to Recommendation Q.932)

Illustration of the application of the three protocol types

I.1 *Introduction*

This Appendix is provided as an illustration of the application of the three protocol types defined in this Recommendation. The examples shown should not be taken as definitive examples, since the support of the Keypad and the Feature key management protocols are network dependent.

The signalling sequences shown are not exhaustive and are only intended to illustrate possible supplementary service control sequences.

I.2 *Example use of the Keypad protocol*

This example shows the application of the Keypad protocol using the Keypad facility and Display information elements to establish a second call while holding the first one. It should be noted that the Keypad protocol does not necessarily allow a supplementary service to be supported to the same degree of functionality as the approach based on the Functional protocol. In addition, this protocol does not impose a need for the terminal to be aware of any states other than those required for basic call control. An objective of the Keypad protocol is to provide for the support of supplementary services in circumstances where a reduced level of functionality can be tolerated.

The example in Figure I-1/Q.932 illustrates a user feature request using the Keypad protocol. The network associates the contents of the Keypad information element with the appropriate feature. The user is shown to subsequently enter supplementary service parameters using the Keypad protocol. Feature status information may be provided by the network in the Display information element. The network completes feature processing and the user is shown to clear call reference. Alternatively, depending on the specific feature request, a CALL PROCEEDING message might be returned by the network and normal call processing procedures would continue.

The specific example shown in Figure I-2/Q.932 illustrates the support of a holdB/Fretrieve function based on the use of INFORMATION messages for the conveyance of Keypad facility or Display information elements. An enquiry call is established through the conveyance of the called party address digits via a Keypad facility information element within INFORMATION messages. These address digits are sent after putting the existing call on hold through the transfer of a facility request via a Keypad facility information element within an INFORMATION message.

Figure I-1/Q.932 [T45.932], p.

Figure I-2/Q.932, p.

I.3 *Example of use of the Feature key management protocol*

This example illustrates the use of the Feature key management protocol for the invocation of a supplementary service by a user having initiated a call establishment by sending a SETUP message with incomplete (or no) address information, after having entered the overlap sending state upon receipt of the SETUP ACKNOWLEDGE message. Figure I-3/Q.932 depicts the user providing supplementary service parameters. This is accomplished via the Keypad facility information element within INFORMATION messages after having invoked the request of a supplementary service by sending a Feature activation information element contained in an INFORMATION message to the network. The association of the feature identifier number (provided within the Feature activitation information element) with a given supplementary service has to be arranged between the user and the network at subscription time.

Figure I-3/Q.932, p.

I.4 *Examples of use of the Functional protocol*

I.4.1 *Call related supplementary service procedures*

I.4.1.1 *Invocation with call establishment*

The example message sequence shows the initiation of a call establishment simultaneously with a supplementary service invocation.

Figure I-4/Q.932, p.

I.4.1.2 *Invocation with call clearing*

The example message sequence shows the initiation of normal call clearing simultaneously with a supplementary service invocation.

Figure I-5/Q.932, p.

I.4.1.3 *Invocation during the active phase of a call*

The example message sequence shows the initiation of a supplementary service via the established signalling association CR_a at any time during the active phase of a call.

Figure I-6/Q.932, p.

I.4.2 *Call independent supplementary service procedures*

I.4.2.1 *Establishment of a user-to-network transaction for supplementary service control*

Figure I-7/Q.932, p.

I.4.2.2 *Clearing of a user-to-network transaction for supplementary service control*

Figure I-8/Q.932, p.

H.T. [T46.932]

TABLE I-1/Q.932

Key to the Figures I-1/Q.932 to I-8/Q.932

Layer 2 frames:	
SABME	{
Set asynchronous balance mode extended	
}	
UA	{
Unnumbered acknowledgement frame	
}	
DISC	Disconnect frame
Layer 3 messages:	
INFO	*Information*
SETUP ACK	*Setup acknolwledge*
DISC	*Disconnect*
REL	*Release*
REL COMP	*Release complete*
{	
Layer 3 message information elements/parameters:	
}	
FAC	Facility information element
F	Facility identifier
Invoke	Invoke operation type
RR	Return result operation type
RE	Return error operation type
CR a	{
Call reference of an active call	
}	
CR 1	{
Call reference assigned call independently	
}	

Table I-1/Q.932 [T46.932], p.

