

Source: SA5 (Telecom Management)

Title: 2 Rel-4/5 CR 32.111-4 (Fault Management; Part 4: Alarm IRP: CMIP solution set): Correction of syntax error in type SetComment

Document for: Approval

Agenda Item: 7.5.3

Doc-1st-Level	Spec	CR	Rev	Phase	Subject	Cat	Version-Current	Doc-2nd-Level	Workitem
SP-030416	32.111-4	021	-	Rel-4	Correction of syntax error in type SetComment	F	4.5.0	S5-036800	OAM-CM
SP-030416	32.111-4	022	-	Rel-5	Correction of syntax error in type SetCommentInfo	A	5.5.0	S5-036799	OAM-CM

CR-Form-v7
CHANGE REQUEST
⌘ 32.111-4 CR 021 ⌘ rev - ⌘ Current version: 4.5.0 ⌘

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

Proposed change affects: UICC apps ME Radio Access Network Core Network

Title:	⌘	Correction of syntax error in type SetComment	
Source:	⌘	SA5 (clemens.suerbaum@siemens.com)	
Work item code:	⌘	OAM-CM	Date: ⌘ 05/09/2003
Category:	⌘	F	Release: ⌘ Rel-4
		<i>Use one of the following categories:</i> 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 .	<i>Use one of the following releases:</i> 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) Rel-4 (Release 4) Rel-5 (Release 5) Rel-6 (Release 6)

Reason for change:	⌘	Missing tag in SetComment for SystemId. Since SystemId and CommentText are from the same type, no univocal encoding/decoding is possible, if optional SystemId is not present.	
Summary of change:	⌘	The CR adds a tag to the SystemId element in the ASN.1 sequence SetCommentInfo. Additionally the name of the type is changed to SetCommentInfo to use the usual naming of information syntax for actions.	
Consequences if not approved:	⌘	It would be impossible to decode the sequence SetCommentInfo correctly.	

Clauses affected:	⌘	5.3.4, 6									
Other specs affected:	⌘	<table border="1" style="display: inline-table; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">Y</td> <td style="width: 20px; text-align: center;">N</td> </tr> <tr> <td style="text-align: center;"> </td> <td style="text-align: center;">X</td> </tr> <tr> <td style="text-align: center;"> </td> <td style="text-align: center;">X</td> </tr> <tr> <td style="text-align: center;">X</td> <td style="text-align: center;"> </td> </tr> </table> Other core specifications	Y	N		X		X	X		⌘ Rel-5 32.111-4
		Y	N								
			X								
	X										
X											
Test specifications											
O&M Specifications											
Other comments:	⌘	Rel-5 Mirror CR in S5-036799.									

Change in Clause 5.3.4

5.3.4 setComment (O)

```
setComment ACTION
  BEHAVIOUR
    setCommentBehaviour;
  MODE
    CONFIRMED;
  WITH INFORMATION SYNTAX
    TS32-111-4TypeModule-.SetCommentInfo;
  WITH REPLY SYNTAX
    TS32-111-4TypeModule-.SetCommentReply;
  REGISTERED AS { ts32-111AlarmAction 4};
```

```
setCommentBehaviour BEHAVIOUR
```

DEFINED AS

"The NM invokes this action to associate a comment to one or more alarms.

The 'Action information' field contains:

- *alarmReferenceList*
Contains a list of alarm identifiers to which the comment must be associated.
- *commentUserId*
Contains the identity of the NM User that invokes this operation.
- *commentSystemId*
Contains the identity of the NM that invokes this operation.
- *commentText*
Contains the text of the comment.

The 'Action response' is composed of the following data:

- *errorAlarmReferenceList*
List of pair of alarmId and failure reason.
- *status*
It contains the results of the NM action. Possible values: actionSucceeded (0), actionPartiallyFailed (12) or another value indicating the reason of the error.";

End of Change in Clause 5.3.4

Change in Clause 6

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)
}

```

AlarmComments ::= SET OF SingleAlarmComment

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
}

```

CommentText ::= GraphicString

CommentTime ::= GeneralizedTime

```

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

OperationList ::= SET OF OperationName

OperationName ::= GraphicString

ParameterList ::= SET OF ParameterName

ParameterListOfList ::= SET OF ParameterList

ParameterName ::= GraphicString

| SetCommentInfo ::= SEQUENCE
{
  alarmReferenceList      SET OF AlarmReference,
  commentUserId           UserId,
  commentSystemId         [2] SystemId OPTIONAL,
  commentText             CommentText
}

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

SingleAlarmComment ::= SEQUENCE
{
  commentText             CommentText
  commentTime            CommentTime,
  commentUserId           UserId,
  commentSystemId        SystemId OPTIONAL
}

SystemId ::= GraphicString

SupportedAlarmIRPVersions ::= SET OF IRPVersionNumber

UserId ::= GraphicString

END -- of module TS32-111-4TypeModule

