INTERNATIONAL STANDARD

# ISO/IEC 10165-2

First edition
1992-10-15



# Information technology — Open Systems Interconnection — Structure of management information: Definition of management information

*Technologies de l'information — Interconnexion de systèmes ouverts — Structure des informations de gestion: Définition des informations de gestion*

Reference number
ISO/IEC 10165-2:1992(E)

# Contents

Page

# Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 10165-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in collaboration with CCITT. The identical text is published as CCITT Recommendation X.721.

ISO/IEC 10165 consists of the following parts under the general title *Information technology – Open Systems Interconnection – Structure of management information*:

- Part 1: *Management information model*
- Part 2: *Definition of management information*
- Part 4: *Guidelines for the definition of managed objects*
- Part 5: *Generic management information*
- Part 6: *Requirements and guidelines for implementation conformance statement proformas associated with management information*

Annex A forms an integral part of this part of ISO/IEC 10165. Annexes B, C, D, E, F and G are for information only.

# Introduction

ISO/IEC 10165 is a multipart standard developed according to ISO 7498 and ISO/IEC 7498-4. ISO/IEC 10165 is related to the following International Standards:

– ISO/IEC 9595:1990, *Information technology – Open Systems Interconnection – Common management information service definition;*

– ISO/IEC 9596:1990, *Information technology – Open Systems Interconnection – Common management information protocol;*

– ISO/IEC 10040:1992, *Information technology – Open Systems Interconnection – Systems management overview;*

– ISO/IEC 10064:1992, *Information technology – Open Systems Interconnection – Systems Management.*

This page intentionally left blank

**INTERNATIONAL STANDARD**

**CCITT RECOMMENDATION**

# INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – STRUCTURE OF MANAGEMENT INFORMATION: DEFINITION OF MANAGEMENT INFORMATION



## 1 Scope

This Recommendation | International Standard

- defines the managed object classes, attribute types, name bindings, packages, specific attributes, action types, parameter types and notification types documented in accordance with CCITT Rec. X.722 | ISO/IEC 10165-4;
- specifies compliance requirements placed on other CCITT Recommendations | International Standards that make use of these definitions.

This Recommendation | International Standard is applicable to the development of OSI managed object class specifications and provides generic definitions that support OSI systems management functions. These definitions may also be used in other Recommendations | International Standards specifying object classes, attributes, notifications and action types.

## 2 Normative references

The following CCITT Recommendations | International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent editions of the Recommendations|Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The CCITT Secretariat maintains a list of the currently valid CCITT Recommendations.

### 2.1 Identical CCITT Recommendations| International Standards

- CCITT Recommendation X.720 (1992) | ISO/IEC 10165-1 : 1992, *Information technology - Open Systems Interconnection - Structure of management information: Management information model.*
- CCITT Recommendation X.722 (1992) | ISO/IEC 10165-4 : 1992, *Information technology - Open Systems Interconnection - Structure of management information: Guidelines for definition of managed objects.*
- CCITT Recommendation X.730 (1992) | ISO/IEC 10164-1 : 1992, *Information technology - Open Systems Interconnection - Systems Management: Object management function.*
- CCITT Recommendation X.731 (1992) | ISO/IEC 10164-2 : 1992, *Information technology - Open System Interconnection - Systems Management: State management function.*
- CCITT Recommendation X.732 (1992) | ISO/IEC 10164-3 : 1992, *Information technology - Open Systems Interconnection - Systems Management: Attributes for representing relationships.*
- CCITT Recommendation X.733 (1992) | ISO/IEC 10164-4 : 1992, *Information technology - Open Systems Interconnection - Systems Management: Alarm reporting function.*

- CCITT Recommendation X.734[1] | ISO/IEC 10164-5 : 1992, *Information technology - Open Systems Interconnection - Systems Management Event report management function.*
- CCITT Recommendation X.735[1] | ISO/IEC 10164-6 : 1992, *Information technology - Open Systems Interconnection - Systems Management: Log control function.*
- CCITT Recommendation X.736 (1992) | ISO/IEC 10164-7 : 1992, *Information technology - Open Systems Interconnection - Systems Management: Security alarm reporting function.*

## 2.2 Paired CCITT Recommendations| International Standards equivalent in technical content

- CCITT Recommendation X.200 (1988), *Reference model of Open Systems Interconnection for CCITT applications.*

  ISO 7498 : 1984, *Information processing systems - Open Systems Interconnection - Basic Reference Model.*
- CCITT Recommendation X.208 (1988), *Specification of abstract syntax notation one (ASN.1).*

  ISO/IEC 8824: 1990, *Information technology - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1).*
- CCITT Recommendation X.501 (1988), *The Directory - Models.*

  ISO/IEC 9594-2 : 1990 *Information technology - Open Systems Interconnection - The Directory - Part 2: The Models.*
- CCITT Recommendation X.710 (1991), *Common management information service definition for CCITT applications.*

  ISO/IEC 9595 : 1991, *Information technology - Open Systems Interconnection - Common management information service definition.*
- CCITT Recommendation X.711 (1991), *Common management information protocol specification for CCITT applications.*

  ISO/IEC 9596-1: 1991, *Information technology - Open Systems Interconnection - Common management information protocol specification - Part 1: Specification.*



# 3 Definitions

For the purposes of this Recommendation | International Standard the following definitions apply.

## 3.1 Event report management function definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.734 | ISO/IEC 10164-5:

a) discriminator;

b) event forwarding discriminator;

c) potential event report.

## 3.2 Management information model definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.720 | ISO/IEC 10165-1:

a) Attribute type;

b) Distinguished name;

c) Relative distinguished name.

---

[1] Presently at state of draft Recommendation.

# 4 Abbreviations

| | |
|---|---|
| ASN.1 | Abstract Syntax Notation One |
| CMIS | Common Management Information Service |
| CMIP | Common Management Information Protocol |
| EFD | Event Forwarding Discriminator |
| Id | Identifier |
| PDU | Protocol Data Unit |
| RDN | Relative Distinguished Name |

# 5 Notation

Attribute types and specific attributes are defined in this Recommendation | International Standard using the templates, defined in CCITT Rec. X.722 | ISO/IEC 10165-4.

The behavioural aspects of the specific attributes described here shall be incorporated into the definition of the managed object class importing these specific attributes.

# 6 Definition of managed object classes

This Recommendation | International Standard defines Managed Object classes which are referenced by the System Management Functions of CCITT Recs. in X.730 - X.736 | ISO/IEC 10164 parts 1 to 7, or which are intended to be used as superclasses for the purpose of inheritance in the definition of Managed Object classes in other standards. The syntax of the attributes referenced in the templates are defined in clause 13.

## 6.1 Alarm record

The **alarmRecord** managed object class is used to define the information stored in the log as a result of receiving alarm notifications or alarm reports. The semantics of the managed object class, namely its attributes and behaviour are derived from the Alarm notification described in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
alarmRecord        MANAGED OBJECT CLASS
 DERIVED FROM  eventLogRecord;
 CHARACTERIZED BY
-- The appropriate object identifier values for the eventType attribute, inherited from eventLogRecord
-- managed object class are communicationAlarm, qualityofServiceAlarm, processingErrorAlarm,
-- equipmentAlarm and environmentalAlarm --
          alarmRecordPackage      PACKAGE
          BEHAVIOUR
          alarmRecordBehaviour    BEHAVIOUR
          DEFINED AS "This managed object is used to represent logged information that resulted from alarm
          notifications or event reports";;
          ATTRIBUTES
          probableCause     GET,
          perceivedSeverity GET;;;
 CONDITIONAL PACKAGES
          specificProblemsPackage PACKAGE
          ATTRIBUTES
          specificProblems  GET;
          REGISTERED AS{smi2Package 1}; PRESENT IF "the Specific problems parameter is present in the alarm
          notification or event report corresponding to the instance of alarm record",

          backedUpStatusPackage  PACKAGE
          ATTRIBUTES
          backedUpStatus   GET;
          REGISTERED AS{smi2Package 2}; PRESENT IF "the backedUpStatus attribute has a value TRUE and the
          Backed up status parameter is present in the alarm notification or event report corresponding to the instance
          of alarm record",
```

```
        backUpObjectPackage     PACKAGE
        ATTRIBUTES
        backUpObject      GET;
        REGISTERED AS{smi2Package 3}; PRESENT IF "Backup object parameter is present in the alarm notification
        or event report corresponding to the instance of alarm record",

        trendIndicationPackage    PACKAGE
        ATTRIBUTES
        trendIndication     GET;
        REGISTERED AS{smi2Package 4}; PRESENT IF "the Trend indication parameter is present in the alarm
        notification or event report corresponding to the instance of alarm record",

        thresholdInfoPackage      PACKAGE
        ATTRIBUTES
        thresholdInfo       GET;
        REGISTERED AS{smi2Package 5}; PRESENT IF "the value for probableCause attribute is thresholdCrossed",

        stateChangeDefinitionPackage      PACKAGE
        ATTRIBUTES
        stateChangeDefinition     GET;
        REGISTERED AS{smi2Package 6}; PRESENT IF "there is a state transition for the states defined in State
        Management Function, corresponding to the alarm type specified in the alarm record",

        monitoredAttributesPackage        PACKAGE
        ATTRIBUTES
        monitoredAttributes       GET;
        REGISTERED AS{smi2Package 7}; PRESENT IF "the monitoredAttributes parameter is present in the alarm
        notification or event report corresponding to the instance of alarm record",

        proposedRepairActionsPackage    PACKAGE
        ATTRIBUTES
        proposedRepairActions GET;
        REGISTERED AS{smi2Package 8}; PRESENT IF "the proposedRepairActions parameter is present in the
        alarm notification or event report corresponding to the instance of alarm record";

REGISTERED AS  {smi2MObjectClass 1};
```

## 6.2 Attribute value change record

The **attributeValueChangeRecord** managed object class is used to define the information stored in the log as a result of receiving attribute value change notifications or attribute value change event reports. The semantics of the managed object class, namely its attributes and behaviour are derived from the Attribute Value Change notification described in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
attributeValueChangeRecord      MANAGED OBJECT CLASS
 DERIVED FROM    eventLogRecord;
 CHARACTERIZED BY
-- The appropriate object identifier values for the eventType attribute, inherited from eventLogRecord
-- managed object class, is attributeValueChange
attributeValueChangeRecordPackage       PACKAGE
        BEHAVIOUR
        attributeValueChangeRecordBehaviour     BEHAVIOUR
        DEFINED AS "This managed object is used to represent logged information that resulted from attribute value
        change notifications or event reports";;
        ATTRIBUTES
        attributeValueChangeDefinition     GET;;;

CONDITIONAL PACKAGES
        sourceIndicatorPackage          PRESENT IF "the sourceIndicator parameter is present in the
        attributeValueChange notification or event report corresponding to the instance of attribute value change
        record",

        attributeIdentifierListPackage      PRESENT IF "the attributeIdentifierList parameter is present in the
        attributeValueChange notification or event report corresponding to the instance of attribute value change
        record";
```

```
REGISTERED AS  {smi2MObjectClass 2};
```

## 6.3 Discriminator

The **discriminator** managed object class is used to define the criteria for controlling management services. The semantics of the managed object class, namely its attributes and behaviour are described in CCITT Rec. X.734 | ISO/IEC 10164-5.

```
discriminator        MANAGED OBJECT CLASS
 DERIVED FROM    top;
 CHARACTERIZED BY
discriminatorPackage        PACKAGE
        BEHAVIOUR
        discriminatorBehaviour    BEHAVIOUR
        DEFINED AS "This managed object is used to represent the criteria for controlling management services.";;
        ATTRIBUTES
        discriminatorId    GET,
        discriminatorConstruct
        REPLACE-WITH-DEFAULT
        DEFAULT VALUE Attribute-ASN1Module.defaultDiscriminatorConstruct    GET-REPLACE,
        administrativeState        GET-REPLACE,
        operationalState   GET;
        NOTIFICATIONS
        stateChange,
        attributeValueChange,
        objectCreation,
        objectDeletion;;;
-- the above events are defined in CCITT Rec. X.731 | ISO/IEC10164- 2, CCITT Rec. X.730 | ISO/IEC10164-1
CONDITIONAL PACKAGES
        availabilityStatusPackage PRESENT IF "any of the scheduling packages, ( duration, weekly scheduling,
        external) are present",

        duration            PRESENT IF "the discriminator function is scheduled to start at a specified time and stop at
        either a specified time or function continuously ",

        dailyScheduling            PRESENT IF "both the weekly scheduling package and external scheduler packages
        are not present in an instance and daily scheduling is supported by that instance",

        weeklyScheduling PRESENT IF "both the daily scheduling package and external scheduler packages are not
        present in an instance and weekly scheduling is supported by that instance",

        externalScheduler PRESENT IF "both the daily scheduling package and weekly scheduling packages are not
        present in an instance and external scheduling is supported by that instance";
-- see CCITT Rec. X.734 | ISO/IEC 10164-5 for the description of this managed object class.

REGISTERED AS  {smi2MObjectClass 3};
```