APPENDIX II
(to Recommendation Q.932)

Functional reference model for the operation

of supplementary services

This Appendix provides a functional model intended to show how the supplementary services can be operated by combining stimulus or Functional protocol types to interact with a unique supplementary service protocol controller which interfaces with the relevant supplementary functional components which provides and coordinates the required functions associated to each supplementary service (e.g., control of resources).

The intermediate feature function performs the necessary conversions between stimulus protocols and the supplementary service functional primitives which are the only ones treated and known from the supplementary service protocol controller. As an example, the intermediate feature function translates an access code received within the Keypad facility information element or a feature identifier number within a Feature activation information element to a supplementary service priority such as hold or retrieve request.

Blanc

Figura II-1/Q.932, p.45

APPENDIX III
(to Recommendation Q.932)

General description of component encoding rules

III.1 *General component structure*

Each data element within a component has the same structure. A data element consists of three fields, which always appear in the following order. The tag distinguishes one type from another and governs the interpretation of the contents. The length specifies the length of the contents. The contents is the substance of the data element, containing the primary information the data element is intended to convey. Figure III-1/Q.932 shows an overview of a component and a data element.

Figure III-1/Q.932, p.

Each field is coded using one or more octets. Octets are labelled as shown in Figure III-2/Q.932. The first octet is the first transmitted. Bits in an octet are labelled as shown in Figure III-3/Q.932, with bit 1 the least significant and the first transmitted.

Figure III-2/Q.932, p.

Figure III-3/Q.932, p.

The contents of each data element is either one value (primitive) or one or more data elements (constructor), as shown in Figure III-4/Q.932.

Figure III-4/Q.932, p.

III.2 *Tag*

A data element is first interpreted according to its position within the syntax of the message. The tag distinguishes one data element from another and governs the interpretation of the contents. It is one or more octets in length. The tag is composed of ‘‘class’’, ‘‘form’’ and ‘‘tag code’’, as shown in Figure III-5/Q.932.

Figure III-5/Q.932 [T47.932], p.

III.2.1 *Tag class*

All tags use the two most significant bits (8 and 7) to indicate the tag class. These bits are coded as shown in Table III-1/Q.932.

H.T. [T48.932]

TABLE III-1/Q.932

Coding of tag class

Class	Coding (87)
Universal	00
Application-wide	01
Context-specific	10
Private use	11

Table III-1/Q.932 [T48.932], p.

The universal class is used for tags that are exclusively standardized in Recommendation X.209 and are application independent types. Universal tags may be used anywhere a universal data element type is used. The universal class applies across all CCITT Recommendations, i.e., across Recommendation Q.932 facility information elements, CCITT Signalling System No. 7 ASEs, X.400 MHS, X.500 Directory Services, etc.

The application-wide class is used for data elements that are standardized across all applications (ASEs) using CCITT Recommendation Q.932 facility procedures for supplementary services.

The context-specific class is used for data elements that are specified within the context of the next higher construction and take into account the sequence of other data elements within the same construction. This class may be used for tags in a construction, and the tags may be re-used in any other construction.

The private use class is reserved for data elements specific to a nation, a network or a private user. Such data elements are beyond the scope of Recommendation Q.932.

The Tag codes of the application-wide class not assigned in Recommendation Q.932 are reserved for future use.

III.2.2 *Form of the data element*

Bit 6 is used to indicate whether the data element is ‘‘primitive’’ or is one whose structure is atomic (i.e., one value only). A constructor element is one whose content is one or more data elements which may themselves be constructor elements.

Both forms of elements are shown in Figure III-4/Q.932.

H.T. [T49.932]

TABLE III-2/Q.932

Coding of element form

Element form	Coding (6)
Primitive	0
Constructor	1

Table III-2/Q.932 [T49.932], p.

III.2.3 *Tag code*

Bits 1 to 5 of the first octet of the tag plus any extension octets represent a tag code that distinguishes one element type from another of the same class. Tag codes in the range 00000 to 11110 (0 to 30 decimal) are provided in one octet.

