
Source: SA5 (Telecom Management)
Title: 2 Rel-4 & Rel-5 CRs 32.111-4 (Fault Management; Alarm Integration Reference Point; Part 4 CMIP SS) Correction of errors and ambiguities in the Parameter Mapping Tables and ASN.1 Definitions
Document for: Approval
Agenda Item: 7.5.3

Doc-1 st -Level	Spec	CR	R	Phase	Subject	Cat	Ver Cur	Ver New	Doc-2 nd -Level	Workitem
SP-020283	32.111-4	006	-	Rel-4	Correction of errors and ambiguities in the Parameter Mapping Tables and ASN.1 Definitions	F	4.2.0	4.3.0	S5-026245	OAM-FM
SP-020283	32.111-4	007	-	Rel-5	Correction of errors and ambiguities in the Parameter Mapping Tables and ASN.1 Definitions	A	5.0.0	5.1.0	S5-026246	OAM-NIM

CHANGE REQUEST

⌘ **32.111-4 CR 006** ⌘ rev **-** ⌘ Current version: **4.2.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Correction of errors and ambiguities in the Parameter Mapping Tables and ASN.1 Definitions		
Source:	⌘ SA5		
Work item code:	⌘ OAM-FM	Date:	⌘ 24/05/2002
Category:	⌘ F	Release:	⌘ REL-4
	<i>Use one of the following categories:</i>		<i>Use one of the following releases:</i>
	F (correction)	2 (GSM Phase 2)	
	A (corresponds to a correction in an earlier release)	R96 (Release 1996)	
	B (addition of feature),	R97 (Release 1997)	
	C (functional modification of feature)	R98 (Release 1998)	
	D (editorial modification)	R99 (Release 1999)	
	Detailed explanations of the above categories can be found in 3GPP TR 21.900.		REL-4 (Release 4)
			REL-5 (Release 5)

Reason for change:	⌘ The mapping tables for the notification parameters contain some errors and ambiguities. In the ASN.1 definition for AlarmChoice one enumeration is missing.
Summary of change:	⌘ The errors in the mapping tables are corrected and the missing enumeration is added to the definition of AlarmChoice.
Consequences if not approved:	⌘ The CMIP SS (32.111-4) does not properly reflect the Alarm IRP IS (32.111-2).

Clauses affected:	⌘ 4.7.4, 4.7.5, 5.4, 6
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications ⌘ <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications
Other comments:	⌘

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/>. For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

4.7.4 Mapping of Notifications

Table 10 maps the Notifications defined in the Information Service of the Alarm IRP to the equivalent Notifications of the CMIP solution set for the Alarm IRP. The CMIP Notifications are qualified as Mandatory (M) or Optional (O).

Table 10: Mapping of Notifications

Notifications of Information Services of the Alarm IRP	Equivalent Notifications of the CMIP solution set for the Alarm IRP	Qualifier
notifyNewAlarm	environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	M
notifyChangedAlarm	<u>notifyClearedAlarm</u> <u>notifyNewAlarm</u> which are in turn mapped into environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	O
notifyClearedAlarm	environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	M
notifyAckStateChanged	environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	M
notifyAlarmListRebuilt	<u>notifyAalarmListRebuilt</u>	M
notifyComments	<u>notifyComments</u>	O

4.7.5 Mapping of Parameters of each notification

The notifications defined in [9] (Alarm IRP: Information Services) have a set of parameters that are common to all the notifications (see [10]: Notification IRP:CMIP Solution Set). Such common set of parameters are:

ManagedObjectClass, ManagedObjectInstance, EventTime, NotificationType, NotificationId.

In the CMIP Solution Set, all the notifications originated within the Agent are reported to the Managers by means of the CMISE "M-EVENT-REPORT" primitive, which is implemented by means of the "m-EventReport OPERATION" (see [2] and[3]). The argument of m-EventReport OPERATION is defined in [3] as follows:

```

EventReportArgument ::= SEQUENCE {
    managedObjectClass      ObjectClass,
    managedObjectInstance   ObjectInstance,
    eventTime                [5] IMPLICIT GeneralizedTime OPTIONAL,
    eventType                [EventTypeId,
    eventInfo                [8] ANY DEFINED BY eventType OPTIONAL
}
    
```

where eventInfo is further specified, for each specific notification, by means of specific GDMO/ASN1 definitions.

~~Therefore the first four parameters of the notification header are mapped to the first four fields of the EventReportArgument. The fifth parameter of the notification header is mapped to the eventInfo field of the EventReportArgument, together with all the other (not common) parameters of the notification.~~

In the following tables, for the notifications defined in [9], all the parameters (but the common ones) are mapped to their corresponding elements of the CMIP SS notificationsequivalents. Note that the parameter mapping for the notification notifyChangedAlarm is not given. This is because in the CMIP SS the notifications notifyClearedAlarm and notifyNewAlarm are emitted instead of the notification notifyChangedAlarm..

The IS parameter SsystemDN defined in [9] (Alarm IRP: Information Services) is conditional and not used in this the CMIP SSolution Set.

Except for the notification notifyComments the IS parameter alarmType has no direct CMIP SS equivalent. Instead the value of this parameter is reflected by the type of the emitted notification. More specifically

- If the event type is equal to 'Communication Alarm' the notification communicationsAlarm is emitted.
- If the event type is equal to 'Processing Error Alarm' the notification processingErrorAlarm is emitted.
- If the event type is equal to 'Environmental Alarm' the notification environmentalAlarm is emitted.
- If the event type is equal to 'Quality of Service Alarm' the notification qualityofServiceAlarm is emitted.
- If the event type is equal to 'Equipment Alarm' the notification equipmentAlarm is emitted.

Also the IS parameter alarmId is not mapped directly to a parameter in the CMIP SS. This is not required because an alarm is identified unambiguously by the notification identifier of the notification reporting the alarm the first time and the instance of the managed object emitting this notification. Notifications referring to an alarm already reported (e. g. notifyClearedAlarm, notifyAckStateChanged, notifyComments) do so by specifying in the M-EVENT REPORT parameter 'Event information': correlatedNotifications (X.721 [4] and X.733 [5]) the notification identifier of the notification having reported the new alarm and, if required, the instance of the object having emitted this notification.