## 6.4 Event forwarding discriminator

The **eventForwardingDiscriminator** managed object class is used to define the conditions that shall be satisfied by potential event reports before the event report is forwarded to a particular destination.This managed object class is a subclass of **discriminator** managed object class. The semantics of the managed object class, namely its attributes, management operations and behaviour, are described in CCITT Rec. X. 734 | ISO/IEC 10164-5.

```
eventForwardingDiscriminator        MANAGED OBJECT CLASS
DERIVED FROM    discriminator;
CHARACTERIZED BY
-- The value for the administrative state if not specified at initiation defaults to the value unlocked.
efdPackage        PACKAGE
        BEHAVIOUR
        eventForwardingDiscriminatorBehaviour    BEHAVIOUR
```

```
        DEFINED AS "This managed object is used to represent the criteria that shall be satisfied by potential event
        reports before the event report is forwarded to a particular destination.";;
        ATTRIBUTES
        destination          GET-REPLACE;;;
-- discriminatorConstruct attribute is defined using the attributes of a potential event report object
-- described in CCITT Rec. X.734 | ISO/IEC 10164-5.

CONDITIONAL PACKAGES
        backUpDestinationListPackage     PACKAGE
        ATTRIBUTES
        activeDestination  GET,
        backUpDestinationList     GET-REPLACE;
        REGISTERED AS{smi2Package 9} ; PRESENT IF "the event forwarding discriminator is required to provide a
        backup for the destination",

        modePackage      PACKAGE
        ATTRIBUTES
        confirmedMode    GET;

        REGISTERED AS{smi2Package 10}; PRESENT IF "the event forwarding discriminator permits mode for
        reporting events to be specified by the managing system";

REGISTERED AS  {smi2MObjectClass 4};
```

## 6.5 Event log record

The **eventLogRecord** managed object class is used to define the information stored in the log as a result of receiving notifications or event reports. This is a superclass from which records for specific event types are derived.

```
eventLogRecord     MANAGED OBJECT CLASS
 DERIVED FROM    logRecord;
 CHARACTERIZED BY
eventLogRecordPackage     PACKAGE
        BEHAVIOUR
        eventLogRecordBehaviour          BEHAVIOUR
        DEFINED AS "This managed object represents the information stored in the log as a result of receiving
        notifications or incoming event reports.";;
        ATTRIBUTES
        managedObjectClass       GET,
        managedObjectInstance    GET,
        eventType                GET;;;

CONDITIONAL PACKAGES
        eventTimePackage         PACKAGE
        ATTRIBUTES
        eventTime          GET;
        REGISTERED AS{smi2Package 11}; PRESENT IF "the event time parameter was present in the received
        event report",

        notificationIdentifierPackage        PRESENT IF "the notification Identifier parameter is present in the
        notification or event report corresponding to the instance of an event record or an instance of its subclasses",

        correlatedNotificationsPackage     PRESENT IF "the correlatedNotifications parameter is present in the
        notification or event report corresponding to the instance of an event record or an instance of its subclasses",

        additionalTextPackage              PRESENT IF "the Additional text parameter is present in the notification or
        report corresponding to the instance of event record or an instance of its subclasses",

        additionalInformationPackage       PRESENT IF "the Additional information parameter is present in the
        notification or report corresponding to the instance of event record or an instance of its subclasses";

REGISTERED AS  {smi2MObjectClass 5};
```

## 6.6 Log

The **log** managed object class is used to define the criteria for controlling the logging of the information in an open system. The semantics of the managed object class, namely its attributes and behaviour are described in CCITT Rec. X.735 | ISO/IEC 10164-6.

```
log                 MANAGED OBJECT CLASS
 DERIVED FROM   top;
 CHARACTERIZED BY
-- see CCITT Rec. X.735 | ISO/IEC 10164-6 for the description of this managed object class.
logPackage          PACKAGE
        BEHAVIOUR
        logBehaviour      BEHAVIOUR
        DEFINED AS "This managed object is used to store incoming event reports and local system notifications.
        Additional details are defined in CCITT Rec. X. 735 | ISO/IEC 10164-6. ";;
        ATTRIBUTES
        logId      GET,
        discriminatorConstruct     GET-REPLACE ,
        administrativeState        GET-REPLACE,
        operationalState   GET,
        availabilityStatus PERMITTED VALUES Attribute-ASN1Module.LogAvailability
                           REQUIRED VALUES Attribute-ASN1Module.UnscheduledLogAvailability        GET,
        logFullAction      GET-REPLACE;
        NOTIFICATIONS
        objectCreation,
        objectDeletion,
        attributeValueChange,
        stateChange,
        processingErrorAlarm;;;

CONDITIONAL PACKAGES
        finiteLogSizePackage      PACKAGE
        ATTRIBUTES
        maxLogSize        GET-REPLACE,
        currentLogSize    GET,
        numberOfRecords           GET;
        REGISTERED AS{smi2Package 12};      PRESENT IF "an instance supports it",

        logAlarmPackage PACKAGE
        ATTRIBUTES
        capacityAlarmThreshold  GET-REPLACE ADD-REMOVE;
        REGISTERED AS{smi2Package 13} ;     PRESENT IF "a log is of finite size and halts logging when the
        availability status has the log full value.",

        availabilityStatusPackage                   PRESENT IF "any of the scheduling packages, (duration, weekly
        scheduling, external) are present. The presence of this package makes available the off-duty value of the
        available status attribute to the object.",

        duration                            PRESENT IF "the logging function is scheduled to start at a specified time
        and stop at either a specified time or function continuously.",

        dailyScheduling                             PRESENT IF "both the weekly scheduling package and external
        scheduler packages are not present in an instance and daily scheduling is supported by that instance.",

        weeklyScheduling                    PRESENT IF "both the daily scheduling package and external scheduler
        packages are not present in an instance and weekly scheduling is supported by that instance.",

        externalScheduler                   PRESENT IF "both the daily scheduling package and weekly scheduling
        packages are not present in an instance and external scheduling is supported by that instance.";

REGISTERED AS   {smi2MObjectClass 6};
```

## 6.7 Log record

The **logRecord** managed object class is used to define the records contained in a log managed object. The semantics of the managed object class, namely its attributes and behaviour are described in CCITT Rec. X.735 | ISO/IEC 10164-6.

```
logRecord           MANAGED OBJECT CLASS
 DERIVED FROM   top;
 CHARACTERIZED BY
logRecordPackage PACKAGE
          BEHAVIOUR
          logRecordBehaviour       BEHAVIOUR
          DEFINED AS "This managed object represents the information stored in the logs";;
          ATTRIBUTES
          logRecordId       GET,
          loggingTime       GET;;;

REGISTERED AS   {smi2MObjectClass 7};
```

## 6.8 Object creation record

The **objectCreationRecord** managed object class is used to define the information stored in the log as a result of receiving object creation notifications or Object Creation event reports. The semantics of the managed object class, namely its attributes and behaviour are derived from the Object Creation notification described in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
objectCreationRecord        MANAGED OBJECT CLASS
 DERIVED FROM   eventLogRecord;
 CHARACTERIZED BY
-- The appropriate object identifier value for the eventType attribute, inherited from eventLogRecord
-- managed object class, is objectCreation
 objectCreationRecordPackage       PACKAGE
          BEHAVIOUR
          objectCreationRecordBehaviour    BEHAVIOUR
          DEFINED AS "This managed object is used to represent logged information that resulted from object creation
          notifications or event reports";;;;

CONDITIONAL PACKAGES
          sourceIndicatorPackage   PRESENT IF "the sourceIndicator parameter is present in the object creation
          notification or event report corresponding to the instance of object creation record",

          attributeListPackage       PRESENT IF "the attributeList parameter is present in the object creation notification
          or event report corresponding to the instance of object creation record";

REGISTERED AS  {smi2MObjectClass 8};
```

## 6.9 Object deletion record

The **objectDeletionRecord** managed object class is used to define the information stored in the log as a result of receiving object deletion notifications or Object deletion event reports. The semantics of the managed object class, namely its attributes and behaviour are derived from the Object Deletion notification described in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
objectDeletionRecord        MANAGED OBJECT CLASS
 DERIVED FROM   eventLogRecord;
 CHARACTERIZED BY
-- The appropriate object identifier value for the eventType attribute, inherited from eventLogRecord
-- managed object class, is objectDeletion
 objectDeletionRecordPackage       PACKAGE
          BEHAVIOUR
          objectDeletionRecordBehaviour    BEHAVIOUR
          DEFINED AS "This managed object is used to represent logged information that resulted from object deletion
          notifications or event reports";;;;

 CONDITIONAL PACKAGES
```

sourceIndicatorPackage PRESENT IF "the sourceIndicator parameter is present in the object deletion notification or event report corresponding to the instance of object deletion record",

attributeListPackage PRESENT IF "the attributeList parameter is present in the object deletion notification or event report corresponding to the instance of object deletion record";

REGISTERED AS {smi2MObjectClass 9};

## 6.10 Relationship change record

The **relationshipChangeRecord** managed object class is used to define the information stored in the log as a result of receiving relationship change notifications or relationship change event reports. The semantics of the managed object class, namely its attributes and behaviour are derived from the Relationship Change notification described in CCITT Rec.X.732 | ISO/IEC 10164-3.

```
relationshipChangeRecord MANAGED OBJECT CLASS
 DERIVED FROM   eventLogRecord;
-- The appropriate object identifier value for the eventType attribute, inherited from eventLogRecord
-- managed object class, is relationshipChange
CHARACTERIZED BY
relationshipChangeRecordPackage PACKAGE
        BEHAVIOUR
        relationshipChangeRecordBehaviour       BEHAVIOUR
        DEFINED AS "This managed object is used to represent logged information that resulted from
        relationship change notifications or event reports";;
        ATTRIBUTES
        relationshipChangeDefinition    GET;;;

CONDITIONAL PACKAGES
        sourceIndicatorPackage          PRESENT IF "the Source indicator parameter is present in the
        relationshipChange notification or event report corresponding to the instance of relationship change record",

        attributeIdentifierListPackage  PRESENT IF "the Attribute identifier list parameter is present in the
        relationshipChange notification or event report corresponding to the instance of relationship change record";

REGISTERED AS {smi2MObjectClass 10};
```

## 6.11 Security alarm report record

The **securityAlarmReportRecord** managed object class is used to define the information stored in the log as a result of receiving security alarm notifications or security alarm event reports. The semantics of the managed object class, namely its attributes and behaviour are derived from the Security Alarm notification described in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
securityAlarmReportRecord MANAGED OBJECT CLASS
 DERIVED FROM   eventLogRecord;
 CHARACTERIZED BY
-- The appropriate object identifier values for the eventType attribute, inherited from eventLogRecord
-- managed object class, are integrityViolation, operationalViolation, physicalViolation,
-- securityServiceOrMechanismViolation and timeDomainViolation --
securityAlarmRecordPackage      PACKAGE
        BEHAVIOUR
        securityAlarmReportRecordBehaviour      BEHAVIOUR
        DEFINED AS "This managed object is used to represent logged information that resulted from security alarm
        notifications or event reports";;
        ATTRIBUTES
        securityAlarmCause      GET,
        securityAlarmSeverity   GET,
        securityAlarmDetector   GET,
        serviceUser             GET,
        serviceProvider         GET;;;

REGISTERED AS {smi2MObjectClass 11};
```

## 6.12 State change record

The **stateChangeRecord** managed object class is used to define the information stored in the log as a result of receiving state change notifications or state change event reports. The semantics of the managed object class, namely its attributes and behaviour are derived from the State Change notification described in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
stateChangeRecord          MANAGED OBJECT CLASS
 DERIVED FROM   eventLogRecord;
-- The appropriate object identifier value for the eventType attribute, inherited from eventLogRecord
-- managed object class, is stateChange
CHARACTERIZED BY
stateChangeRecordPackage          PACKAGE
          BEHAVIOUR
          stateChangeRecordBehaviour      BEHAVIOUR
          DEFINED AS "This managed object is used to represent logged information that resulted from state change
          notifications or event reports";;
          ATTRIBUTES
          stateChangeDefinition      GET;;;

CONDITIONAL PACKAGES
          sourceIndicatorPackage              PRESENT IF "the Source indicator parameter is present in the stateChange
          notification or event report corresponding to the instance of state change record",

          attributeIdentifierListPackage      PRESENT IF "the Attribute identifierList parameter is present in the
          stateChange notification or event report corresponding to the instance of state change record";

REGISTERED AS  {smi2MObjectClass 12};
```

## 6.13 System

The **system** managed object class is used to represent a set of hardware and software that forms an autonomous whole capable of performing information processing and/or information transfer.