The extension mechanism is to code bits 1 to 5 of the first octet as 11111. Bit 8 of the following octet serves as an extension indication. If bit 8 of the extension octet is set to 0, then no further octets for this tag are used. If bit 8 is set to 1, the following octet is also used for extension of the tag code. The resultant tag consists of bits 1 to 7 of each extension octet with bit 7 of the first extension octet being most significant and bit 1 of the last extension octet being least significant. Tag code 31 is encoded as 0011111 in bits 7 to 1 of a single extension octet. Higher tag codes continue from this point using the minimum possible number of extension octets.

Figure III-6/Q.932 shows the detailed format of the tag code.

Figure III-6/Q.932 [T50.932], p.

III.3 *Length of the contents*

The length of the contents is coded to indicate the number of octets in the contents. The length does not include the tag nor the length of the contents octets.

The length of the contents uses the short, long or indefinite form. If the length is less than 128 octets, the short form is used. In the short form, bit 8 is coded 0, and the length is encoded as a binary number using bits 1 to 7.

If the length of the contents is greater than 127 octets, then the long form of the length of the contents is used. The long form length is from 2 to 127 octets long. Bit 8 of the first octet is coded 1, and bits 1 to 7 of the first octet encode a number one less than the size of the length in octets as an unsigned binary number whose MSB and LSB are bits 7 and 1, respectively. The length itself is encoded as an unsigned binary number whose MSB and LSB are bit 8 of the second octet and bit 1 of the last octet, respectively. This binary number should be encoded in the fewest possible octets, with no leading octets having the value 0.

The indefinite form is one octet long and may (but need not) be used in place of the short or long form, whenever the element is a constructor. It has the value 10000000. When this form is employed, a special end-of-contents (EOC) indicator terminates the contents.

There is no notation for the end-of-contents indicator. Although considered part of the contents syntactically, the end-of-contents indicator has no semantic significance.

The representation for the end-of-contents indicator is an element whose class is universal, whose form is primitive, whose identifier code has the value 0, and whose contents is unused and absent (see Table III-3/Q.932).

H.T. [T51.932]

TABLE III-3/Q.932

Representation for the end-of-contents indicator

EOC 00 (hex)	Length 00 (hex)	Contents Absent

Table III-3/Q.932 [T51.932], p.

Figure III-7/Q.932 shows the formats of the length field described above. The maximum value that may be encoded is constrained by Q.931 information element size limitations.

Figure III-7/Q.932 [T52.932], p.

III.4 *Contents*

The contents is the substance of the data element and contains the information the data element is intended to convey. Its length is variable, but always an integral number of octets. The contents is interpreted in a type-dependent manner, i.e., according to the tag value.

Acronyms used in Recommendation Q.932

English	French	Spanish	Meaning
APDU	APDU	UDPA	Application Protocol Data Unit
ASN.1	ASN.1	NSA.1	Abstract Syntax Notation One (see Recommendations X.208/X.209)
CEI	CEI	IEC	Connection Endpoint Identifier (see Recommendation Q.920)
CES	CES	SEC	Connection Endpoint Suffix (see Recommendation Q.920)
IA5	IA5	AI5	International Alphabet No. 5
ISDN	RNIS	RDSI	Integrated Services Digital Network
LSB	LSB	BMenosS	Least Significant Bit
MSB	MSB	BM´asS	Most Significant Bit
NT2	NT2	TR2	Network Termination Type Two (see Recommendation I.411)
ROSE	ROSE	ESOR	Remote Operations Service Element (see Recommendations X.219/X.229)
SAPI	SAPI	IPAS	Service Access Point Identifier (see Recommendation Q.920)
SPID	SPID	IDPS	Service Profile Identifier
TEI	TEI	IET	Terminal Endpoint Identifier (see Recommendation Q.920)
TID	TID	IDT	Terminal Identifier
USID	USID	IDSU	User Service Identifier

References

[1] CCITT Recommendation *Basic user-network interface — Layer 1 specification* , Vol. III, Rec. I.430.