Table 11: Mapping of Parameters of "notifyNewAlarm" and "notifyClearedAlarm"

IS Parameter Name Notification parameters of Information Services	CMIP SS Equivalent Notification equivalences	Qualifier
<u>objectclass</u>	M-EVENT REPORT parameter 'Managed object class'	M
<u>objectInstance</u>	M-EVENT REPORT parameter 'Managed object instance'	M
<u>notificationId</u>	M-EVENT REPORT parameter 'Event information': <u>notificationIdentifier</u>	M
<u>eventTime</u>	M-EVENT REPORT parameter 'Event time'	M
<u>systemDN</u>	--	--
<u>notificationType</u>	M-EVENT REPORT parameter 'Event type'	M
<u>notificationId</u>	<u>notificationIdentifier</u> (note 1)	M
<u>probableCause</u>	M-EVENT REPORT parameter 'Event information': <u>probableCause</u>	M
<u>specificProblems</u>	M-EVENT REPORT parameter 'Event information': <u>specificProblems</u>	O
<u>perceivedSeverity</u>	M-EVENT REPORT parameter 'Event information': <u>perceivedSeverity</u>	M
<u>alarmType</u>	The semantics of this parameter is conveyed by the <u>notification type</u> .	--
<u>backedUpStatus</u>	M-EVENT REPORT parameter 'Event information': <u>backedUpStatus</u>	O
<u>backUpObject</u>	M-EVENT REPORT parameter 'Event information': <u>backUpObject</u>	O
<u>trendIndication</u>	M-EVENT REPORT parameter 'Event information': <u>trendIndication</u>	O
<u>thresholdInfo</u>	M-EVENT REPORT parameter 'Event information': <u>thresholdInfo</u>	O
<u>correlatedNotifications</u>	M-EVENT REPORT parameter 'Event information': <u>correlatedNotifications</u>	O
<u>stateChangeDefinition</u>	M-EVENT REPORT parameter 'Event information': <u>stateChangeDefinition</u>	O
<u>monitoredAttributes</u>	M-EVENT REPORT parameter 'Event information': <u>monitoredAttributes</u>	O
<u>proposedRepairActions</u>	M-EVENT REPORT parameter 'Event information': <u>proposedRepairActions</u>	O
<u>additionalText</u>	M-EVENT REPORT parameter 'Event information': <u>additionalText</u>	O
<u>alarmId</u>	-- (note 2)	--

NOTE 1: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302.
 NOTE 2: In the CMIP Solution Set the alarmId is not used. In the CMIP Solution Set the alarm notifications are univocally identified by means of notificationIdentifier and managedObjectInstance.

Table 11: Mapping of Parameters of "notifyClearedAlarm"

IS Parameter Name	CMIP SS Equivalent	Qualifier
<u>objectclass</u>	M-EVENT REPORT parameter 'Managed object class'	M
<u>objectInstance</u>	M-EVENT REPORT parameter 'Managed object instance'	M
<u>notificationId</u>	M-EVENT REPORT parameter 'Event information': <u>notificationIdentifier</u>	M
<u>eventTime</u>	M-EVENT REPORT parameter 'Event time'	M
<u>systemDN</u>	--	--
<u>notificationType</u>	M-EVENT REPORT parameter 'Event type'	M
<u>probableCause</u>	M-EVENT REPORT parameter 'Event information': <u>probableCause</u>	M
<u>perceivedSeverity</u>	M-EVENT REPORT parameter 'Event information': <u>perceivedSeverity</u>	M
<u>alarmType</u>	The semantics of this parameter is conveyed by the <u>notification type</u> .	--
<u>correlatedNotifications</u>	M-EVENT REPORT parameter 'Event information': <u>correlatedNotifications</u>	O
<u>alarmId</u>	M-EVENT REPORT parameter 'Event information': <u>correlatedNotifications</u>	M

Table 12: Mapping of Parameters of 'notifyAckStateChanged'

IS Parameter Name Notification parameters of Information Services	CMIP SS Equivalent Notification equivalences	Qualifier
objectclass	M-EVENT REPORT parameter 'Managed object class'	M
objectInstance	M-EVENT REPORT parameter 'Managed object instance'	M
notificationId	M-EVENT REPORT parameter 'Event information': notificationIdentifier	M
eventTime	M-EVENT REPORT parameter 'Event time'	M
systemDN	--	--
notificationType	M-EVENT REPORT parameter 'Event type'	M
notificationId	notificationIdentifier (note 1)	M
probableCause	M-EVENT REPORT parameter 'Event information': probableCause	M
specificProblems	specificProblems	Ø
perceivedSeverity	M-EVENT REPORT parameter 'Event information': perceivedSeverity	M
alarmType	The semantics of this parameter is conveyed by the notification type.	--
alarmId	M-EVENT REPORT parameter 'Event information': correlatedNotifications --- (note-2)	--
ackTime	M-EVENT REPORT parameter 'Event information':	M
ackState	additionalInformation	M
ackUserId		M
ackSystemId		O
NOTE 1: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302.		
NOTE 2: In the CMIP Solution Set the alarmId is not used. In the CMIP Solution Set the alarm notifications are univocally identified by means of notificationIdentifier and managedObjectInstance.		

Table 13: Mapping of Parameters of 'notifyAlarmListRebuilt'

Notification parameters of Information Services	CMIP Notification equivalents	Qualifier
notificationId	notificationIdentifier (note)	M
reason	reason	M
objectClass	rebuiltObjectClass	M
objectInstance	rebuiltObjectInstance	M
NOTE: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302.		