The specification of the sequence of name bindings to be used in constructing the distinguished name for a system managed object is outside the scope of this Recommendation | International Standard . Examples of names for systems are specified in CCITT Rec. X.720 | ISO/IEC 10165-1.

NOTE- This definition does not correspond to real open system but corresponds to real systems in CCITT Rec. X.200 | ISO 7498.

An instance of this managed object class may be used as the superior in naming managed objects representing either information processing and or information transfer resources contained within this instance.

```
system              MANAGED OBJECT CLASS
 DERIVED FROM   top;
 CHARACTERIZED BY
 systemPackage    PACKAGE
          ATTRIBUTES
          systemId GET,
          systemTitle          GET,
          operationalState   GET    usageState        GET;;;

CONDITIONAL PACKAGES
          administrativeStatePackage          PACKAGE
          ATTRIBUTES
          administrativeState          GET-REPLACE ;
          REGISTERED AS{smi2Package 14};          PRESENT IF "an instance supports it.",

          supportedFeaturesPackage          PACKAGE
          ATTRIBUTES
          supportedFeatures          GET-REPLACE ADD-REMOVE;
          REGISTERED AS{smi2Package 15}; PRESENT IF "an instance supports it.";

REGISTERED AS  {smi2MObjectClass 13};
```

### 6.14 Top

The **top** managed object class, is that class of which every other managed object class is a subclass.

The semantics of the attributes objectClass,packages,nameBindings, and allomorphs are defined in CCITT Rec. X.720 | ISO/IEC 10165-1.

```
top                 MANAGED OBJECT CLASS
CHARACTERIZED BY
topPackage          PACKAGE
        BEHAVIOUR
        topBehaviour;
        ATTRIBUTES
        objectClass GET,
        nameBinding     GET ;;;

CONDITIONAL PACKAGES
        packagesPackage         PACKAGE
        ATTRIBUTES      packages        GET;
        REGISTERED AS{smi2Package 16};
        PRESENT IF "any registered package, other than this package has been instantiated",

        allomorphicPackage      PACKAGE
        ATTRIBUTES
        allomorphs      GET;
        REGISTERED AS{smi2Package 17};
        PRESENT IF " if an object supports allomorphism" ;

REGISTERED AS  {smi2MObjectClass 14};

topBehaviour        BEHAVIOUR
        DEFINED AS "This is the top level of managed object class hierarchy and every other managed object class is
        a specialization of either this generic class (top) or a specialization of subclass of top. The parameter
        miscellaneousError is to be used when a processing failure has occurred and the error condition encountered
        does not match any of object's defined specific error types.";
```

## 7 Name binding for managed object classes

This clause specifies name bindings for the managed object classes. Additional name bindings may be defined for these managed object classes and their subclasses.

### 7.1 Discriminator

```
discriminator-system        NAME BINDING
                    SUBORDINATE OBJECT CLASSdiscriminator       AND SUBCLASSES;
                    NAMED BY
                    SUPERIOR OBJECT CLASS system AND SUBCLASSES ;
                    WITH ATTRIBUTE      discriminatorId;
                    CREATE
                    WITH-REFERENCE-OBJECT ,
                    WITH-AUTOMATIC-INSTANCE-NAMING;
                    DELETE
                    ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS  {smi2NameBinding 1};
```

### 7.2 Log

```
log-system          NAME BINDING
                    SUBORDINATE OBJECT CLASSlog       AND SUBCLASSES ;
                    NAMED BY
                    SUPERIOR OBJECT CLASS system AND SUBCLASSES ;
```

```
                    WITH ATTRIBUTE      logId;
                    CREATE
                    WITH-REFERENCE-OBJECT ,
                    WITH-AUTOMATIC-INSTANCE-NAMING;
                    DELETE
                    ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS  {smi2NameBinding 2};
```

### 7.3 Log record

```
logRecord-log       NAME BINDING
                    SUBORDINATE OBJECT CLASS logRecord AND SUBCLASSES;
                    NAMED BY
                    SUPERIOR OBJECT CLASS      log AND SUBCLASSES;
                    WITH ATTRIBUTE
                    logRecordId;
                    DELETE
                    only-if-no-contained-objects;

REGISTERED AS  {smi2NameBinding 3};
```

## 8 Definition of packages

This Recommendation | International Standard defines a number of packages which are referenced by the System Management Functions of CCITT Recs.X.730 - 736 | ISO/IEC 10164 parts 1 to 7.

### 8.1 Additional information

The **additionalInformationPackage** package contains the additionalInformation attribute described in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
additionalInformationPackage      PACKAGE
                    ATTRIBUTES
                    additionalInformation     GET;

REGISTERED AS  {smi2Package 18};
```

### 8.2 Additional text

The **additionalTextPackage** package contains the additionalText attribute described in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
additionalTextPackage      PACKAGE
                    ATTRIBUTES
                    additionalText   GET;

REGISTERED AS  {smi2Package 19};
```

### 8.3 Attribute identifier list

The **attributeIdentifierListPackage** package contains the attributeIdentifierList attribute described in CCITT Rec. X.730 | ISO/IEC 10164-1, CCITT Rec. X.731 | ISO/IEC 10164-2, and CCITT Rec. X.732 | ISO/IEC 10164-3.

```
attributeIdentifierListPackage      PACKAGE
                    ATTRIBUTES
                    attributeIdentifierList     GET;

REGISTERED AS  {smi2Package 20};
```

## 8.4 Attribute list

The **attributeListPackage** package contains the attributeList attribute described in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
attributeListPackage        PACKAGE
                    ATTRIBUTES
                    attributeList     GET;

REGISTERED AS  {smi2Package 21};
```

## 8.5 Availability status

The **availabilityStatusPackage** package contains the availabilityStatus attribute described in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
availabilityStatusPackage   PACKAGE
        BEHAVIOUR
        availabilityStatusBehaviour        BEHAVIOUR
        DEFINED AS " This package is described in CCITT Rec. X.734,X.735 | ISO/IEC 10164-5, 10164-6. It is used
        to indicate the availability of the resource according to a predetermined time schedule." ;;
        ATTRIBUTES
        availabilityStatus REQUIRED VALUES Attribute-ASN1Module.SchedulingAvailability     GET;

REGISTERED AS  {smi2Package 22};
```

## 8.6 Correlated notifications

The **correlatedNotificationsPackage** package contains the correlatedNotifications attribute described in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
correlatedNotificationsPackage      PACKAGE
                    ATTRIBUTES
                    correlatedNotifications   GET;

REGISTERED AS  {smi2Package 23};
```

## 8.7 Notification identifier

The **notificationIdentifierPackage** package contains the notificationIdentifier attribute described in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
notificationIdentifierPackage       PACKAGE
                    ATTRIBUTES
                    notificationIdentifier     GET;

REGISTERED AS  {smi2Package 24};
```

## 8.8 Daily scheduling

The semantics of **dailyScheduling** package are described in CCITT Rec. X.734 | ISO/IEC 10164-5 and CCITT Rec. X.735 | ISO/IEC 10164-6.

```
dailyScheduling     PACKAGE
                    ATTRIBUTES
                    intervalsOfDay  REPLACE-WITH-DEFAULT
                                    DEFAULT VALUE Attribute-ASN1Module.defaultIntervalsOfDay
                                    GET-REPLACE ADD-REMOVE;

REGISTERED AS  {smi2Package 25};
```

## 8.9 Duration

The semantics of **duration** package are described in CCITT Rec.X.734 | ISO/IEC 10164-5 and CCITT Rec.X.735 | ISO/IEC 10164-6.

```
duration            PACKAGE
                    ATTRIBUTES
                    startTime       GET-REPLACE,
                    stopTime        REPLACE-WITH-DEFAULT
                                    DEFAULT VALUE Attribute-ASN1Module.defaultStopTime
                                    GET-REPLACE;

REGISTERED AS  {smi2Package 26};
```

## 8.10 External scheduler

The semantics of **externalScheduler** package are described in CCITT Rec. X.734 | ISO/IEC 10164-5 and CCITT Rec. X.735 | ISO/IEC 10164-6.

```
externalScheduler       PACKAGE
                        ATTRIBUTES
                        schedulerName GET;
REGISTERED AS  {smi2Package 27};
```

## 8.11 Source indicator

The **sourceIndicatorPackage** package contains the sourceIndicator attribute described in CCITT Rec. X.730 | ISO/IEC 10164-1, CCITT Rec. X.731 | ISO/IEC 10164-2, and CCITT Rec. X.732 | ISO/IEC 10164-3.

```
sourceIndicatorPackage    PACKAGE
                          ATTRIBUTES
                          sourceIndicator GET;

REGISTERED AS  {smi2Package 28};
```

## 8.12 Weekly scheduling

The semantics of **weeklyScheduling** package are described in CCITT Rec.X.734 | ISO/IEC 10164-5 and CCITT Rec.X.735 | ISO/IEC 10164-6.

```
weeklyScheduling  PACKAGE
                  ATTRIBUTES
                  weekMask      REPLACE-WITH-DEFAULT
                                DEFAULT VALUE Attribute-ASN1Module.defaultWeekMask
                                GET-REPLACE ADD-REMOVE;

REGISTERED AS  {smi2Package 29};
```

# 9 Definition of generic attribute types

This Recommendation | International Standard defines a number of generic attribute types. These are intended to be used in the definition of specific attributes appearing in the definitions of Managed Object Classes.

Each attribute type definition comprises

- the structure of the attribute value;
- the inherent properties of the attribute in terms of its behaviour;
- the permitted matching operations that may be performed on an attribute having this type;
- the ways in which specific attributes having this type relate to other attributes.

An attribute type definition does not include an object identifier. This shall form part of the definition of each of the specific attributes of that type. Definitions of specific attributes may also refine the definition given in the type, for example

- to extend the behaviour definition to relate the behaviour to the operation of the resource represented by the managed object;
- parameters be associated with the attribute.

The template definitions are specified below for each attribute's type. The ASN.1 supporting productions are defined in 14.2.

## 9.1 Counter

The semantics of the **counter** attribute type are defined below.

Counters are a management abstraction of an underlying counting process. Two types of counters are modelled to meet different needs. The non-settable (or simple) counter is defined to operate autonomously, i.e. not subject to change by Management operation, in order to enable different managing stations to access its information without interference. The settable counter, in contrast, is defined to allow setting or resetting by Management operation, and therefore is more suited to use by a single managing station. Both types are defined as attributes.

A counter is regarded as associated with some internal event, which may be, but is not in general, a defined event represented in management information. The current value is incremented by 1 when this event occurs. It can take any values in its range. When a counter reaches its maximum value, it wraps around and counts upwards from 0; overflow information is not in general retained. An attribute value change notification may be defined for counters with wrap behaviour.

NOTE – The rule that the counter value can increase only in steps of 1 is a descriptive convention that simplifies the description of counter-threshold in 9.3.1. It does not imply that it will always be possible to observe each value in the counter's range, since the events counted may occur in rapid succession.

**Not-settable counter-definition**

attribute value

- single value.

inherent properties

- current value is a non-negative integer;
- it has a maximum value (see specification properties);
- the counting direction is up, with increment 1;
- the current value wraps around when it reaches its maximum;
- the initial value is 0.

permitted operations

- Get only.

implicit relations

- directly related to a single counter-threshold where applied;
- as an implementation option, may trigger a defined event when it wraps.

specification properties

- the internal event that is counted;
- maximum value;
- "estimated" wrap around period, to indicate necessary reading rate.