[2] CCITT Recommendation *Primary rate user-network interface — Layer 1 specification* , Vol. III, Rec. I.431.

[3] CCITT Recommendation *ISDN user-network interface — Data link layer specification* , Vol. VI, Rec. Q.921.

[4] CCITT Recommendation *ISDN user-network interface layer 3 specification for basic call control* , Vol. VI, Rec. Q.931.

[5] CCITT Recommendation *ISDN user-network interface layer 3 — General aspects* , Vol. VI, Rec. Q.930.

[6] CCITT Recommendation *ISDN user-network interface data link layer — General aspects* , Vol. VI, Rec. Q.920.

[7] CCITT Recommendation *Specification of abstract syntax notation specification of abstract syntax notation one (ASN.1)* , Vol. VIII, Rec. X.208.

[8] CCITT Recommendation *Specification of basic encoding rules for abstract syntax notation one (ASN.1)* , Vol. VIII, Rec. X.209.

[9] CCITT Recommendation *Remote operations: model, notation and service definition* , Vol. VIII, Rec. X.219.

[10] CCITT Recommendation *Remote operations: protocol specification* , Vol. VIII, Rec. X.229.

MONTAGE: PAGE 424 = PAGE BLANCHE

SECTION 2

USER-NETWORK MANAGEMENT

Recommendation Q.940

ISDN USER-NETWORK INTERFACE PROTOCOL

FOR MANAGEMENT — GENERAL ASPECTS

1 General

This Recommendation is one of a proposed series of Recommendations describing the management model, service elements and protocol to be provided at the ISDN user-network interface. These Recommendations also specify the management functions required to support the ISDN subscriber installation. This Recommendation describes the Management Architecture and provides a general overview of the management services and functions.

Other Recommendations in this series will specify the System Management Service Elements and Protocol and the procedures associated with management functions.

The management functions provided at the user-network interface have, as an objective, full alignment with the network management functions being addressed by the Telecommunications Management Network (TMN) and the Management Framework for Open System Interconnection (OSI). While the TMN defines management functions from a network perspective, this Recommendation describes the management functions from the subscriber perspective and provides for remote user management functions.

1.1 *Scope*

This series of Recommendations will provide for a common approach for management communications to support procedures used by a remote maintenance centre, internal or external to the network and those initiated locally.

These Recommendations deal with the specification of the following items:

a) the specification of a Management Architecture and identification of communications paths;

b) the specification of management functionality to be provided at the ISDN user-network interface;

c) the specification of an information exchange protocol for the exchange of management information between two peer system management application entities (SMAE);

d) the specification of primitives between the Management Application process (user) and the SMAE (i.e., the primitives at the systems management service interface (SMSI));

e) the specification of service primitives between the SMAE service element and the next lower layer service elements (i.e., primitives at the presentation layer service access point (PSAP));

f) the specification of a convergence function that may be required to permit the direct access of the SMAE service elements to services provided by layer 3 (i.e., the primitives at the network layer service access point (NSAP)).

1.2 *Field of application*

The protocols and procedures described in these Recommendations provide the means to support management functions at the ISDN user-network interface. Management activities that manage network services, operations such as network resource configuration, routing

information and maintenance activities shall be supported by the functions and protocols defined in these Recommendations. In particular these management functions should be able to support specific requirements such as those defined in the I.60-Series of Recommendations (Subscriber Access and Installation Maintenance). These protocols make it possible to control loopbacks and diagnostic tests, initiate and terminate event reporting and to exchange management information across the ISDN user-network interface, i.e., between equipment connected to the SB/FT reference points.

The physical layer signals in the digital transmission section which are used to control maintenance functions are outside the scope of this Recommendation.

The protocols can be used on the D Channel of both the basic and primary rate interface structures and across both reference points S and T. The higher layer protocols can also be used on other ISDN channels and services.

The protocols and procedures described in these Recommendations take into account that interactions with the TMN will occur. It is, therefore, desirable that the services and protocols to be used to support access management are aligned, wherever possible, with those to be defined for the TMN and OSI management.