IS Parameter Name	CMIP SS Equivalent	Qualifier
objectclass	M-EVENT REPORT parameter 'Event information': rebuiltObjectClass	M
objectInstance	M-EVENT REPORT parameter 'Event information': rebuiltObjectInstance	M
notificationId	M-EVENT REPORT parameter 'Event information': notificationIdentifier	M
eventTime	M-EVENT REPORT parameter 'Event time'	M
systemDN	--	--
notificationType	M-EVENT REPORT parameter 'Event type'	M
reason	M-EVENT REPORT parameter 'Event information': reason	M

Table 14: Mapping of Parameters of 'notifyComments'

IS Parameter Name	Notification parameters of Information Services	CMIP SS Equivalent	Notification equivalents	Qualifier
notificationId		notificationIdentifier	(note 1)	M
objectClass		M-EVENT REPORT parameter 'Event information':	alarmedObjectClass	M
objectInstance		M-EVENT REPORT parameter 'Event information':	alarmedObjectInstance	M
notificationId		M-EVENT REPORT parameter 'Event information':	notificationIdentifier	M
eventTime		M-EVENT REPORT parameter 'Event information':	alarmEventTime	M
systemDN		--		--
notificationType		M-EVENT REPORT parameter 'Event type'	eventType	M
alarmType		M-EVENT REPORT parameter 'Event information':	alarmType	M
probableCause		M-EVENT REPORT parameter 'Event information':	alarmProbableCause	M
perceivedSeverity		M-EVENT REPORT parameter 'Event information':	alarmPerceivedSeverity	M
comments		M-EVENT REPORT parameter 'Event information':	comments	M
alarmId		M-EVENT REPORT parameter 'Event information':	correlatedNotifications --- (note 2)	--
NOTE 1: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302. NOTE 2: In the CMIP Solution Set the alarmId is not used. In the CMIP Solution Set the alarm notifications are univocally identified by means of notificationIdentifier and managedObjectInstance.				

5.4 Notifications

5.4.1 notifyAlarmListRebuilt (M)

alarmListRebuilt **NOTIFICATION**
BEHAVIOUR

alarmListRebuiltBehaviour;

WITH INFORMATION SYNTAX

TS32-111-4TypeModule .AlarmListRebuiltInfo;

REGISTERED AS { ts32-111AlarmNotification 1};

alarmListRebuiltBehaviour **BEHAVIOUR**
DEFINED AS

"This notification is used by the Agent to inform the NM that the alarm list has been rebuilt.

The 'Event Information' field contains the following data:

- *notificationIdentifier*
This ITU-T X.721 standardised parameter, together with MOI (Managed Object Instance), unambiguously identifies this notification.
- *rebuiltObjectClass*
This parameter carries the IRPAgent MOC when the entire AlarmList has been rebuilt. It carries a different MOC when the AlarmList has been partially rebuilt.
- *rebuiltObjectInstance*
This parameter carries DN of the IRPAgent when the entire AlarmList has been rebuilt. It carries the DN of another MOI when the AlarmList has been partially rebuilt and only the MOIs subordinate of this rebuilt MOI may be affected by this partial rebuilt.
- *reason*
The parameter indicates the reason for alarm list rebuilding (if applicable)."

5.4.2 notifyComments (O)

notifyComments **NOTIFICATION**
BEHAVIOUR

notifyCommentsBehaviour;

WITH INFORMATION SYNTAX

TS32-111-4TypeModule .NotifyComments;

REGISTERED AS { ts32-111AlarmNotification 2};

notifyCommentsBehaviour **BEHAVIOUR**
DEFINED AS

"This notification is used by the Agent to inform the NM that one or more comments have been associated to one alarm.

The 'Event Information' field contains the following data:

- *alarmedObjectClass*
This parameter specifies the object class representing the resource that raised the alarm to which the comment was attached.
- *alarmedObjectInstance*
This parameter specifies the object instance representing the resource that raised the alarm to which the comment was attached.
- *notificationIdentifier*

This parameter specifies the notification identifier (ITU-T X.733 [5]), which, together with the instance of the object emitting this notification, unambiguously identifies this notification.

- *alarmEventTime*
This parameter specifies the time when the alarm, to which the comment was attached, was first raised by the alarmed resource.
- *alarmType*
This parameter specifies the event type of the notification that reported the alarm to which the comment was attached.
- *alarmProbableCause*
This parameter specifies the probable cause (ITU-T X.733 [5]) of the alarm to which the comment was attached.
- *alarmPerceivedSeverity*
This parameter specifies the perceived severity (ITU-T X.733 [5]) of the alarm to which the comment is attached.
- *comments*
This parameter carries the text of the comment.

□alarmedObjectClass: defined in ITU-T X.710 [2] and X.711 [3]
 □alarmedObjectInstance: defined in ITU-T X.710 [2] and X.711 [3]
 □notificationIdentifier: This ITU-T X.721 standardised parameter, together with MOI (Managed Object Instance), unambiguously identifies this notification.
 □alarmEventTime: defined in ITU-T X.721
 □alarmType: the eventType of the alarm to which this comment is associated.
 □alarmProbableCause: defined in ITU-T X.721
 □alarmPerceivedSeverity: defined in ITU-T X.721
 □comments: the text of the comment.
 ";