**Settable counter definition**

attribute value

- single value.

inherent properties

- current value is a non-negative integer;
- it has a maximum value (see specification properties);
- the counting direction is up, with increment 1;

- the current value wraps around when it reaches its maximum;
- the initial value is 0.

permitted operations

- Get;
- Set to arbitrary value (within range);
- Set to the default specified in the managed object definition, or to zero otherwise.

implicit relations

- directly related to a single counter-threshold, where applied;
- as an implementation option, may trigger a defined event when it wraps or its value is otherwise changed except by the normal counting process.

specification properties

- the internal event that is counted;
- maximum value;
- "estimated" wrap around period, to indicate necessary reading rate.

```
counter         ATTRIBUTE
                WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.Count;
                MATCHES FOR EQUALITY,ORDERING ;;
```

## 9.2 Gauge

The semantics of the **gauge** attribute type is defined below.

The gauge is the management abstraction of the value of a dynamic variable, such as the number of connections currently operated by a protocol machine or the rate of change of a traffic counter. There is no restriction on what the dynamic variable may be, within the constraints set out below.

The value of the gauge is subject to change in either direction. The value of the increment or decrement is unconstrained, except that changes that would take the gauge value beyond its maximum or minimum leave the gauge value at its maximum or minimum respectively.

In order to support use by multiple managing stations, gauges are defined to be read-only.

attribute value

- single value.

inherent properties

- current value is a non-negative integer or real (see specification properties);
- it has a maximum and a minimum value (see specification properties);
- it may increase or decrease by arbitrary amounts;
- it does not wrap around.

permitted actions

- Get only.

implicit relations

- directly related to a tide-mark and or gauge-threshold, where applied;
- only one minimum and one maximum tide-mark may be applied;
- only one (possibly multi-level) threshold may be applied;
- can be used to measure other management information.

specification properties

- the dynamic variable measured, with its type: integer or real;
- maximum and minimum value.

```
gauge           ATTRIBUTE
                WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ObservedValue;
                MATCHES FOR EQUALITY, ORDERING ;;
```

## 9.3 Threshold

A threshold is the general mechanism for generating a defined notification from changes in numeric attributes values. Two types of thresholds are defined, for counters and for gauges, because the required behaviour is significantly different in the two cases. Both allow attribute values that are of integer data types. Other types of threshold may be defined in the future.

### 9.3.1 Counter-threshold

The **counter-Threshold** is the general mechanism for generating a defined notification from changes in any value of attribute type counter.

The Counter-Threshold is related to a defined notification. It has the basic property that the defined notification is triggered when the value of the count becomes equal to the comparison level of the threshold. The definition also allows for more complex styles of operation when needed: the comparison attribute is in general a set of levels, for example to represent different degrees of severity of a fault condition, and the notification is triggered whenever the count value reaches any of the levels.

In addition, an offset mechanism is also available to allow particular counting intervals to be detected, as follows. If the offset value is not zero, whenever the threshold is triggered by the counter value reaching a comparison level, that comparison level is incremented by the offset value. This is regarded as taking place instantaneously, i.e. before the count is incremented. Thus, for each level, the threshold triggers a notification every time the count increases by an interval equal to the offset value. When the comparison value exceeds the modulus of the counter, it wraps around.

For a counter-threshold, the time at which the threshold offset was last applied, or the time at which the count was last initialized is the arm-time.

The Counter-Threshold is modelled as a set-valued attribute.

value type

- comparison level, integer;
- offset value, integer;
- notifications on/off switch, boolean.

inherent properties

- the comparison levels are non-negative integers;
- the offset values are non-negative integers;
- the notifications switch is on or off.

permitted operations

- Get, Set, Add, Remove.

implicit relations

- directly related to a single count;
- directly related to a defined notification.

specification properties

- count to which it applies;
- the defined notification that may be triggered.

Any relationship between a threshold attribute and any underlying mechanism is specified as part of the behaviour of the managed object class containing the threshold.

```
counter-Threshold   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX
        Attribute-ASN1Module.CounterThreshold;
         MATCHES FOR EQUALITY ;;
```

### 9.3.2 Gauge-threshold

The **gauge-Threshold** is the general mechanism for generating one or more defined notifications from changes in any value of attribute type gauge. A hysteresis mechanism is provided to avoid the repeated triggering of event notifications when the gauge makes small oscillations around a threshold value. This capability is provided by specifying threshold values in pairs; one being a high threshold value and the other being a low threshold value. The difference between threshold values is the hysteresis interval.

The Gauge-Threshold is a set-valued attribute used to define threshold levels associated with notifications. As a set valued attribute, it may have zero or more members. Each member consists of a sequence of two submembers, notifyLow and notifyHigh. Each of these submembers in turn has a structure and consists of a sequence of a gauge value and its associated on/off switch.

**notifyHigh**: This structured submember defines the value that the gauge must reach or exceed in order to optionally generate notification, together with a switch value that controls the generation of the notification.

**notifyLow:** This structured submember defines the value that the gauge must fall to or below in order to optionally generate notification, together with a switch value that controls the generation of the notification.

Gauge-Thresholds have the constraint that notifyHigh has a gauge value that is greater than or equal to that of notifyLow.

Gauge-Thresholds have the following constraints

- notify-high's gauge value and notify-low's threshold value are of the same type as the gauge;
- notify-high's threshold value is greater or equal to notify-low's threshold value;
- notify-high's on/off switch and notify-low's on/off switch are of type boolean.

Gauge-Thresholds have the following behaviour

- initially, if notify-high's on/off switch is true and the gauge value becomes equal to or greater than notify-high's gauge value, in a positive going direction, then the defined event notification is triggered; subsequent crossings of notify-high's gauge value shall not cause further generation of event reports unless the gauge value becomes equal to or less than notify-low's threshold value.
- initially, if notify-low's on/off switch is true and the gauge value becomes equal to or less than notify-low's gauge value, in a negative going direction, then the defined event notification is triggered; subsequent crossings of notify-low's gauge value shall not cause further generation of event reports unless the gauge value becomes equal to or greater than notify-high's gauge value.

For either case, the time at which the notification is again enabled is defined as the arm-time.

Any relationship between a threshold attribute and any underlying mechanism is specified as part of the behaviour of the managed object class containing the threshold.

```
gauge-Threshold     ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.GaugeThreshold;
        MATCHES FOR EQUALITY ;;
```

### 9.4 Tide-mark

The semantics of the **TideMark** attribute type is defined below.

The tide-mark is a mechanism that records the maximum or minimum value reached by a gauge during a measurement period. A tide-mark is essentially read-only, except that it may be reset to a desired value which is the current value of the associated gauge. Each individual tide-mark is specified to be either a maximum tide-mark or a minimum tide-mark, and therefore moves (other than when it is reset) only up or down respectively. Thus, a maximum tide-mark changes (increases) only when its gauge increases beyond the current tide-mark value, and a minimum tide-mark changes (decreases) only when its gauge moves below the current tide-mark value.

A tide-mark is defined as a set-valued attribute with three components. Two value components are defined, the current value and the value immediately prior to the last reset, so as to support different measurement requirements. The third component is the last reset time.

attribute value

- current value of tide-mark;
- former value of tide-mark, i.e. value immediately before the last reset;

– last reset time.

inherent properties

– it is associated with a gauge;
– it has a direction (maximum or minimum);
– the current and former values are integer or real depending upon the associated gauge.

permitted operations

– Get returns the entire attribute value, viz. the current value, the former value and the last reset time;
– Set to default sets the former value to the current value,the current value to the value of the associated gauge,and the last reset time to the current time.

implicit relations

– directly related to a gauge;
– may be directly related to a defined event that is triggered when the current value changes.

specification properties

– gauge to which it applies;
– direction (maximum or minimum).

```
tideMark            ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.TideMarkInfo;
        MATCHES FOR EQUALITY ;;
```



## 10 Definition of specific attribute types

This Recommendation | International Standard defines a number of attributes which are referenced by the System Management Functions of CCITT Recs. X.730 - 736 | ISO/IEC 10164 Parts 1 to 7, or which are of widespread applicability in Managed Object definitions. Unlike the attribute types defined in clause 9, the definitions here are directly useable in the definitions of Managed Objects without further elaboration. In particular they include an object identifier by which the attribute is identified. It follows that any of these attributes can only appear once in a Managed Object. Uses of these attributes in Managed Objects may also refine the definition, for example

– to extend the behaviour definition to relate the behaviour to the operation of the resource represented by the Managed Object;
– to restrict the values which the attribute may take to a subset of those defined by the attribute type;
– to require support for some values of the type;
– to define relations (such as mutual constraints on permitted values) with other attributes.

The template definition along with the allocated object identifier are specified below for each attributes type. The ASN.1 supporting productions are defined in 13.2.

### 10.1 Attributes types used for naming

The following attributes can be used as naming attributes in a relative distinguished name.

#### 10.1.1 Discriminator Id

The **discriminatorId** attribute type is used in naming instances of Discriminator managed object class.

```
discriminatorId     ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SimpleNameType;
        MATCHES FOR Equality, Substrings, ORDERING;
        BEHAVIOUR
        rDNIdBehaviour;

REGISTERED AS  {smi2AttributeID 1};
```

```
rDNIdBehaviour     BEHAVIOUR
        DEFINED AS " If the string choice for the syntax is used matching on the substrings is permitted . If the
        number choice for the syntax is used then matching on ordering is permitted.";
```

### 10.1.2 Log Id

The **logId** attribute type is used in naming instances of Log managed object class.

```
logId              ATTRIBUTE
                   WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SimpleNameType;
                   MATCHES FOR EQUALITY, SUBSTRINGS;
                   BEHAVIOUR
                   rDNIdBehaviour;

REGISTERED AS  {smi2AttributeID 2};
```

### 10.1.3 Log record Id

The **logRecordId** attribute type is used in naming instances of Log Records managed object class.

```
logRecordId        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.LogRecordId;
        MATCHES FOR EQUALITY, ORDERING ;

REGISTERED AS  {smi2AttributeID 3};
```

### 10.1.4 System Id

The **systemId** attribute type may be used in naming instances of System managed object class.

```
systemId           ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SystemId;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 4};
```

### 10.1.5 System title

The systemTitle attribute type may be used in naming instances of System managed object class.

```
systemTitle        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SystemTitle;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 5};
```

## 10.2 Counter

This Recommendation | International Standard defines a number of counter types which are referenced by the System Management Functions in CCITT Recs. X.730-736 | ISO/IEC 10164 Parts 1 to 7.

NOTE – There are no counter types currently defined in the system management functions and this subclause is a place holder for future additions. Annex A defines a number of specific counter attributes generally useful to layer management.

## 10.3 Counter-threshold

This Recommendation | International Standard defines a number of counter-threshold types which are referenced by the System Management Functions in CCITT Recs. X.730-736 | ISO/IEC 10164 Parts 1 to 7.

NOTE – There are no counter-threshold types currently defined in the system management functions and this subclause is a place holder for future additions. Annex A defines a number of specific counter-threshold attributes generally useful to layer management.

## 10.4 Gauge

This Recommendation | International Standard defines a number of Gauge types which are referenced by the System Management Functions in CCITT Recs. X.730-736 | ISO/IEC 10164 Parts 1 to 7.

NOTE – There are no Gauge types currently defined in the system management functions and this subclause is a place holder for future additions.

## 10.5 Gauge-threshold

This Recommendation | International Standard defines a number of Gauge-Threshold types which are referenced by the System Management Functions in CCITT Recs. X.730-736 | ISO/IEC 10164 Parts 1 to 7.

NOTE – There are no gauge threshold types currently defined in the system management functions and this subclause is a place holder for future addition.

## 10.6 Tide-mark

This Recommendation | International Standard defines a number of Tide-Mark types which are referenced by the System Management Functions of CCITT Recs. X.730-736 | ISO/IEC 10164 Parts 1 to 7.

NOTE – There are no tide-mark types currently defined in the system management functions and this subclause is a place holder for future addition.