2 Categories of management information exchange

Management information exchanges may be categorized into the following three categories:

a) Event notification: information transfer initiated by one system reporting instantaneously the occurrence of an event (e.g., a fault occurrence) to another system.

b) Data transfer: information exchange initiated by one system in order to get management-related information from another system. These exchanges follow the ‘‘request followed by response’’ paradigm.

c) Control information: information exchanges which are of an executive nature, where one system requests that an action be performed by another system (e.g., for test access and downloading of parameters).

3 Management functions

Management functions may be classified in accordance with fields of application. The following major functions have been identified:

a) Fault management

— Maintenance functions

— Fault tracing

— Spontaneous error reporting

— Error threshold alarm reporting

— Continuous monitoring

— Diagnostic testing

— Resource (re)initialization

— Confidence testing

— Resource identification

— Trouble isolation.

b) Configuration management

— Routing changes

— Data base changes

— Equipment identification

— Network/equipment reconfiguration.

c) Accounting management

— Reporting of billing data.

d) Performance management

— Collecting and reporting of traffic data

— Performance monitoring

— Applying controls.

e) Security management.

4 Management reference models

4.1 *Communications path model*

Figure 1/Q.940 shows the entities which may contain System Management Entities (SME) which may require capability to communicate. System Management Entities may be located in the local exchanges, subscriber installations, remote management centres or network management centres.

The management functions supported by the various systems may differ depending on system requirements and may vary between different networks. However, the communications facilities provided by the systems management entities should be as common as possible.

The scope of this Recommendation covers those functions and protocols that have immediate impact on the user-network interface.

The system management entities may be in a TE, NT2 or management service provider. Although communication between any two management entities may be possible in the model, it does not imply that information held at a particular management entity is available to all other management entities. Security mechanisms may be used to restrict access to the information.

Figure 1/Q.940, p.

Figure 1/Q.940 shows that three types of management communications can be accommodated:

a) TE (or Remote Management Centre) < > TE (1< >2);

b) TE < > Network Management Function (1< >3);

c) TE < > Network Management Function < > TE (1 < > 3 < >2);

Types a) and b) are direct peer communication. In type c), the TE requests the Network Management Entity to act as an agent which then, on behalf of the requesting TE, communicates with another TE.

4.1.1 *Secure access to management and maintenance functions*

To facilitate maintenance procedures and fault sectionalization, maintenance entities located in different management domains may communicate. However, since management and maintenance information is of critical importance to system integrity, access to management functions and information is subject to prior authorization and security restrictions upon access.

The security restrictions are normally enforced by the recipient of the management information but may be enforced by the originator independently of any security imposed by the recipient. The security measures may include requirements for peer-entity authentication.

The use of adequate security mechanisms is especially important in the case of a network since many users may be affected by unauthorized access.

Whenever system management communication crosses an S or T reference point, the requirement for access authorization must be presumed.

Note — This does not preclude implicit actions on layer management parameters as specified within the relevant signalling protocols, e.g., Recommendations Q.921 and Q.931. These actions are, however, beyond the scope of this Recommendation.

4.2 *System management entity*

Figure 2B/FQ.940 shows the internal structure of the SME.

4.2.1 *System management application entity (SMAE)*

The SMAE is an application layer entity that supports system management functions. The SMAE is responsible for communication with peer systems.

The function of the SMAE is to provide the communications necessary to make a system management accessible to another SMAP. It is not necessary for the SMAE to be provided if only local system management is required.

4.2.2 **system management application process (SMAP)**

An SMAP is an application process of a system performing management functions. The SMAP controls the SMAE, and includes the Management Information Base (MIB) and may include one or more managers providing various functionalities.

4.2.3 **management information base (MIB)**

The MIB is the repository of all information relevant to the operation of a system. Both the SMAP and Layer Management Entities (LME) have access to the MIB.

4.2.4 **layer management entity (LME)**

The LME is that part of a Layer Entity which manages resources and parameters residing in its layer protocol entity.

4.2.5 **protocol entity (PE)**

The PE is that part of a layer entity which is dedicated to peer-to-peer communications. A layer PE provides services to the next upper layer and uses services of the next lower layer.

Figura 2/Q.940, p.