6 ASN.1 definitions for Alarm IRP

```
TS32-111-4TypeModule {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-Operation-
Maintenance(3) ts-32-111(111) part4(4) informationModel(0) asn1Module(2) version1(1)}
```

```
DEFINITIONS IMPLICIT TAGS ::=
BEGIN
--EXPORTS everything
IMPORTS
```

```
NotificationIdentifier, Destination, EventTime, ProbableCause, PerceivedSeverity
FROM Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1}
```

```
AlarmInfo
FROM Notification-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 2}
```

```
CMISFilter, ObjectInstance, ObjectClass, EventTypeId
FROM CMIP-1 {joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3)};
```

```
baseNodeUMTS OBJECT IDENTIFIER ::= { itu-t (0) identified-organization (4) etsi (0) mobileDomain
(0)
umts-Operation-Maintenance (3) }
ts32-111Prefix OBJECT IDENTIFIER ::= { baseNodeUMTS ts-32-111(111)}
ts32-111Part4 OBJECT IDENTIFIER ::= { ts32-111Prefix part4(4)}
```

```

ts32-111-4InfoModel OBJECT IDENTIFIER ::= { ts32-111Part4 informationModel(0)}

ts32-111AlarmObjectClass      OBJECT IDENTIFIER ::= { ts32-111-4InfoModel managedObjectClass(3)}
ts32-111AlarmPackage         OBJECT IDENTIFIER ::= { ts32-111-4InfoModel package(4)}
ts32-111AlarmParameter       OBJECT IDENTIFIER ::= { ts32-111-4InfoModel parameter(5)}
ts32-111AlarmAttribute       OBJECT IDENTIFIER ::= { ts32-111-4InfoModel attribute(7)}
ts32-111AlarmAction          OBJECT IDENTIFIER ::= { ts32-111-4InfoModel action(9)}
ts32-111AlarmNotification     OBJECT IDENTIFIER ::= { ts32-111-4InfoModel action(10)}

-- Start of 3GPP SA5 own definitions
AckErrorList ::= SET OF ErrorInfo
AlarmReference ::= SEQUENCE
{
  moi ObjectInstance OPTIONAL, -- absent if scope of uniqueness of notificationId is across IRPAgent
  notificationIdentifier NotificationIdentifier
}
AckOrUnackAlarms ::= SEQUENCE
{
  alarmReferenceList SET OF AlarmReference, -- ITU-T X.721
  ackUserId          UserId,
  ackSystemId       SystemId OPTIONAL
}
AckOrUnackAlarmsReply ::= SEQUENCE
{
  status              ErrorCauses,
  errorAlarmReferenceList AckErrorList
}
AckState ::= ENUMERATED
{
  acknowledged      (0),
  unacknowledged    (1)
}
AckTime ::= GeneralizedTime
AlarmChoice ::= ENUMERATED
{
  allAlarms          _____(0),
  allActiveAlarms    _____(1),
  allActiveAndAckAlarms (2),
  allActiveAndUnackAlarms (3),
  allClearedAndUnackAlarms (4),
  allUnackAlarms     _____(5)
}
AlarmsCountSummary ::= SEQUENCE
{
  activeAlarmsCount      INTEGER, -- this is the sum of criticalCount, majorCount,
  minorCount, warningCount
                                -- and indeterminateCount
  criticalCount          _____INTEGER,
  majorCount             INTEGER,
  minorCount             INTEGER,
  warningCount           INTEGER,
  indeterminateCount     INTEGER,
  clearedCount           _____INTEGER
}
AlarmListRebuiltInfo ::= SEQUENCE
{
  notificationIdentifier NotificationIdentifier, -- ITU-T X.721
  rebuiltObjectClass     ObjectClass,
  rebuiltObjectInstance  ObjectInstance,
  reason                 ErrorCauses
}
Comment ::= GraphicString
ErrorCauses ::= ENUMERATED
{
  noError (0), -- operation / notification successfully performed
  wrongFilter (1), -- the value of the filter parameter is not valid
  wrongAlarmAckState (2), -- the value of the alarmAckState parameter (e.g. getAlarmCount) is not
  valid
  ackPartlySuccessful (3), -- acknowledgment request partly successful
  unackPartlySuccessful (4), -- unacknowledgment request partly successful
  wrongAlarmReference (5), -- alarm identifier used in the alarm reference list not found (e.g. in
  case of acknowledgement request)
  wrongAlarmReferenceList (6), -- the alarm reference list (e.g. in case of acknowledgement
  request) is empty or completely wrong
  alarmAlreadyAck (7), -- alarm to be acknowledged is already in this state
  alarmAlreadyUnack (8), -- alarm to be acknowledged is already in this state

```

```

wrongUserId (9), -- the user identifier in the unacknowledgement operation is not the same as
in the previous acknowledgementAlarms request
wrongSystemId (10), -- the system identifier in the unacknowledgement operation is not the same as
in the previous acknowledgementAlarms request
alarmAckNotAllowed (11), -- current management system not allowed to acknowledge the alarm (e.g.
due to acknowledgement competence rules)
setCommentPartlySuccessful (12), -- the setComment action partly successful (e.g. some alarmId
are not in the alarmList)
unspecifiedErrorReason (255) -- operation failed, specific error unknown
}
ErrorInfo ::= SEQUENCE
{
    moi ObjectInstance OPTIONAL, -- absent if uniqueness of notificationIdentifier is across
IRPagent
    notificationIdentifier NotificationIdentifier, -- ITU-T X.721
    reason ErrorCauses
}
GeneralObjectId ::= INTEGER
GetAlarmCount ::= SEQUENCE
{
    alarmAckState AlarmChoice OPTIONAL,
    filter CMISFilter OPTIONAL -- ITU-T X.711
}
GetAlarmCountReply ::= SEQUENCE
{
    criticalCount INTEGER,
    majorCount INTEGER,
    minorCount INTEGER,
    warningCount INTEGER,
    indeterminateCount INTEGER,
    clearedCount INTEGER,
    status ErrorCauses
}
GetAlarmIRPVersionReply ::= SEQUENCE
{
    versionNumberList SupportedAlarmIRPVersions,
    status ErrorCauses
}
GetAlarmList ::= SEQUENCE
{
    alarmAckState AlarmChoice OPTIONAL,
    destination Destination, -- ITU-T X.721
    filter CMISFilter OPTIONAL -- ITU-T X.711
}
GetAlarmListReply ::= SEQUENCE
{
    alignmentId INTEGER,
    status ErrorCauses
}
GetNotificationProfileReply ::= SEQUENCE
{
    notificationNameProfile NotificationList,
    notificationParameterProfile ParameterListOfList,
    status ErrorCauses
}
GetOperationProfileReply ::= SEQUENCE
{
    operationNameProfile OperationList,
    operationParameterProfile ParameterListOfList,
    status ErrorCauses
}
IRPVersionNumber ::= GraphicString
NotificationList ::= SET OF NotificationName
NotificationName ::= GraphicString
NotifyComments ::= SEQUENCE
{
    alarmedObjectClass ObjectClass, -- ITU-T X.711
    alarmedObjectInstance ObjectInstance, -- ITU-T X.711
    notificationIdentifier NotificationIdentifier, -- ITU-T X.721
    alarmEventTime EventTime, -- ITU-T X.721
    alarmType EventTypeId, -- ITU-T X.711
    alarmProbableCause ProbableCause, -- ITU-T X.721
    alarmPerceivedSeverity PerceivedSeverity, -- ITU-T X.721