## 10.7 Miscellaneous attribute types

### 10.7.1 Events related

#### 10.7.1.1 Additional information

The semantics of the **additionalInformation** attribute type are specified in the Additional Information parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
additionalInformation          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AdditionalInformation;
        BEHAVIOUR
        additionalInformationBehaviour    BEHAVIOUR
        DEFINED AS      "This attribute is used to supply additional information in notifications.";;

REGISTERED AS   {smi2AttributeID 6};
```

#### 10.7.1.2 Additional text

The semantics of the **additionalText** attribute type are specified in the Additional Text parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
additionalText      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AdditionalText;
        MATCHES FOR EQUALITY, SUBSTRINGS;
        BEHAVIOUR
        additionalTextBehaviour   BEHAVIOUR
        DEFINED AS      "This attribute is used to specify additional textual information in notifications ";;

REGISTERED AS   {smi2AttributeID 7};
```

#### 10.7.1.3 Attribute identifier list

The semantics of the **attributeIdentifierList** attribute type are specified in the Attribute Identifier List parameter in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
attributeIdentifierList          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AttributeIdentifierList;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION ;
        BEHAVIOUR
        attributeIdentifierListBehaviour     BEHAVIOUR
        DEFINED AS      "This attribute contains a list of attribute identifiers.";;
```

REGISTERED AS {smi2AttributeID 8};

#### 10.7.1.4 Attribute list

The semantics of the **attributeList** attribute type are specified in the Attribute List parameter in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
attributeList          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AttributeList;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION ;
        BEHAVIOUR
        attributeListBehaviour      BEHAVIOUR
        DEFINED AS      "This attribute contains a list of attribute identifiers and their values.";;

REGISTERED AS {smi2AttributeID 9};
```

#### 10.7.1.5 Attribute value change definition

The semantics of the **attributeValueChangeDefinition** attribute type are defined in the Attribute Value Change definition parameter in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
attributeValueChangeDefinition      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AttributeValueChangeDefinition;
        BEHAVIOUR
        attributeValueChangeDefinitionBehaviour    BEHAVIOUR
        DEFINED AS      "This attribute contains a set of attribute identifiers and their old and new values.";;

REGISTERED AS {smi2AttributeID 10};
```

#### 10.7.1.6 Backed-up status

The semantics of the **backedUpStatus** attribute type are specified in the Backed-up Status parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
backedUpStatus    ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.BackedUpStatus;
        MATCHES FOR EQUALITY ;

REGISTERED AS {smi2AttributeID 11};
```

#### 10.7.1.7 Correlated notifications

The semantics of the **correlatedNotifications** attribute type are specified in the Correlated Notifications parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
correlatedNotifications       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.CorrelatedNotifications;
        BEHAVIOUR
        correlatedNotificationsBehaviour   BEHAVIOUR
        DEFINED AS      "This attribute contains a set of notification identifiers and, if necessary, their associated
        managed object instance names, for the notifications that are considered to be correlated to this notification.";;

REGISTERED AS {smi2AttributeID 12};
```

#### 10.7.1.8 Event time

The semantics of the **eventTime** attribute type are specified in the Event Time parameter in CCITT Rec. X.710 | ISO/IEC 9595.

```
eventTime          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.EventTime;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR       timeOrdering;

REGISTERED AS {smi2AttributeID 13};

timeOrdering       BEHAVIOUR
```

DEFINED AS
"The year, month, day, hour, minute and seconds field are compared in order to determine whether the specified value is greater or less than the value of the attribute. The values for the year, month,day, hour,minute and seconds are determined from the character string representation and the year value is first compared. If equal the month value is compared and this process is continued if the compared fields are equal";

#### 10.7.1.9 Event type

The semantics of the **eventType** attribute type are specified in the Event type parameter in CCITT Rec. X.710 | ISO/IEC 9595.

```
eventType          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.EventTypeId;
        MATCHES FOR EQUALITY ;

REGISTERED AS  {smi2AttributeID 14};
```

#### 10.7.1.10 Monitored attributes

The semantics of the **monitoredAttributes** attribute type are specified in the Monitored Attributes parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
monitoredAttributes          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.MonitoredAttributes;

REGISTERED AS  {smi2AttributeID 15};
```

#### 10.7.1.11 Notification identifier

The semantics of the **notificationIdentifier** attribute type are specified in the Notification identifier parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
notificationIdentifier ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.NotificationIdentifier;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        notificationIdentifierBehaviour      BEHAVIOUR
        DEFINED AS      "This attribute contains a unique identifier for the notification, which may be present in the
        correlated notifications attribute of other notifications.";;

REGISTERED AS   {smi2AttributeID 16};
```

#### 10.7.1.12 Perceived severity

The semantics of the **perceivedSeverity** attribute type are specified in the Perceived Severity parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
perceivedSeverity   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.PerceivedSeverity;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 17};
```

#### 10.7.1.13 Probable cause

The semantics of the **probableCause** attribute type are specified in the Probable Cause parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
probableCause      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ProbableCause;
        MATCHES FOR EQUALITY ;

REGISTERED AS  {smi2AttributeID 18};
```

### 10.7.1.14 Proposed repair actions

The semantics of the **proposedRepairActions** attribute type are specified in the Proposed Repair Action parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
proposedRepairActions      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ProposedRepairActions;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 19};
```

### 10.7.1.15 Relationship change definition

The semantics of the **relationshipChangeDefinition** attribute type are defined in the Relationship change definition parameter in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
relationshipChangeDefinition        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AttributeValueChangeDefinition;
        BEHAVIOUR
        relationshipChangeDefinitionBehaviour      BEHAVIOUR
        DEFINED AS      "This attribute contains a set of relationship attribute identifiers and their old and new
        values.";;

REGISTERED AS  {smi2AttributeID 20};
```

### 10.7.1.16 Security alarm cause

The semantics of the **securityAlarmCause** attribute type are specified in the Security alarm cause parameter in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
securityAlarmCause          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SecurityAlarmCause;
        MATCHES FOR EQUALITY ;
        BEHAVIOUR
        securityAlarmCauseBehaviour      BEHAVIOUR
        DEFINED AS      "This attribute specifies the cause of the security alarm";;

REGISTERED AS  {smi2AttributeID 21};
```

### 10.7.1.17 Security alarm detector

The semantics of the **securityAlarmDetector** attribute type are specified in the Security alarm Detector parameter in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
securityAlarmDetector       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SecurityAlarmDetector;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        securityAlarmDetectorBehaviour   BEHAVIOUR
        DEFINED AS      "This attribute identifies the entity that detected the security alarm";;

REGISTERED AS  {smi2AttributeID 22};
```

### 10.7.1.18 Security alarm severity

The semantics of the **securityAlarmSeverity** attribute type are specified in the Security alarm severity parameter in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
securityAlarmSeverity       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SecurityAlarmSeverity;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        securityAlarmSeverityBehaviour   BEHAVIOUR
        DEFINED AS      "This attribute indicates the severity of the security alarm";;

REGISTERED AS  {smi2AttributeID 23};
```

#### 10.7.1.19 Service provider

The semantics of the **serviceProvider** attribute type are specified in the Service provider parameter in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
serviceProvider      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ServiceProvider;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        serviceProviderBehaviour BEHAVIOUR
        DEFINED AS      "This attribute contains information about the service provider associated with the service
        request that caused the security alarm";;

REGISTERED AS  {smi2AttributeID 24};
```

#### 10.7.1.20 Service user

The semantics of the **serviceUser** attribute type are specified in the Service user parameter in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
serviceUser          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ServiceUser;
        MATCHES FOR EQUALITY ;
        BEHAVIOUR
        serviceUserBehaviour      BEHAVIOUR
        DEFINED AS      "This attribute contains information about the service user associated with the service
        request that caused the security alarm";;

REGISTERED AS  {smi2AttributeID 25};
```

#### 10.7.1.21 Source indicator

The semantics of the **sourceIndicator** attribute type are defined in the Source Indicator parameter in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
sourceIndicator      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SourceIndicator;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        sourceIndicatorBehaviour BEHAVIOUR
        DEFINED AS      "This attribute is used in notifications to indicate the source of the operation that resulted in
        generating the notification";;

REGISTERED AS  {smi2AttributeID 26};
```

#### 10.7.1.22 Specific problems

The semantics of the **specificProblems** attribute type are specified in the Specific Problems parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
specificProblems     ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SpecificProblems;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION ;

REGISTERED AS  {smi2AttributeID 27};
```

#### 10.7.1.23 State change definition

The semantics of the stateChangeDefinition attribute type are defined in the State change definition parameter in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
stateChangeDefinition        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AttributeValueChangeDefinition;
        BEHAVIOUR
        stateChangeDefinitionBehaviour    BEHAVIOUR
        DEFINED AS      "This attribute contains a set of state attribute identifiers and their old and new values.";;

REGISTERED AS  {smi2AttributeID 28};
```

#### 10.7.1.24 Threshold info

The semantics of the **thresholdInfo** attribute type are specified in the Threshold Info parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
thresholdInfo        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ThresholdInfo;

REGISTERED AS  {smi2AttributeID 29};
```

#### 10.7.1.25 Trend indication

The semantics of the **trendIndication** attribute type are specified in the Trend Indication parameter in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
trendIndication      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.TrendIndication;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 30};
```

### 10.7.2 States related

#### 10.7.2.1 Administrative state

The semantics of the **administrativeState** attribute type are specified in the Administrative State attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
administrativeState ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AdministrativeState;
        MATCHES FOR EQUALITY ;

REGISTERED AS  {smi2AttributeID 31};
```

#### 10.7.2.2 Alarm status

The semantics of the **alarmStatus** attribute type are specified in the Alarm Status attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
alarmStatus          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AlarmStatus;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION ;

REGISTERED AS  {smi2AttributeID 32};
```

#### 10.7.2.3 Availability status

The semantics of the **availabilityStatus** attribute type are specified in the Availability Status attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
availabilityStatus   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.AvailabilityStatus;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION ;

REGISTERED AS  {smi2AttributeID 33};
```

#### 10.7.2.4 Control status

The semantics of the **controlStatus** attribute type are specified in the Control Status attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
controlStatus        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ControlStatus;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION ;

REGISTERED AS  {smi2AttributeID 34};
```

#### 10.7.2.5 Operational state

The semantics of the **operationalState** attribute type are specified in the Operational State attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
operationalState    ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.OperationalState;
        MATCHES FOR EQUALITY ;

REGISTERED AS  {smi2AttributeID 35};
```

#### 10.7.2.6 Procedural status

The semantics of the **proceduralStatus** attribute type are specified in the Procedural Status attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
proceduralStatus    ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ProceduralStatus;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 36};
```

#### 10.7.2.7 Standby status

The semantics of the **standbyStatus** attribute type are specified in the Standby Status attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
standbyStatus       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.StandbyStatus;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 37};
```

#### 10.7.2.8 Unknown status

The semantics of the **unknownStatus** attribute type are specified in the Unknown Status attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
unknownStatus       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.UnknownStatus;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 38};
```

#### 10.7.2.9 Usage state

The semantics of the **usageState** attribute type are specified in the Usage State attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
usageState          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.UsageState;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 39};
```

#### 10.7.2.10 State

The semantics of the **State** group attribute type are specified in the State attribute group in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
state                ATTRIBUTE GROUP
        DESCRIPTION " This is defined as an empty attribute group. The elements of this group are composed of
        state attributes in the managed object. The state attributes may include those specified in CCITT Rec.
        X.731|ISO/IEC 10164-2 and others that are specific to the managed object class.";

REGISTERED AS  {smi2AttributeGroup 1};
```

### 10.7.3 Relationships related

#### 10.7.3.1 Back-up object

The semantics of the **backUpObject** attribute type are specified in Back-Up Object relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3. This attribute is also used in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
backUpObject        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.BackUpRelationshipObject;
                MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 40};
```

#### 10.7.3.2 Backed-up object

The semantics of the **backedUpObject** attribute type are specified in Backed-Up Object relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
backedUpObject    ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.BackUpRelationshipObject;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 41};
```

#### 10.7.3.3 Member

The semantics of **member** attribute type are specified in Member relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
member              ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.GroupObjects;
        MATCHES FOR EQUALITY,SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 42};
```

#### 10.7.3.4 Owner

The semantics of **owner** attribute type are specified in Owner relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
owner               ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.GroupObjects;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 43};
```

#### 10.7.3.5 Peer

The semantics of **peer** attribute type are specified in Peer relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
peer                ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.BackUpRelationshipObject;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 44};
```

#### 10.7.3.6 Primary

The semantics of **primary** attribute type are specified in Primary relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
primary                ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.PrioritisedObject;

REGISTERED AS  {smi2AttributeID 45};
```

#### 10.7.3.7 Provider object

The semantics of **providerObject** attribute type are specified in Provider Object relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
providerObject      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.PrioritisedObject;

REGISTERED AS  {smi2AttributeID 46};
```

#### 10.7.3.8 Relationships

The semantics of the **relationships** group attribute type are specified in the Relationships group attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
relationships          ATTRIBUTE GROUP
        DESCRIPTION " This is defined as an empty attribute group. The elements of this group are composed of
        relationship attributes in the managed object. The relationship attributes may include those specified in CCITT
        Rec. X.732 | ISO/IEC 10164-3 and others that are specific to the managed object class.";

REGISTERED AS  {smi2AttributeGroup 2};
```

#### 10.7.3.9 Secondary

The semantics of **secondary** attribute type are used in defining Secondary relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
secondary              ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.PrioritisedObject;

REGISTERED AS  {smi2AttributeID 47};
```

#### 10.7.3.10 User object

The semantics of **userObject** attribute type are used in defining User Object Relationship attribute in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
userObject             ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.PrioritisedObject;

REGISTERED AS  {smi2AttributeID 48};
```

### 10.7.4 Other attribute types

#### 10.7.4.1 Active destination

The semantics of the **activeDestination** attribute type are specified in the Active destination attribute in CCITT Rec. X.734 | ISO/IEC 10164-5.