It should be noted that this model presently permits communication between peer management processes either by attaching to a Presentation Layer Access Point (PSAP) or by attaching directly to the Network Layer Service Access Point (NSAP). A convergence function may be provided as an alternative to the full seven layer OSI Reference Model (as specified in Recommendation X.200) to accommodate simple terminals that may be used in the ISDN environment. If provided, the functions will be kept to a minimum, i.e., the OSI layer services lost by elimination of layers 4-6 will not be recovered by the convergence function. Therefore, the use of all seven layers is to be preferred. This has the consequence that ‘‘convergence functions’’ may need to be specified.

4.2.6 *Management information protocol (MIP)*

The Management Information Protocol provides the support for information exchange between peer SMAEs.

4.3 *Managed objects: a hierarchical object model*

4.3.1 *Definitions*

4.3.1.1 **managed object**

A managed object is a collection of data objects and telecommunications or information processing resources that may be managed by means of the management protocol specified in this Recommendation.

4.3.1.2 A **data object** is an object that is the direct recipient of an action or generator of an event report.

4.3.2 *Hierarchical object model*

The maintenance functions are described as asymmetric functions using symmetrical communications paths. A maintenance activity is always started by an Invoker who is asking an Executor to manipulate event reports or data objects. These can be classified as belonging to individual managed objects. Each elementary operation that will have to access or refer to data objects will identify these by specifying first the managed object to which they belong and then identifying them within the managed object.

A hierarchical object model is defined that allows access to any individual data object in a simple way. When a given managed object may be duplicated, an instance identifier will help to resolve the ambiguity.

As an example, the model for user-network ISDN access interface is represented by the hierarchical tree of Figure 3/Q.940.

Figure 3/Q.940, p.

The parameters and event reports pertaining to a particular managed object can then be defined implicitly within the managed object. Some managed objects may be empty when no data object is identified within them. In this case they are only present as an indication of a hierarchical level.

It has to be noted that the ISDN user-network access interface model only contains managed objects that belong to the network access functions, i.e., that are involved in the provision of the required bearer service (signalling and lower layer protocols on the bearer channels). The protocols that are not involved in the provision of the bearer service are excluded from this model as they belong to the application part.

Note — The identity of an object at the executing end may not be known to the Invoker when it requests a maintenance action at the remote end of a connection. In this case the Executor will be able to identify the object by the context of the connection path used to convey the maintenance request.

As an example, remote maintenance may be required on an existing B Channel connection. The channel identity is only locally significant at each end. The maintenance request must be transmitted over the signalling connection that is used to control the B Channel associated with the existing call. The identity of the B Channel will be implied by the signalling connection used to convey the maintenance request.

5 Management structure and activities

This section considers the specific structure and activities of management in terms of system management, layer management and protocol processing for management purposes.

5.1 *System management*

This section introduces the concept of system management, its boundaries and other structures and activities related to management.

5.1.1 *Introduction*

The scope of system management is described in terms of the bounds of the SMAP. The boundaries show where the SMAP ends and other objects (either inside or outside the system) begin. The boundaries provide a sense of the relationship of the SMAP to other objects and therefore a sense of the SMAP scope.

5.1.2 *System management boundaries*

The boundaries of the SMAP are shown in Figure 4/Q.940.

Figure 4/Q.940, p.

This Figure shows the relationship between the SMAP and two other major components. The Communications component contains the seven layers of the reference model. The people and software component contains the peopleB/Fsoftware in the local environment that use the local systems manager.

The SMAE is the system management application entity, and (N)-LME represents the layer managers in the system.

5.1.2.1 *Local interface*

The local interface is located between the SMAP and the people and software that request services from the SMAP. Service requestB/Fresponses pass through this boundary to invoke one or more system management functions. Local interfaces, when present, are beyond the scope of this Recommendation.

5.1.2.2 *Layer management service interface (LMSI)*

The Layer Management Service Interface is the boundary between the SMAP and the individual layer management [(N)-LMEs]. Data and control information pass through this boundary. The boundary provides a way for each layer manager to gain access to parameters within the scope of that layer. This service interface is not subject to standardization.