```

```
comments          SET OF Comment
}

OperationList ::= SET OF OperationName

OperationName ::= GraphicString

ParameterList ::= SET OF ParameterName

ParameterListOfList ::= SET OF ParameterList

ParameterName ::= GraphicString

SetComment ::= SEQUENCE
{
  alarmReferenceList  SET OF AlarmReference,
  commentUserId      UserId,
  commentSystemId    SystemId,
  commentText        Comment
}

SetCommentReply ::= SEQUENCE
{
  badAlarmReferenceList  SET OF ErrorInfo,
  status                 ErrorCauses
}

SystemId ::= GraphicString

SupportedAlarmIRPVersions ::= SET OF IRPVersionNumber
UserId ::= GraphicString

END -- of module TS32-111-4TypeModule
```

CHANGE REQUEST

⌘ **32.111-4 CR 007** ⌘ rev **-** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Correction of errors and ambiguities in the Parameter Mapping Tables and ASN.1 Definitions		
Source:	⌘ SA5		
Work item code:	⌘ OAM-NIM	Date:	⌘ 24/05/2002
Category:	⌘ A	Release:	⌘ REL-5
	Use <u>one</u> of the following categories: F (correction) A (corresponds to a correction in an earlier release) B (addition of feature), C (functional modification of feature) D (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

Reason for change:	⌘ The mapping tables for the notification parameters contain some errors and ambiguities. In the ASN.1 definition for AlarmChoice one enumeration is missing.
Summary of change:	⌘ The errors in the mapping tables are corrected and the missing enumeration is added to the definition of AlarmChoice.
Consequences if not approved:	⌘ The CMIP SS (32.111-4) does not properly reflect the Alarm IRP IS (32.111-2).

Clauses affected:	⌘ 4.7.4, 4.7.5, 5.4, 6		
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
Other comments:	⌘		

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/>. For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

4.7.4 Mapping of Notifications

Table 10 maps the Notifications defined in the Information Service of the Alarm IRP to the equivalent Notifications of the CMIP solution set for the Alarm IRP. The CMIP Notifications are qualified as Mandatory (M) or Optional (O).

Table 10: Mapping of Notifications

Notifications of Information Services of the Alarm IRP	Equivalent Notifications of the CMIP solution set for the Alarm IRP	Qualifier
notifyNewAlarm	environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	M
notifyChangedAlarm	<u>notifyClearedAlarm</u> <u>notifyNewAlarm</u> which are in turn mapped into environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	O
notifyClearedAlarm	environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	M
notifyAckStateChanged	environmentalAlarm ITU-T X.721 [4] equipmentAlarm ITU-T X.721 [4] qualityofServiceAlarm ITU-T X.721 [4] processingErrorAlarm ITU-T X.721 [4] communicationAlarm ITU-T X.721 [4]	M
notifyAlarmListRebuilt	<u>notifyAalarmListRebuilt</u>	M
notifyComments	<u>notifyComments</u>	O

4.7.5 Mapping of Parameters of each notification

The notifications defined in [9] (Alarm IRP: Information Services) have a set of parameters that are common to all the notifications (see [10]: Notification IRP:CMIP Solution Set). Such common set of parameters are:

ManagedObjectClass, ManagedObjectInstance, EventTime, NotificationType, NotificationId.

In the CMIP Solution Set, all the notifications originated within the Agent are reported to the Managers by means of the CMISE "M-EVENT-REPORT" primitive, which is implemented by means of the "m-EventReport OPERATION" (see [2] and[3]). The argument of m-EventReport OPERATION is defined in [3] as follows:

```

EventReportArgument ::= SEQUENCE {
    managedObjectClass      ObjectClass,
    managedObjectInstance   ObjectInstance,
    eventTime                [5] IMPLICIT GeneralizedTime OPTIONAL,
    eventType                [EventTypeId,
    eventInfo                [8] ANY DEFINED BY eventType OPTIONAL
}
    
```

where eventInfo is further specified, for each specific notification, by means of specific GDMO/ASN1 definitions.

~~Therefore the first four parameters of the notification header are mapped to the first four fields of the EventReportArgument. The fifth parameter of the notification header is mapped to the eventinfo field of the EventReportArgument, together with all the other (not common) parameters of the notification.~~

In the following tables, for the notifications defined in [9], all the parameters (but the common ones) are mapped to their corresponding elements of the CMIP SS notificationsequivalents. Note that the parameter mapping for the notification notifyChangedAlarm is not given. This is because in the CMIP SS the notifications notifyClearedAlarm and notifyNewAlarm are emitted instead of the notification notifyChangedAlarm..

The IS parameter SsystemDN defined in [9] (Alarm IRP: Information Services) is conditional and not used in this the CMIP SSolution Set.

Except for the notification notifyComments the IS parameter alarmType has no direct CMIP SS equivalent. Instead the value of this parameter is reflected by the type of the emitted notification. More specifically

- If the event type is equal to 'Communication Alarm' the notification communicationsAlarm is emitted.
- If the event type is equal to 'Processing Error Alarm' the notification processingErrorAlarm is emitted.
- If the event type is equal to 'Environmental Alarm' the notification environmentalAlarm is emitted.
- If the event type is equal to 'Quality of Service Alarm' the notification qualityofServiceAlarm is emitted.
- If the event type is equal to 'Equipment Alarm' the notification equipmentAlarm is emitted.

Also the IS parameter alarmId is not mapped directly to a parameter in the CMIP SS. This is not required because an alarm is identified unambiguously by the notification identifier of the notification reporting the alarm the first time and the instance of the managed object emitting this notification. Notifications referring to an alarm already reported (e. g. notifyClearedAlarm, notifyAckStateChanged, notifyComments) do so by specifying in the M-EVENT REPORT parameter 'Event information': correlatedNotifications (X.721 [4] and X.733 [5]) the notification identifier of the notification having reported the new alarm and, if required, the instance of the object having emitted this notification.