```
activeDestination   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ActiveDestination;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 49};
```

#### 10.7.4.2 Allomorphs

The **allomorphs** attribute appears in a managed object if and only if that managed object supports allomorphism.

```
allomorphs          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.Allomorphs;
         MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 50};
```

#### 10.7.4.3 Back-up destination list

The semantics of the **backUpDestinationList** attribute type are specified in the Backup destination list attribute in CCITT Rec. X.734 | ISO/IEC 10164-5.

```
backUpDestinationList       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.BackUpDestinationList;
        MATCHES FOR EQUALITY ,SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 51};
```

#### 10.7.4.4 Capacity alarm threshold

The semantics of the **capacityAlarmThreshold** attribute type are specified in the Capacity Alarm Threshold attribute in CCITT Rec. X.735 | ISO/IEC10164-6.

```
capacityAlarmThreshold     ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.CapacityAlarmThreshold;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 52};
```

#### 10.7.4.5 Confirmed mode

The semantics of the **confirmedMode** attribute type are specified in the Confirmed mode attribute in CCITT Rec. X.734 | ISO/IEC10164-5.

```
confirmedMode      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ConfirmedMode;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 53};
```

#### 10.7.4.6 Current log size

The semantics of the **currentLogSize** attribute type are specified in the Current Log Size attribute in CCITT Rec. X.735 | ISO/IEC 10164-6.

```
currentLogSize      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.CurrentLogSize;
        MATCHES FOR EQUALITY, ORDERING;

REGISTERED AS  {smi2AttributeID 54};
```

#### 10.7.4.7 Destination

The semantics of the **destination** attribute type are specified in the Destination Address attribute in CCITT Rec. X.734 | ISO/IEC 10164-5.

```
destination          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.Destination;
        MATCHES FOR EQUALITY ;

REGISTERED AS  {smi2AttributeID 55};
```

#### 10.7.4.8 Discriminator construct

The semantics of the **discriminatorConstruct** attribute type are specified in the Discriminator Construct attribute in CCITT Rec. X.734 | ISO/IEC 10164-5.

```
discriminatorConstruct      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.DiscriminatorConstruct;

REGISTERED AS  {smi2AttributeID 56};
```

#### 10.7.4.9 Intervals of day

The semantics of the **intervalsOfDay** attribute type are specified in the Daily Scheduling package in CCITT Rec. X.734 | ISO/IEC 10164-5 and CCITT Rec. X.735 | ISO/IEC 10164-6.

```
intervalsOfDay      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.IntervalsOfDay;

REGISTERED AS  {smi2AttributeID 57};
```

#### 10.7.4.10 Log full action

The semantics of the **logFullAction** attribute type are specified in the Log Full Action attribute in CCITT Rec. X.735 | ISO/IEC 10164-6.

```
logFullAction       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.LogFullAction;
        MATCHES FOR EQUALITY ;

REGISTERED AS  {smi2AttributeID 58};
```

#### 10.7.4.11 Logging time

The semantics of the **loggingTime** attribute type are specified in the Logging Time attribute in CCITT Rec. X.735 | ISO/IEC 10164-6.

```
loggingTime         ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.LoggingTime;
        MATCHES FOR EQUALITY, ORDERING;

REGISTERED AS  {smi2AttributeID 59};
```

#### 10.7.4.12 Managed object class

The **managedObjectClass** attribute type is specified to allow filtering of the Managed Object Class parameter when events reports are logged as records.

```
managedObjectClass      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ObjectClass;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 60};
```

#### 10.7.4.13 Managed object instance

The **managedObjectInstance** attribute type is specified to allow filtering of the Managed Object Instance parameter when events reports are logged as records.

```
managedObjectInstance   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ObjectInstance;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 61};
```

#### 10.7.4.14 Max log size

The semantics of the **maxLogSize** attribute type are specified in the Max Log Size attribute in CCITT Rec. X.735 | ISO/IEC 10164-6.

```
maxLogSize          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.MaxLogSize;
        MATCHES FOR EQUALITY, ORDERING ;
```

```
        BEHAVIOUR
        maxSizeOrderingBehaviour        BEHAVIOUR
        DEFINED AS " The ordering is the same as for sequentially increasing postive integers except that a value of
        zero is largest and denotes infinite size.";;

REGISTERED AS  {smi2AttributeID 62};
```

### 10.7.4.15 Name binding

The **nameBinding** attribute appears in every managed object and identifies the name binding with which the managed object was instantiated. The semantics of this attribute are described in CCITT Rec. X.720 | ISO/IEC 10165-1.

```
nameBinding        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.NameBinding;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 63};
```

### 10.7.4.16 Number of records

The semantics of the **numberOfRecords** attribute type are specified in the Number of Records attribute in CCITT Rec. X.735 | ISO/IEC 10164-6.

```
numberOfRecords  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.NumberOfRecords;
        MATCHES FOR EQUALITY, ORDERING ;

REGISTERED AS  {smi2AttributeID 64};
```

### 10.7.4.17 Object class

The **objectClass** attribute appears in every managed object and indicates the managed object class to which the managed object belongs. The semantics of ordering on object classes are described in CCITT Rec. X.720 | ISO/IEC 10165-1.

```
objectClass        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ObjectClass;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 65};
```

### 10.7.4.18 Packages

The **packages** attribute appears in every managed object in which any registered package, other than the packagePackage has been instantiated.

```
packages           ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.Packages;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;

REGISTERED AS  {smi2AttributeID 66};
```

### 10.7.4.19 Scheduler name

The semantics of **schedulerName** attribute type are specified in the External Scheduler scheduling package in CCITT Rec. X.734 | ISO/IEC 10164-5 and CCITT Rec. X.735 | ISO/IEC 10164-6.

```
schedulerName      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.ObjectInstance;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 67};
```

### 10.7.4.20 Start time

The semantics of **startTime** attribute type are specified in the Weekly Scheduling package in CCITT Rec. X.734 | ISO/IEC 10164-5 and CCITT Rec. X.735 | ISO/IEC 10164-6.

```
startTime          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.StartTime;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR timeOrdering;

REGISTERED AS  {smi2AttributeID 68};
```

#### 10.7.4.21 Stop time

The semantics of **stopTime** attribute type are specified in the Weekly Scheduling package in CCITT Rec. X.734 | ISO/IEC 10164-5 and CCITT Rec. X.735 | ISO/IEC 10164-6. The matching criterion "Ordering" is applicable only when the ASN.1 type GeneralizedTime choice is selected for StopTime.

```
stopTime           ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.StopTime;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR timeOrdering;

REGISTERED AS  {smi2AttributeID 69};
```

#### 10.7.4.22 Supported Features

The semantics of the **supportedFeatures** attribute type are given below.

```
supportedFeatures ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.SupportedFeatures;
        MATCHES FOR SET-COMPARISON, SET-INTERSECTION, EQUALITY;
        BEHAVIOUR
        supportedFeaturesBehaviour      BEHAVIOUR
        DEFINED AS      "This attribute is used to identify features within the system that are capable of being
        managed. The registration of each feature is performed by the development authority for the feature and will
        typically identify a specification in which the details of what can be managed are provided";;

REGISTERED AS  {smi2AttributeID 70};
```

#### 10.7.4.23 Week mask

The semantics of **weekMask** attribute type are specified in the Weekly Scheduling package in CCITT Rec. X.734 | ISO/IEC 10164-5 and CCITT Rec. X.735 | ISO/IEC 10164-6.

```
weekMask           ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.WeekMask;
        MATCHES FOR EQUALITY;

REGISTERED AS  {smi2AttributeID 71};
```

## 11 Definition of action types

This Recommendation | International Standard defines a number of action types which are referenced by the System Management Functions in CCITT Recs. X.730-736 | ISO/IEC 10164 Parts 1 to 7.

NOTE – There are no action types currently defined in the system management functions and this section is a place holder for future addition.

## 12 Definition of parameters

This Recommendation | International Standard defines a number of parameters which are referenced by the System Management Functions in CCITT Recs. X.730-736 | ISO/IEC 10164 Parts 1 to 7. The definition of parameter template is specified in CCITT Rec. X.722 | ISO/IEC 10165-4.

The **miscellaneousError** parameter is included in this Recommendation | International Standard as a possible specific error when processing failure is reported using CMIP linked reply mechanism. This may be imported in any managed object class definition .

```
miscellaneousError PARAMETER
        CONTEXT         SPECIFIC-ERROR;
        WITH SYNTAX Parameter-ASN1Module.MiscellaneousError;
        BEHAVIOUR
        miscellaneousErrorBehaviour     BEHAVIOUR
        DEFINED AS " When a processing error failure has occurred and the error condition encountered does not
        match any of the object's defined specific error types, this value is used.";;

REGISTERED AS  {smi2Parameter 1};
```

## 13 Definition of notification types

This Recommendation | International Standard defines a number of notification types which are applicable to a wide variety of managed object classes.

Each notification type definition comprises

- the structure of the notification data carried in the management protocol;
- the behaviour of the notification;
- the structure of the notification result data carried in the management protocol;
- assignment of an object identifier value.

The template definition along with the allocated object identifier are specified below for each notification type. The ASN.1 supporting productions are defined in 14.3.

### 13.1 Attribute value change

The semantics of the **attributeValueChange** notification type are specified in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
attributeValueChange        NOTIFICATION
        BEHAVIOUR       attributeValueChangeBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.AttributeValueChangeInfo
                AND ATTRIBUTE IDS
                sourceIndicator             sourceIndicator,
                attributeIdentifierList     attributeIdentifierList,
                attributeValueChangeDefinition    attributeValueChangeDefinition,
                notificationIdentifier      notificationIdentifier,
                correlatedNotifications     correlatedNotifications,
                additionalText              additionalText,
                additionalInformation       additionalInformation;

REGISTERED AS  { smi2Notification 1};

attributeValueChangeBehaviour
BEHAVIOUR
        DEFINED AS "This notification type is used to report changes to the attribute such as addition or deletion of
        members to one or more set valued attributes, replacement of the value of one or more attributes and setting
        attribute values to their defaults.";
```

### 13.2 Communications alarm

The semantics of the **communicationsAlarm** notification type are specified in CCITT Rec. X.733 |ISO/IEC 10164-4.

```
communicationsAlarm         NOTIFICATION
        BEHAVIOUR       communicationsAlarmBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.AlarmInfo
                AND ATTRIBUTE IDS
                probableCause           probableCause,
                specificProblems        specificProblems,
                perceivedSeverity       perceivedSeverity,
                backedUpStatus          backedUpStatus,
                backUpObject            backUpObject,
                trendIndication         trendIndication,
                thresholdInfo           thresholdInfo,
                notificationIdentifier  notificationIdentifier,
```

```
                correlatedNotifications   correlatedNotifications,
                stateChangeDefinition     stateChangeDefinition,
                monitoredAttributes       monitoredAttributes,
                proposedRepairActions     proposedRepairActions,
                additionalText            additionalText,
                additionalInformation     additionalInformation;

REGISTERED AS  {smi2Notification 2};

communicationsAlarmBehaviour
BEHAVIOUR
        DEFINED AS "This notification type is used to report when the object detects a communications error.";
```

## 13.3 Environmental alarm

The semantics of the **environmentalAlarm** notification type are specified in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
environmentalAlarm          NOTIFICATION
        BEHAVIOUR           environmentalAlarmBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.AlarmInfo
                AND ATTRIBUTE IDS
                probableCause             probableCause,
                specificProblems          specificProblems,
                perceivedSeverity         perceivedSeverity,
                backedUpStatus            backedUpStatus,
                backUpObject              backUpObject,
                trendIndication           trendIndication,
                thresholdInfo             thresholdInfo,
                notificationIdentifier    notificationIdentifier,
                correlatedNotifications   correlatedNotifications,
                stateChangeDefinition     stateChangeDefinition,
                monitoredAttributes       monitoredAttributes,
                proposedRepairActions     proposedRepairActions,
                additionalText            additionalText,
                additionalInformation     additionalInformation;