5.1.2.2.1 *From system management to layer management*

The boundary between system management and (N)-layer management supports the flow from system management to layer management of:

1) requests to read, set, and perform actions with respect to various values, counters, statuses, etc., within a given layer;

2) response to inquiries made by an (N)-layer management entity upon the system management function;

3) data from the (N)-layer management of other systems.

5.1.2.2.2 *From layer management to system management*

The boundaries between system management and (N)-layer management supports the flow from (N)-layer management to system management of:

1) responses to read, set and request for action that came from system management;

2) request to send data to (N)-layer management in another system;

3) requests to place data into the Management Information Base;

4) requests to obtain information from the Management Information Base.

5.1.2.3 **system management service interface (SMSI)**

The System Management Service Interface is the boundary between the SMAP and the SMAE. The SMAE is a type of application entity which communicates system management messages to its peer SMAE in another system. Data and control information to and from the SMAE pass through this boundary. A service definition defines this boundary, and this service boundary defines system management.

5.1.3 *System management functions*

The responsibilities of system management are considered from two points of view:

a) Local system responsibilities (included for completeness of description):

— to initiate the (N)-layer manager for each layer, upon system activation;

— to serve as the manager of information that is common to several layers or that is supplied externally.

b) Communications responsibilities:

— to provide support for the exchange of information between the (N)-LMEs of a single layer so that the (N)-LMEs do not need to provide separate protocols for such exchanges;

— to coordinate the activities of the various SMAPs within telecommunication networks and subscriber installations.

5.1.4 *Relationship to (N)-layer management*

System management provides the only vehicle for the exchange of information between layers. Direct communication of management information between layers is deliberately precluded in the reference model to prevent inter-layer dependencies from occurring.

Since inter-layer exchanges of information will have to occur (i.e., error statistics), system management has been designated as the vehicle through which this exchange will occur. Each layer will have defined sets of information it may make known or will need to acquire.

System management implements the means of acquiring and disseminating this information. This may require activities on the part of system management that span several systems.

System management maintains the MIB and provides the support of (N)-LME access to the MIB.

5.1.5 *Relationship to the Management Information Base*

The SMAP is responsible for the MIB and provides authorized access to the MIB across the system boundaries.

5.2 *Layer management*

This section introduces the concept of layer management and its relationships to other entities.

5.2.1 *Scope*

In keeping with the general principle that each layer is independent of all others, each layer has its own management functions. These layer management functions are described in this Recommendation as the (N)-LME.

The role of the (N)-LME is threefold. Firstly, it serves to coordinate the activities of the (N)-entities within the layer. Secondly, it serves as the ''window'' to system management for the entities within the layer. Thirdly, in conjunction with both system management and its peer LMEs it manages the layer.

The (N)-LMEs are restricted to activities within an (N)-layer. The (N)-LME must not interact directly with a layer manager of any other layer.

5.2.2 *Relationship to (N)-entities which operate protocols*

The (N)-LME is charged with coordinating the activities and relationships of various (N)-entities which operate the protocols within the layer.

The (N)-LME is responsible for accessing the MIB on behalf of the (N)-entities. It will access the MIB to retrieve external parameters that the (N)-entity will need to operate, and to store and retrieve operating data that is in external storage contained within the scope of the peer management entity. The (N)-LME is also the focus for control of the (N)-entities by system management.

5.2.3 *Relationship between peer (N)-LMEs*

The (N)-LMEs will frequently need to exchange information. This exchange ordinarily will be accomplished through the peer SMAPs. However, in some cases, layer management protocols are necessary. These cases are limited to the following:

1) where the exchange of information, or the circumstances under which such information might be exchanged would necessarily interfere with the support of the SMAE by the lower layers: for example, loop testing at layer 1 might be supported by a layer 1 management protocol, and exchange of routing information might be supported by a layer 3 management protocol;

2) where layer management protocols already exist; for example, see Recommendation Q.921.

In no event may a layer management protocol interact directly with any other layer. System management provides the only means for data transfer.

5.2.4 *Relationship to system management*

The (N)-LMEs rely upon services from system management for three purposes. These are to provide communication for intra-layer management activities, to coordinate inter-layer management activities and to serve as a general repository for management information.