Table 11: Mapping of Parameters of "notifyNewAlarm" and "notifyClearedAlarm"

IS Parameter Name Notification parameters of Information Services	CMIP SS Equivalent Notification equivalences	Qualifier
objectclass	M-EVENT REPORT parameter 'Managed object class'	M
objectInstance	M-EVENT REPORT parameter 'Managed object instance'	M
notificationId	M-EVENT REPORT parameter 'Event information': notificationIdentifier	M
eventTime	M-EVENT REPORT parameter 'Event time'	M
systemDN	--	--
notificationType	M-EVENT REPORT parameter 'Event type'	M
notificationId	notificationIdentifier (note 1)	M
probableCause	M-EVENT REPORT parameter 'Event information': probableCause	M
specificProblems	M-EVENT REPORT parameter 'Event information': specificProblems	O
perceivedSeverity	M-EVENT REPORT parameter 'Event information': perceivedSeverity	M
alarmType	The semantics of this parameter is conveyed by the notification type.	--
backedUpStatus	M-EVENT REPORT parameter 'Event information': backedUpStatus	O
backUpObject	M-EVENT REPORT parameter 'Event information': backUpObject	O
trendIndication	M-EVENT REPORT parameter 'Event information': trendIndication	O
thresholdInfo	M-EVENT REPORT parameter 'Event information': thresholdInfo	O
correlatedNotifications	M-EVENT REPORT parameter 'Event information': correlatedNotifications	O
stateChangeDefinition	M-EVENT REPORT parameter 'Event information': stateChangeDefinition	O
monitoredAttributes	M-EVENT REPORT parameter 'Event information': monitoredAttributes	O
proposedRepairActions	M-EVENT REPORT parameter 'Event information': proposedRepairActions	O
additionalText	M-EVENT REPORT parameter 'Event information': additionalText	O
alarmId	-- (note 2)	--

NOTE 1: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302.
 NOTE 2: In the CMIP Solution Set the alarmId is not used. In the CMIP Solution Set the alarm notifications are univocally identified by means of notificationIdentifier and managedObjectInstance.

Table 11: Mapping of Parameters of "notifyClearedAlarm"

IS Parameter Name	CMIP SS Equivalent	Qualifier
objectclass	M-EVENT REPORT parameter 'Managed object class'	M
objectInstance	M-EVENT REPORT parameter 'Managed object instance'	M
notificationId	M-EVENT REPORT parameter 'Event information': notificationIdentifier	M
eventTime	M-EVENT REPORT parameter 'Event time'	M
systemDN	--	--
notificationType	M-EVENT REPORT parameter 'Event type'	M
probableCause	M-EVENT REPORT parameter 'Event information': probableCause	M
perceivedSeverity	M-EVENT REPORT parameter 'Event information': perceivedSeverity	M
alarmType	The semantics of this parameter is conveyed by the notification type.	--
correlatedNotifications	M-EVENT REPORT parameter 'Event information': correlatedNotifications	O
alarmId	M-EVENT REPORT parameter 'Event information': correlatedNotifications	M

Table 12: Mapping of Parameters of 'notifyAckStateChanged'

IS Parameter Name Notification parameters of Information Services	CMIP SS Equivalent Notification equivalences	Qualifier
objectclass	M-EVENT REPORT parameter 'Managed object class'	M
objectInstance	M-EVENT REPORT parameter 'Managed object instance'	M
notificationId	M-EVENT REPORT parameter 'Event information': notificationIdentifier	M
eventTime	M-EVENT REPORT parameter 'Event time'	M
systemDN	--	--
notificationType	M-EVENT REPORT parameter 'Event type'	M
notificationId	notificationIdentifier (note 1)	M
probableCause	M-EVENT REPORT parameter 'Event information': probableCause	M
specificProblems	specificProblems	Ø
perceivedSeverity	M-EVENT REPORT parameter 'Event information': perceivedSeverity	M
alarmType	The semantics of this parameter is conveyed by the notification type.	--
alarmId	M-EVENT REPORT parameter 'Event information': correlatedNotifications --- (note-2)	--
ackTime	M-EVENT REPORT parameter 'Event information':	M
ackState	additionalInformation	M
ackUserId		M
ackSystemId		O
NOTE 1: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302.		
NOTE 2: In the CMIP Solution Set the alarmId is not used. In the CMIP Solution Set the alarm notifications are univocally identified by means of notificationIdentifier and managedObjectInstance.		

Table 13: Mapping of Parameters of 'notifyAlarmListRebuilt'

Notification parameters of Information Services	CMIP Notification equivalents	Qualifier
notificationId	notificationIdentifier (note)	M
reason	reason	M
objectClass	rebuiltObjectClass	M
objectInstance	rebuiltObjectInstance	M
NOTE: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302.		

IS Parameter Name	CMIP SS Equivalent	Qualifier
objectclass	M-EVENT REPORT parameter 'Event information': rebuiltObjectClass	M
objectInstance	M-EVENT REPORT parameter 'Event information': rebuiltObjectInstance	M
notificationId	M-EVENT REPORT parameter 'Event information': notificationIdentifier	M
eventTime	M-EVENT REPORT parameter 'Event time'	M
systemDN	--	--
notificationType	M-EVENT REPORT parameter 'Event type'	M
reason	M-EVENT REPORT parameter 'Event information': reason	M

Table 14: Mapping of Parameters of 'notifyComments'

IS Parameter Name Notification parameters of Information Services	CMIP SS Equivalent Notification equivalents	Qualifier
notificationId	notificationIdentifier (note 1)	M
objectClass	M-EVENT REPORT parameter 'Event information': alarmedObjectClass	M
objectInstance	M-EVENT REPORT parameter 'Event information': alarmedObjectInstance	M
notificationId	M-EVENT REPORT parameter 'Event information': notificationIdentifier	M
eventTime	M-EVENT REPORT parameter 'Event information': alarmEventTime	M
systemDN	--	--
notificationType	M-EVENT REPORT parameter 'Event type'eventType	M
alarmType	M-EVENT REPORT parameter 'Event information': alarmType	M
probableCause	M-EVENT REPORT parameter 'Event information': alarmProbableCause	M
perceivedSeverity	M-EVENT REPORT parameter 'Event information': alarmPerceivedSeverity	M
comments	M-EVENT REPORT parameter 'Event information': comments	M
alarmId	M-EVENT REPORT parameter 'Event information': correlatedNotifications --- (note 2)	--
NOTE 1: notificationIdentifier is a parameter of the Notification Header also defined in 3GPP TS 32.302.		
NOTE 2: In the CMIP Solution Set the alarmId is not used. In the CMIP Solution Set the alarm notifications are univocally identified by means of notificationIdentifier and managedObjectInstance.		