```

<p>End of Change in Clause 6 End of Document</p>

CR-Form-v7
CHANGE REQUEST
⌘ 32.111-4 CR 022 ⌘ rev - ⌘ Current version: 5.5.0 ⌘

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

Proposed change affects: UICC apps ME Radio Access Network Core Network

Title:	⌘	Correction of syntax error in type SetCommentInfo	
Source:	⌘	SA5 (clemens.suerbaum@siemens.com)	
Work item code:	⌘	OAM-CM	Date: ⌘ 05/09/2003
Category:	⌘	A	Release: ⌘ Rel-5
		<i>Use one of the following categories:</i> 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 .	<i>Use one of the following releases:</i> 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) Rel-4 (Release 4) Rel-5 (Release 5) Rel-6 (Release 6)

Reason for change:	⌘	Missing tag in SetCommentInfo for SystemId. Since SystemId and CommentText are from the same type, no univocal encoding/decoding is possible, if optional SystemId is not present.
Summary of change:	⌘	The CR adds a tag to the SystemId element in the ASN.1 sequence SetCommentInfo.
Consequences if not approved:	⌘	It would be impossible to decode the sequence SetCommentInfo correctly.

Clauses affected:	⌘	6								
Other specs affected:	⌘	<table border="1" style="display: inline-table; border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">Y</td> <td style="width: 20px;">N</td> </tr> <tr> <td style="width: 20px;">X</td> <td style="width: 20px;"> </td> </tr> <tr> <td style="width: 20px;"> </td> <td style="width: 20px;">X</td> </tr> <tr> <td style="width: 20px;"> </td> <td style="width: 20px;">X</td> </tr> </table> Other core specifications ⌘ Test specifications O&M Specifications	Y	N	X			X		X
Y	N									
X										
	X									
	X									
Other comments:	⌘	Rel-5 Mirror CR of S5-036800.								

Change in Clause 6

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 notification(10)}
```

```
-- Start of 3GPP SA5 own definitions
```

```
AckErrorList ::= SET OF ErrorInfo
```

```
AlarmReference ::= SEQUENCE
```

```
{
  moi ObjectInstance OPTIONAL, -- absent if scope of uniqueness of
  notificationIdentifier NotificationIdentifier
}
```

```
AckOrUnackAlarmsInfo ::= SEQUENCE
```

```
{
  alarmReferenceList SET OF AlarmReference,
  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)
}

AlarmComments ::= SET OF SingleAlarmComment

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
}

AlarmListAlignmentRequirement ::= ENUMERATED
{
  alignmentRequired      (0),    -- An alarm alignment is required.
  alignmentNotRequired   (1)    -- An alarm alignment is not required.
}

ClearAlarmsInfo ::= SEQUENCE
{
  alarmReferenceList     SET OF AlarmReference,
  clearUserId            UserId,
  clearSystemId          SystemId OPTIONAL
}

ClearAlarmsReply ::= SEQUENCE
{
  status                 ErrorCauses,
  errorAlarmReferenceList ClearErrorList
}

ClearErrorList ::= SET OF ErrorInfo

CommentText ::= GraphicString

CommentTime ::= GeneralizedTime

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)
  clearAlarmsPartlySuccessful (13), -- only some alarms to be cleared could be cleared
  clearAlarmsNotAllowed  (14), -- current management system not allowed to clear the alarm
  clearAlarmsAlarmAlreadyCleared (15), -- alarm to be cleared is already cleared
  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

GetAlarmCountInfo ::= 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

NotifyAlarmListRebuiltInfo ::= SEQUENCE
{
    notificationIdentifier              NotificationIdentifier,    -- ITU-T X.721
    rebuiltObjectClass                 ObjectClass,              -- ITU-T X.721
    rebuiltObjectInstance               ObjectInstance,          -- ITU-T X.721
    reason                             ErrorCauses,
    alarmListAlignmentRequirement      AlarmListAlignmentRequirement OPTIONAL
}

NotifyPotentialFaultyAlarmListInfo ::= SEQUENCE

```

```

{
potentialFaultyObjectClass      ObjectClass,          -- ITU-T X.711
potentialFaultyObjectInstance   ObjectInstance,         -- ITU-T X.711
notificationIdentifier          NotificationIdentifier,    -- ITU-T X.721
reason                          ReasonPotentialFaultyAlarmList
}

OperationList ::= SET OF OperationName

OperationName ::= GraphicString

ParameterList ::= SET OF ParameterName

ParameterListOfList ::= SET OF ParameterList

ParameterName ::= GraphicString

ReasonPotentialFaultyAlarmList ::= ENUMERATED
{
communicationErrorNEAgent      (0), -- A communication error between a NE and the agent has occurred.
agentRestart                    (1), -- The agent has restarted and not yet updated its alarm list.
indeterminate                    (2) -- The reason could not be determined.
}

SetCommentInfo ::= SEQUENCE
{
alarmReferenceList              SET OF AlarmReference,
commentUserId                   UserId,
commentSystemId                 [2] SystemId OPTIONAL,
commentText                     CommentText
}

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

SingleAlarmComment ::= SEQUENCE
{
commentText                     CommentText,
commentTime                     CommentTime,
commentUserId                   UserId,
commentSystemId                 SystemId OPTIONAL
}

SystemId ::= GraphicString

SupportedAlarmIRPVersions ::= SET OF IRPVersionNumber

UserId ::= GraphicString

END -- of module TS32-111-4TypeModule

```

<p>End of Change in Clause 6 End of Document</p>