As system management is the supervisor for any action on layer management, the service requestB/Fresponse for external action (e.g., parameter manipulation, statistic gathering, etc.,) will use the SMAP as defined in § 6.1.

5.3 *Protocol processing for management purposes*

5.3.1 *Scope*

On occasion, the (N)-entities do participate in the management process. This occurs when the protocol has embedded within itself information that must be made known to other entities and when events occur that must be made known to other entities.

5.3.2 *Relationship of (N)-entities to (N)-LMEs*

The (N)-entities rely upon the (N)-LME to provide coordination between the various (N)-entities in the (N)-layer, and access to data and services that come from outside the (N)-layer. There is, therefore, a flow of control information between the (N)-entities and the (N)-LME.

Since the (N)-entities exist independently of the other (N)-entities within the (N)-layer, they are dependent upon the (N)-LME to coordinate activities between the various (N)-entities within the sub-system. As an example the (N)-entities rely upon the (N)-LME to determine when requests for connection are being made to establish the association between the connection request at a connection endpoint and the (N)-entity. The (N)-LME also controls the instantiation of (N)-entities at the time of connection requests.

6 Overview of services required by the SMAP

6.1 *High layer context management*

When the two SMAPs are involved in a management dialogue, they may want to establish a context that will be maintained during the life of the dialogue. In this sense two SMAPs typically work in a connection-oriented mode. The SMAE will provide services that will allow it to work in connection-oriented mode by providing the capability to establish and release associations between peer applications.

These services are to be described further in future Recommendations.

The use of a connectionless service is for further study.

6.2 *Definition of a set of generic functions*

As presented in § 5, management covers a large spectrum of applications. These applications may be implemented by dedicated SMAPs that can make use of a reduced set of generic functions. The generic functions are listed hereafter with examples for their use:

— Trigger an action (e.g., activate or deactivate loopbacks or internal tests);

— Event report (e.g., error reporting, alarm reporting);

— Get attributes (e.g., cumulative error counters, get parameter values);

— Set attributes (e.g., set or modify parameters, thresholds, etc.);

— Create and delete managed objects (e.g., create a routing table).

The SMAE provides facilities to allow the generic functions to be communicated between SMAPs.

7 Addressing for information exchange

The information flow takes place between two SMAPs and the originator must be able to address the destination SMAP.

Depending upon the location of the communicating SMAPs different addressing schemes may apply:

1) Explicit Addressing. In this case the remote entity is explicitly addressed by its ISDN address.

2) Implicit Addressing. Implicit addressing relies on mechanisms other than an explicit address in the maintenance message to identify the recipient of the information.

For system management two cases of implicit addressing may be identified:

a) permanent connections;

b) hot line service.

8 Terminal selection

In addition to the normal ISDN addressing mechanisms the maintenance procedures which require actions to perform to particular user equipment require the existence of an identification method that allows access to the unique piece of user equipment to be maintained.

Selection of a unique terminal is based on compatibility checking of various parameters. Compatibility is determined first on the basis of the ISDN address and then on the basis of service information (bearer capability, high layer compatibility, etc.). The service information alone is adequate to provide unique identification if a single unit of equipment satisfies this requirement.

When several TEs connected to the same access, sharing one ISDN address, provide the same functionality, and neither the NSAP nor service information are sufficient, then a unique equipment identifier must be used.

9 Access control

In many cases information accessible through the management function may be private or a management action may result in taking the equipment out of service. Access security to management and maintenance functions must, therefore, be provided.

Access controls may be applied both to the call establishment phase of the maintenance call and also within individual maintenance transactions.

The use of Calling Line Identity provides one method by which maintenance calls can be screened. Further access right discrimination can be performed on the basis of message type in which the management information is carried. Each message type may have its own implied access rights.

Additionally, specific access control can be performed on the basis of an explicit access control parameter. This parameter has the following characteristics:

1) access control mechanisms are defined as parameters of the primitives passed between system management and the service provider;

2) use of access control parameters is optional;

3) in addition to meeting compatibility requirements, management calls must also satisfy the access control requirements;

4) access control information may be encrypted.

Blanc

MONTAGE: PAGE 436 = PAGE BLANCHE