5.4 Notifications

5.4.1 notifyAlarmListRebuilt (M)

alarmListRebuilt **NOTIFICATION**
BEHAVIOUR

alarmListRebuiltBehaviour;

WITH INFORMATION SYNTAX

TS32-111-4TypeModule .AlarmListRebuiltInfo;

REGISTERED AS { ts32-111AlarmNotification 1};

alarmListRebuiltBehaviour **BEHAVIOUR**
DEFINED AS

"This notification is used by the Agent to inform the NM that the alarm list has been rebuilt.

The 'Event Information' field contains the following data:

- *notificationIdentifier*
This ITU-T X.721 standardised parameter, together with MOI (Managed Object Instance), unambiguously identifies this notification.
- *rebuiltObjectClass*
This parameter carries the IRPAgent MOC when the entire AlarmList has been rebuilt. It carries a different MOC when the AlarmList has been partially rebuilt.
- *rebuiltObjectInstance*
This parameter carries DN of the IRPAgent when the entire AlarmList has been rebuilt. It carries the DN of another MOI when the AlarmList has been partially rebuilt and only the MOIs subordinate of this rebuilt MOI may be affected by this partial rebuilt.
- *reason*
The parameter indicates the reason for alarm list rebuilding (if applicable)."

5.4.2 notifyComments (O)

notifyComments **NOTIFICATION**
BEHAVIOUR

notifyCommentsBehaviour;

WITH INFORMATION SYNTAX

TS32-111-4TypeModule .NotifyComments;

REGISTERED AS { ts32-111AlarmNotification 2};

notifyCommentsBehaviour **BEHAVIOUR**
DEFINED AS

"This notification is used by the Agent to inform the NM that one or more comments have been associated to one alarm.

The 'Event Information' field contains the following data:

- *alarmedObjectClass*
This parameter specifies the object class representing the resource that raised the alarm to which the comment was attached.
- *alarmedObjectInstance*
This parameter specifies the object instance representing the resource that raised the alarm to which the comment was attached.
- *notificationIdentifier*

This parameter specifies the notification identifier (ITU-T X.733 [5]), which, together with the instance of the object emitting this notification, unambiguously identifies this notification.

- *alarmEventTime*
This parameter specifies the time when the alarm, to which the comment was attached, was first raised by the alarmed resource.
- *alarmType*
This parameter specifies the event type of the notification that reported the alarm to which the comment was attached.
- *alarmProbableCause*
This parameter specifies the probable cause (ITU-T X.733 [5]) of the alarm to which the comment was attached.
- *alarmPerceivedSeverity*
This parameter specifies the perceived severity (ITU-T X.733 [5]) of the alarm to which the comment is attached.
- *comments*
This parameter carries the text of the comment.

□alarmedObjectClass: defined in ITU-T X.710 [2] and X.711 [3]
 □alarmedObjectInstance: defined in ITU-T X.710 [2] and X.711 [3]
 □notificationIdentifier: This ITU-T X.721 standardised parameter, together with MOI (Managed Object Instance), unambiguously identifies this notification.
 □alarmEventTime: defined in ITU-T X.721
 □alarmType: the eventType of the alarm to which this comment is associated.
 □alarmProbableCause: defined in ITU-T X.721
 □alarmPerceivedSeverity: defined in ITU-T X.721
 □comments: the text of the comment.
 ";