REGISTERED AS  {smi2Notification 3};

environmentalAlarmBehaviour
BEHAVIOUR
        DEFINED AS "This notification type is used to report a problem in the environment.";
```

## 13.4 Equipment alarm

The semantics of the **equipmentAlarm** notification types are specified in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
equipmentAlarm      NOTIFICATION
        BEHAVIOUR       equipmentAlarmBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.AlarmInfo
                AND ATTRIBUTE IDS
                probableCause             probableCause,
                specificProblems          specificProblems,
                perceivedSeverity         perceivedSeverity,
                backedUpStatus            backedUpStatus,
                backUpObject              backUpObject,
                trendIndication           trendIndication,
                thresholdInfo             thresholdInfo,
                notificationIdentifier    notificationIdentifier,
                correlatedNotifications   correlatedNotifications,
                stateChangeDefinition     stateChangeDefinition,
                monitoredAttributes       monitoredAttributes,
                proposedRepairActions     proposedRepairActions,
                additionalText            additionalText,
                additionalInformation     additionalInformation;
```

```
REGISTERED AS  {smi2Notification 4};

equipmentAlarmBehaviour
BEHAVIOUR
        DEFINED AS "This notification type is used to report a failure in the equipment.";
```

## 13.5 Integrity violation

The semantics of the **integrityViolation** notification type are specified in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
integrityViolation     NOTIFICATION
        BEHAVIOUR       integrityViolationBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.SecurityAlarmInfo
                AND ATTRIBUTE IDS
                securityAlarmCause      securityAlarmCause,
                securityAlarmSeverity   securityAlarmSeverity,
                securityAlarmDetector   securityAlarmDetector,
                serviceUser             serviceUser,
                serviceProvider         serviceProvider,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  {smi2Notification 5};

integrityViolationBehaviour  BEHAVIOUR
        DEFINED AS      "This notification is used to report that a potential interruption in information flow has
        occurred such that information may have been illegally modified, inserted or deleted.";
```

## 13.6 Object creation

The semantics of the **objectCreation** notification type are specified in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
objectCreation      NOTIFICATION
        BEHAVIOUR       objectCreationBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.ObjectInfo
                AND ATTRIBUTE IDS
                sourceIndicator         sourceIndicator,
                attributeList           attributeList,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  {smi2Notification 6};

objectCreationBehaviour
BEHAVIOUR
        DEFINED AS "This notification type is used to report the creation of a managed object to another open
        system.";
```

## 13.7 Object deletion

The semantics of the **objectDeletion** notification type are specified in CCITT Rec. X.730 | ISO/IEC 10164-1.

```
objectDeletion      NOTIFICATION
        BEHAVIOUR       objectDeletionBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.ObjectInfo
                AND ATTRIBUTE IDS
                sourceIndicator         sourceIndicator,
                attributeList           attributeList,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
```

```
                    additionalText          additionalText,
                    additionalInformation   additionalInformation;

REGISTERED AS  {smi2Notification 7};

objectDeletionBehaviour
BEHAVIOUR
        DEFINED AS "This notification type is used to report the deletion of a managed object to another open
        system.";
```

## 13.8 Operational violation

The semantics of the **operationalViolation** notification type are specified in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
operationalViolation        NOTIFICATION
        BEHAVIOUR           operationalViolationBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.SecurityAlarmInfo
                AND ATTRIBUTE IDS
                securityAlarmCause      securityAlarmCause,
                securityAlarmSeverity   securityAlarmSeverity,
                securityAlarmDetector   securityAlarmDetector ,
                serviceUser             serviceUser,
                serviceProvider         serviceProvider,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  {smi2Notification 8};
operationalViolationBehaviour       BEHAVIOUR
        DEFINED AS      "This notification is used to report that the provision of the requested service was not
        possible due to the unavailability, malfunction or incorrect invocation of the service.";
```

## 13.9 Physical violation

The semantics of the **physicalViolation** notification type are specified in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
physicalViolation   NOTIFICATION
        BEHAVIOUR       physicalViolationBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.SecurityAlarmInfo
                AND ATTRIBUTE IDS
                securityAlarmCause      securityAlarmCause,
                securityAlarmSeverity   securityAlarmSeverity,
                securityAlarmDetector   securityAlarmDetector ,
                serviceUser             serviceUser,
                serviceProvider         serviceProvider,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  { smi2Notification 9};

physicalViolationBehaviour  BEHAVIOUR
        DEFINED AS      "This notification is used to report that a physical resource has been violated in a way that
        indicates a potential security attack.";
```

## 13.10 Processing error alarm

The semantics of the **processingErrorAlarm** notification type are specified in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
processingErrorAlarm        NOTIFICATION
        BEHAVIOUR           processingErrorAlarmBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.AlarmInfo
                AND ATTRIBUTE IDS
```

```
                probableCause           probableCause,
                specificProblems        specificProblems,
                perceivedSeverity       perceivedSeverity,
                backedUpStatus          backedUpStatus,
                backUpObject            backUpObject,
                trendIndication         trendIndication,
                thresholdInfo           thresholdInfo,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                stateChangeDefinition   stateChangeDefinition,
                monitoredAttributes     monitoredAttributes,
                proposedRepairActions   proposedRepairActions,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  {smi2Notification 10};

processingErrorAlarmBehaviour     BEHAVIOUR
        DEFINED AS      "This notification type is used to report processing failure in a managed object.";
```

## 13.11 Quality of service alarm

The semantics of the **qualityofServiceAlarm** notification type are specified in CCITT Rec. X.733 | ISO/IEC 10164-4.

```
qualityofServiceAlarm       NOTIFICATION
        BEHAVIOUR       qualityofServiceAlarmBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.AlarmInfo
                AND ATTRIBUTE IDS
                probableCause           probableCause,
                specificProblems        specificProblems,
                perceivedSeverity       perceivedSeverity,
                backedUpStatus          backedUpStatus,
                backUpObject            backUpObject,
                trendIndication         trendIndication,
                thresholdInfo           thresholdInfo,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                stateChangeDefinition   stateChangeDefinition,
                monitoredAttributes     monitoredAttributes,
                proposedRepairActions   proposedRepairActions,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  {smi2Notification 11};

qualityofServiceAlarmBehaviour     BEHAVIOUR
        DEFINED AS      "This notification type is used to report a failure in the quality of service of the managed
        object.";
```

## 13.12 Relationship change

The semantics of the **relationshipChange** notification type are specified in CCITT Rec. X.732 | ISO/IEC 10164-3.

```
relationshipChange NOTIFICATION
        BEHAVIOUR       relationshipChangeBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.RelationshipChangeInfo
                AND ATTRIBUTE IDS
                sourceIndicator         sourceIndicator,
                attributeIdentifierList attributeIdentifierList,
                relationshipChangeDefinition    relationshipChangeDefinition,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                additionalText          additionalText,
                additionalInformation   additionalInformation;
```

```
REGISTERED AS  { smi2Notification 12};

relationshipChangeBehaviour        BEHAVIOUR
        DEFINED AS      "This notification type is used to report the change in the value of one or more relationship
        attributes of a managed object, that result through either internal operation of the managed object or via
        management operation.";
```

### 13.13 Security service or mechanism violation

The semantics of the **securityServiceOrMechanismViolation** notification type are specified in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
securityServiceOrMechanismViolation        NOTIFICATION
        BEHAVIOUR       securityServiceOrMechanismViolationBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.SecurityAlarmInfo
                AND ATTRIBUTE IDS
                securityAlarmCause      securityAlarmCause,
                securityAlarmSeverity   securityAlarmSeverity,
                securityAlarmDetector   securityAlarmDetector,
                serviceUser             serviceUser,
                serviceProvider         serviceProvider,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  { smi2Notification 13};

securityServiceOrMechanismViolationBehaviour       BEHAVIOUR
        DEFINED AS      "This notification is used to report that a security attack has been detected by a security
        service or mechanism.";
```

### 13.14 State change

The semantics of the **stateChange** notification type are specified in CCITT Rec. X.731 | ISO/IEC 10164-2.

```
stateChange        NOTIFICATION
        BEHAVIOUR       stateChangeBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.StateChangeInfo
                AND ATTRIBUTE IDS
                sourceIndicator         sourceIndicator,
                attributeIdentifierList attributeIdentifierList,
                stateChangeDefinition   stateChangeDefinition,
                notificationIdentifier  notificationIdentifier,
                correlatedNotifications correlatedNotifications,
                additionalText          additionalText,
                additionalInformation   additionalInformation;

REGISTERED AS  { smi2Notification 14};

stateChangeBehaviour       BEHAVIOUR
        DEFINED AS      "This notification type is used to report the change in the the value of one or more state
        attributes of a managed object, that result through either internal operation of the managed object or via
        management operation.";
```

### 13.15 Time domain violation

The semantics of the **timeDomainViolation** notification type are specified in CCITT Rec. X.736 | ISO/IEC 10164-7.

```
timeDomainViolation        NOTIFICATION
        BEHAVIOUR       timeDomainViolationBehaviour;
        WITH INFORMATION SYNTAX Notification-ASN1Module.SecurityAlarmInfo
```

```
        AND ATTRIBUTE IDS
                securityAlarmCause          securityAlarmCause,
                securityAlarmSeverity       securityAlarmSeverity,
                securityAlarmDetector       securityAlarmDetector ,
                serviceUser                 serviceUser,
                serviceProvider             serviceProvider,
                notificationIdentifier      notificationIdentifier,
                correlatedNotifications     correlatedNotifications,
                additionalText              additionalText,
                additionalInformation       additionalInformation;

REGISTERED AS  { smi2Notification 15};

timeDomainViolationBehaviour       BEHAVIOUR
        DEFINED AS      "This notification is used to report that an event has occurred at an unexpected or prohibited
        time.";
```

# 14 Supporting productions

## 14.1 Managed object class

This clause specifies the required ASN.1 value notation for the value reference used in the MANAGED OBJECT CLASS template.

```
ManagedObjectClassesDefinitions {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 0}

DEFINITIONS ::=
BEGIN

--EXPORTS everything

smi2MObjectClass OBJECT IDENTIFIER ::= {joint-iso-ccitt ms(9) smi(3) part2(2) managedObjectClass(3)}
smi2NameBinding OBJECT IDENTIFIER ::= {joint-iso-ccitt ms(9) smi(3) part2(2) nameBinding(6)}
smi2Package OBJECT IDENTIFIER ::= {joint-iso-ccitt ms(9) smi(3) part2(2) package(4)}

END
```