6 ASN.1 definitions for Alarm IRP

```
TS32-111-4TypeModule {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-Operation-
Maintenance(3) ts-32-111(111) part4(4) informationModel(0) asn1Module(2) version1(1)}
```

```
DEFINITIONS IMPLICIT TAGS ::=
BEGIN
--EXPORTS everything
IMPORTS
```

```
NotificationIdentifier, Destination, EventTime, ProbableCause, PerceivedSeverity
FROM Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1}
```

```
AlarmInfo
FROM Notification-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 2}
```

```
CMISFilter, ObjectInstance, ObjectClass, EventTypeId
FROM CMIP-1 {joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3)};
```

```
baseNodeUMTS OBJECT IDENTIFIER ::= { itu-t (0) identified-organization (4) etsi (0) mobileDomain
(0)
umts-Operation-Maintenance (3) }
ts32-111Prefix OBJECT IDENTIFIER ::= { baseNodeUMTS ts-32-111(111)}
ts32-111Part4 OBJECT IDENTIFIER ::= { ts32-111Prefix part4(4)}
```

```

ts32-111-4InfoModel OBJECT IDENTIFIER ::= { ts32-111Part4 informationModel(0)}

ts32-111AlarmObjectClass      OBJECT IDENTIFIER ::= { ts32-111-4InfoModel managedObjectClass(3)}
ts32-111AlarmPackage          OBJECT IDENTIFIER ::= { ts32-111-4InfoModel package(4)}
ts32-111AlarmParameter       OBJECT IDENTIFIER ::= { ts32-111-4InfoModel parameter(5)}
ts32-111AlarmAttribute        OBJECT IDENTIFIER ::= { ts32-111-4InfoModel attribute(7)}
ts32-111AlarmAction           OBJECT IDENTIFIER ::= { ts32-111-4InfoModel action(9)}
ts32-111AlarmNotification     OBJECT IDENTIFIER ::= { ts32-111-4InfoModel action(10)}

-- Start of 3GPP SA5 own definitions
AckErrorList ::= SET OF ErrorInfo
AlarmReference ::= SEQUENCE
{
  moi ObjectInstance OPTIONAL, -- absent if scope of uniqueness of notificationId is across IRPAgent
  notificationIdentifier NotificationIdentifier
}
AckOrUnackAlarms ::= SEQUENCE
{
  alarmReferenceList SET OF AlarmReference, -- ITU-T X.721
  ackUserId          UserId,
  ackSystemId        SystemId OPTIONAL
}
AckOrUnackAlarmsReply ::= SEQUENCE
{
  status              ErrorCauses,
  errorAlarmReferenceList AckErrorList
}
AckState ::= ENUMERATED
{
  acknowledged      (0),
  unacknowledged    (1)
}
AckTime ::= GeneralizedTime
AlarmChoice ::= ENUMERATED
{
  allAlarms           _____(0),
  allActiveAlarms     _____(1),
  allActiveAndAckAlarms (2),
  allActiveAndUnackAlarms (3),
  allClearedAndUnackAlarms (4),
  allUnackAlarms      _____(5)
}
AlarmsCountSummary ::= SEQUENCE
{
  activeAlarmsCount      INTEGER, -- this is the sum of criticalCount, majorCount,
  minorCount, warningCount
                                -- and indeterminateCount
  criticalCount          _____INTEGER,
  majorCount             INTEGER,
  minorCount             INTEGER,
  warningCount           INTEGER,
  indeterminateCount     INTEGER,
  clearedCount           _____INTEGER
}
AlarmListRebuiltInfo ::= SEQUENCE
{
  notificationIdentifier NotificationIdentifier, -- ITU-T X.721
  rebuiltObjectClass      ObjectClass,
  rebuiltObjectInstance   ObjectInstance,
  reason                  ErrorCauses
}
Comment ::= GraphicString
ErrorCauses ::= ENUMERATED
{
  noError (0), -- operation / notification successfully performed
  wrongFilter (1), -- the value of the filter parameter is not valid
  wrongAlarmAckState (2), -- the value of the alarmAckState parameter (e.g. getAlarmCount) is not
  valid
  ackPartlySuccessful (3), -- acknowledgment request partly successful
  unackPartlySuccessful (4), -- unacknowledgment request partly successful
  wrongAlarmReference (5), -- alarm identifier used in the alarm reference list not found (e.g. in
  case of acknowledgement request)
  wrongAlarmReferenceList (6), -- the alarm reference list (e.g. in case of acknowledgement
  request) is empty or completely wrong
  alarmAlreadyAck (7), -- alarm to be acknowledged is already in this state
  alarmAlreadyUnack (8), -- alarm to be acknowledged is already in this state

```

```

wrongUserId (9), -- the user identifier in the unacknowledgement operation is not the same as
in the previous acknowledgementAlarms request
wrongSystemId (10), -- the system identifier in the unacknowledgement operation is not the same as
in the previous acknowledgementAlarms request
alarmAckNotAllowed (11), -- current management system not allowed to acknowledge the alarm (e.g.
due to acknowledgement competence rules)
setCommentPartlySuccessful (12), -- the setComment action partly successful (e.g. some alarmId
are not in the alarmList)
unspecifiedErrorReason (255) -- operation failed, specific error unknown
}
ErrorInfo ::= SEQUENCE
{
    moi ObjectInstance OPTIONAL, -- absent if uniqueness of notificationIdentifier is across
IRPagent
    notificationIdentifier NotificationIdentifier, -- ITU-T X.721
    reason ErrorCauses
}
GeneralObjectId ::= INTEGER
GetAlarmCount ::= SEQUENCE
{
    alarmAckState AlarmChoice OPTIONAL,
    filter CMISFilter OPTIONAL -- ITU-T X.711
}
GetAlarmCountReply ::= SEQUENCE
{
    criticalCount INTEGER,
    majorCount INTEGER,
    minorCount INTEGER,
    warningCount INTEGER,
    indeterminateCount INTEGER,
    clearedCount INTEGER,
    status ErrorCauses
}
GetAlarmIRPVersionReply ::= SEQUENCE
{
    versionNumberList SupportedAlarmIRPVersions,
    status ErrorCauses
}
GetAlarmList ::= SEQUENCE
{
    alarmAckState AlarmChoice OPTIONAL,
    destination Destination, -- ITU-T X.721
    filter CMISFilter OPTIONAL -- ITU-T X.711
}
GetAlarmListReply ::= SEQUENCE
{
    alignmentId INTEGER,
    status ErrorCauses
}
GetNotificationProfileReply ::= SEQUENCE
{
    notificationNameProfile NotificationList,
    notificationParameterProfile ParameterListOfList,
    status ErrorCauses
}
GetOperationProfileReply ::= SEQUENCE
{
    operationNameProfile OperationList,
    operationParameterProfile ParameterListOfList,
    status ErrorCauses
}
IRPVersionNumber ::= GraphicString
NotificationList ::= SET OF NotificationName
NotificationName ::= GraphicString
NotifyComments ::= SEQUENCE
{
    alarmedObjectClass ObjectClass, -- ITU-T X.711
    alarmedObjectInstance ObjectInstance, -- ITU-T X.711
    notificationIdentifier NotificationIdentifier, -- ITU-T X.721
    alarmEventTime EventTime, -- ITU-T X.721
    alarmType EventTypeId, -- ITU-T X.711
    alarmProbableCause ProbableCause, -- ITU-T X.721
    alarmPerceivedSeverity PerceivedSeverity, -- ITU-T X.721

```

```
comments          SET OF Comment
}

OperationList ::= SET OF OperationName

OperationName ::= GraphicString

ParameterList ::= SET OF ParameterName

ParameterListOfList ::= SET OF ParameterList

ParameterName ::= GraphicString

SetComment ::= SEQUENCE
{
  alarmReferenceList  SET OF AlarmReference,
  commentUserId      UserId,
  commentSystemId    SystemId,
  commentText        Comment
}

SetCommentReply ::= SEQUENCE
{
  badAlarmReferenceList  SET OF ErrorInfo,
  status                 ErrorCauses
}

SystemId ::= GraphicString

SupportedAlarmIRPVersions ::= SET OF IRPVersionNumber
UserId ::= GraphicString

END -- of module TS32-111-4TypeModule
```