## 14.2 Attribute types

This clause specifies the ASN.1 syntax for the supporting productions identified in clauses 9 and 10.

```
Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1}
DEFINITIONS IMPLICIT TAGS::=
BEGIN
--EXPORTS everything
IMPORTS
        CMISFilter,Attribute,BaseManagedObjectId,AttributeId,ObjectInstance,ObjectClass, EventTypeId FROM
CMIP-1 {joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3)}
        DistinguishedName FROM InformationFramework {joint-iso-ccitt ds(5) modules(1)
informationFramework(1) }
        AE-title FROM ACSE-1 {joint-iso-ccitt association-control(2) abstract-syntax(1) apdus(0) version(1)};
-- Note that the syntax of AE-title to be used is from CCITT Rec. X.227 | ISO 8650 corrigendum and
-- not "ANY".
smi2AttributeID OBJECT IDENTIFIER ::= { joint-iso-ccitt ms (9) smi(3) part2(2) attribute(7)}
smi2AttributeGroup OBJECT IDENTIFIER ::= { joint-iso-ccitt ms (9) smi(3) part2(2) attributeGroup(8)}

-- The value assignments for the probable cause used in CCITT Rec. X.733 | ISO/IEC 10164-4 are specified
-- below.
-- These are specified in accordance with the corrigendum to ASN.1 ISO/IEC JTC1/SC21 N5901
arfProbableCause OBJECT IDENTIFIER ::= { joint-iso-ccitt ms(9) smi(3) part2(2) standardSpecificExtension(0)
arf(0)}

adapterError ProbableCause ::= globalValue : { arfProbableCause 1 }
```

```
applicationSubsystemFailure ProbableCause ::= globalValue : { arfProbableCause 2 }
bandwidthReduced ProbableCause ::= globalValue : { arfProbableCause 3 }
callEstablishmentError ProbableCause ::= globalValue : { arfProbableCause 4 }
communicationsProtocolError ProbableCause ::= globalValue : { arfProbableCause 5 }
communicationsSubsystemFailure ProbableCause ::= globalValue : { arfProbableCause 6 }
configurationOrCustomizationError ProbableCause ::= globalValue : { arfProbableCause 7 }
congestion ProbableCause ::= globalValue : { arfProbableCause 8 }
corruptData ProbableCause ::= globalValue : { arfProbableCause 9 }
cpuCyclesLimitExceeded ProbableCause ::= globalValue: { arfProbableCause 10 }
dataSetOrModemError ProbableCause ::= globalValue : { arfProbableCause 11 }
degradedSignal ProbableCause ::= globalValue : { arfProbableCause 12 }
dTE-DCEInterfaceError ProbableCause ::= globalValue : { arfProbableCause 13 }
enclosureDoorOpen ProbableCause ::= globalValue : { arfProbableCause 14 }
equipmentMalfunction ProbableCause ::= globalValue : { arfProbableCause 15 }
excessiveVibration ProbableCause ::= globalValue : { arfProbableCause 16 }
fileError ProbableCause ::= globalValue : { arfProbableCause 17}
fireDetected ProbableCause ::= globalValue : { arfProbableCause 18 }
floodDetected ProbableCause ::= globalValue : { arfProbableCause 19 }
framingError ProbableCause ::= globalValue : { arfProbableCause 20 }
heatingOrVentilationOrCoolingSystemProblem ProbableCause ::= globalValue : { arfProbableCause 21 }
humidityUnacceptable ProbableCause ::= globalValue : { arfProbableCause 22}
inputOutputDeviceError ProbableCause ::= globalValue : { arfProbableCause 23}
inputDeviceError ProbableCause ::= globalValue : { arfProbableCause 24}
lANError ProbableCause ::= globalValue : { arfProbableCause 25 }
leakDetected ProbableCause ::= globalValue : { arfProbableCause 26 }
localNodeTransmissionError ProbableCause ::= globalValue : { arfProbableCause 27 }
lossOfFrame ProbableCause ::= globalValue : { arfProbableCause 28}
lossOfSignal ProbableCause ::= globalValue : { arfProbableCause 29 }
materialSupplyExhausted ProbableCause ::= globalValue : { arfProbableCause 30 }
multiplexerProblem ProbableCause ::= globalValue : { arfProbableCause 31 }
outOfMemory ProbableCause ::= globalValue : { arfProbableCause 32 }
ouputDeviceError ProbableCause ::= globalValue : { arfProbableCause 33 }
performanceDegraded ProbableCause ::= globalValue : { arfProbableCause 34 }
powerProblem ProbableCause ::= globalValue : { arfProbableCause 35 }
pressureUnacceptable ProbableCause ::= globalValue : { arfProbableCause 36 }
processorProblem ProbableCause ::= globalValue : { arfProbableCause 37 }
pumpFailure ProbableCause::= globalValue : { arfProbableCause 38 }
queueSizeExceeded ProbableCause ::= globalValue : { arfProbableCause 39 }
receiveFailure ProbableCause ::= globalValue : { arfProbableCause 40 }
receiverFailure ProbableCause ::= globalValue : { arfProbableCause 41 }
remoteNodeTransmissionError ProbableCause ::= globalValue : { arfProbableCause 42 }
resourceAtOrNearingCapacity ProbableCause ::= globalValue : { arfProbableCause 43 }
responseTimeExcessive ProbableCause ::= globalValue : { arfProbableCause 44 }
retransmissionRateExcessive ProbableCause ::= globalValue : { arfProbableCause 45 }
softwareError ProbableCause ::= globalValue : { arfProbableCause 46 }
softwareProgramAbnormallyTerminated ProbableCause ::= globalValue : { arfProbableCause 47 }
softwareProgramError ProbableCause ::= globalValue : { arfProbableCause 48 }
storageCapacityProblem ProbableCause ::= globalValue : { arfProbableCause 49 }
temperatureUnacceptable ProbableCause ::= globalValue : { arfProbableCause 50 }
thresholdCrossed ProbableCause ::= globalValue : { arfProbableCause 51 }
timingProblem ProbableCause ::= globalValue : { arfProbableCause 52 }
toxicLeakDetected ProbableCause ::= globalValue : { arfProbableCause 53 }
transmitFailure ProbableCause ::= globalValue : { arfProbableCause 54 }
transmitterFailure ProbableCause ::= globalValue : { arfProbableCause 55 }
underlyingResourceUnavailable ProbableCause ::= globalValue : { arfProbableCause 56 }
versionMismatch ProbableCause ::= globalValue : { arfProbableCause 57 }

-- The following applies to CCITT applications only.
-- The use of the following values is subject to 8.1.2.12 of The Alarm Management Function
-- (CCITT Recommendation X.733 | 10164-4).

arfProposedRepairAction OBJECT IDENTIFIER ::= { joint-iso-ccitt ms(9) smi(3) part2(2)
standardSpecificExtension(0) arfpra(2)}
noActionRequired OBJECT IDENTIFIER ::= {arfProposedRepairAction 1}
repairActionRequired OBJECT IDENTIFIER ::= {arfProposedRepairAction 2}
```

```
-- The value assignments for the security alarm cause used in CCITT Rec. X.736 | ISO/IEC 10164-7 are --
specified below.

securityAlarmCause OBJECT IDENTIFIER ::= { joint-iso-ccitt ms(9) smi(3) part2(2) standardSpecificExtension(0)
sarf(1) }
authenticationFailure SecurityAlarmCause ::= {securityAlarmCause 1}
breachOfConfidentiality SecurityAlarmCause ::= {securityAlarmCause 2}
cableTamper SecurityAlarmCause ::= {securityAlarmCause 3}
delayedInformation SecurityAlarmCause ::= {securityAlarmCause 4}
denialOfService SecurityAlarmCause ::= {securityAlarmCause 5}
duplicateInformation SecurityAlarmCause ::= {securityAlarmCause 6}
informationMissing SecurityAlarmCause ::= {securityAlarmCause 7}
informationModificationDetected SecurityAlarmCause ::= {securityAlarmCause 8}
informationOutOfSequence SecurityAlarmCause ::= {securityAlarmCause 9}
intrusionDetection SecurityAlarmCause ::= {securityAlarmCause 10}
keyExpired SecurityAlarmCause ::= {securityAlarmCause 11}
nonRepudiationFailure SecurityAlarmCause ::= {securityAlarmCause 12}
outOfHoursActivity SecurityAlarmCause ::= {securityAlarmCause 13}
outOfService SecurityAlarmCause ::= {securityAlarmCause 14}
proceduralError SecurityAlarmCause ::= {securityAlarmCause 15}
unauthorizedAccessAttempt SecurityAlarmCause ::= {securityAlarmCause 16}
unexpectedInformation SecurityAlarmCause ::= {securityAlarmCause 17}
unspecifiedReason SecurityAlarmCause ::= {securityAlarmCause 18}

-- default value definitions

defaultIntervalsOfDay    IntervalsOfDay ::= {{    intervalStart    {hour 0, minute 0},
                                                  intervalEnd      {hour 23, minute 59}}}
defaultStopTime          StopTime ::= continual : NULL
defaultWeekMask          WeekMask ::= {{ daysOfWeek     '1111111'B,
                                         intervalsOfDay  defaultIntervalsOfDay}}
defaultDiscriminatorConstruct    DiscriminatorConstruct ::= and : {}

-- supporting productions

ActiveDestination ::= Destination (WITH COMPONENTS {single PRESENT, multiple ABSENT})
AdditionalText ::= GraphicString
AdditionalInformation ::= SET OF ManagementExtension
Allomorphs ::= SET OF ObjectClass
AdministrativeState ::= ENUMERATED {locked(0),unlocked(1),shuttingDown(2) }
AttributeIdentifierList ::= SET OF AttributeId
AttributeList ::= SET OF Attribute
AttributeValueChangeDefinition::= SET OF SEQUENCE {
                                  attributeID        AttributeId,
                                  oldAttributeValue          [1] ANY DEFINED BY attributeID OPTIONAL,
                                  newAttributeValue          [2] ANY DEFINED BY attributeID}
AlarmStatus ::= SET OF INTEGER { underRepair(0), critical(1), major(2), minor(3), alarmOutstanding(4)}
AvailabilityStatus ::= SET OF INTEGER { inTest(0), failed(1), powerOff(2), offLine(3), offDuty(4),
        dependency(5), degraded(6), notInstalled (7) , logFull(8)}
-- logFull is defined in CCITT Rec X.735 | ISO/IEC 10164-6.
BackedUpStatus ::= BOOLEAN
-- True implies backed up
-- False implies not backed up
BackUpDestinationList ::= SEQUENCE OF AE-title
BackUpRelationshipObject ::= CHOICE { objectName ObjectInstance, noObject NULL}
CapacityAlarmThreshold ::= SET OF INTEGER (0..100)
ConfirmedMode ::= BOOLEAN
-- True implies the mode for event report is confirmed
ControlStatus ::= SET OF INTEGER { subjectToTest(0), partOfServicesLocked(1), reservedForTest(2),
suspended(3)}
Count ::= INTEGER
CounterThreshold ::= SET OF SEQUENCE {
                                  comparisonLevel          INTEGER,
                                  offsetValue              INTEGER,
                                  notificationOnOff        BOOLEAN}
CorrelatedNotifications ::= SET OF SEQUENCE {
```

```
                        correlatedNotifications   SET OF NotificationIdentifier,
                        sourceObjectInst                 ObjectInstance OPTIONAL }
CurrentLogSize ::= INTEGER

Destination ::= CHOICE {
                    single          AE-title,
                    multipleSET OF AE-title}
-- Note that the syntax of AE-title to be used is from CCITT X.227 | ISO 8650 Technical
-- corrigendum 1 and not "ANY".
DiscriminatorConstruct ::= CMISFilter
EventTime ::= GeneralizedTime
GaugeThreshold ::= SET OF SEQUENCE {
                    notifyLow       NotifyThreshold,
                    notifyHigh      NotifyThreshold}
GaugeThresholdValue ::= ObservedValue
-- same choice must be chosen within a sequence
-- for both GenericOldState and GenericNewState.
GroupObjects ::= SET OF ObjectInstance

IntervalsOfDay ::= SET OF SEQUENCE{
                        intervalStart   Time24,
                        intervalEnd     Time24}

LogAvailability ::= AvailabilityStatus (WITH COMPONENT ( logFull | offDuty))
LogFullAction ::= ENUMERATED { wrap(0), halt (1)}
LoggingTime ::= GeneralizedTime
LogRecordId ::= SimpleNameType(WITH COMPONENTS {number PRESENT, string ABSENT})
MaxLogSize ::= INTEGER {unlimited(0)} -- size in octets
ManagementExtension ::= SEQUENCE {
                        identifier      OBJECT IDENTIFIER,
                        significance    [1] BOOLEAN DEFAULT FALSE,
                        information     [2] ANY DEFINED BY identifier }
MonitoredAttributes ::= SET OF Attribute
NameBinding ::= OBJECT IDENTIFIER
NotificationIdentifier ::= INTEGER -- reuse only when no requirement exists
                    -- for correlating prior notification.
NotifyThreshold ::= SEQUENCE {
                        threshold       ObservedValue,
                        notifyOnOff     BOOLEAN }
NumberOfRecords ::= INTEGER
ObservedValue ::= CHOICE {
                    integer  INTEGER,
                    real             REAL }
OperationalState::= ENUMERATED { disabled(0), enabled(1) }
Packages ::= SET OF OBJECT IDENTIFIER
PerceivedSeverity ::= ENUMERATED{ indeterminate(0),
-- used when it is not possible to assign the following values
                        critical(1), major(2), minor(3), warning(4), cleared(5) }
PrioritisedObject ::= SET OF SEQUENCE {
                    object          ObjectInstance,
                    priority  INTEGER {lowest(0), highest(127)}}
ProbableCause ::= CHOICE {
                    globalValue     OBJECT IDENTIFIER,
                    localValue      INTEGER}
-- The values for the probable cause defined in CCITT Rec X.733 | ISO/IEC 10164-4 are specified
-- above.
ProceduralStatus ::= SET OF INTEGER { initializationRequired(0), notInitialized(1), initializing(2), reporting (3),
terminating (4) }
ProposedRepairActions ::= SET OF SpecificIdentifier
SchedulingAvailability ::= AvailabilityStatus (WITH COMPONENT (offDuty))
SecurityAlarmCause ::= OBJECT IDENTIFIER
SecurityAlarmSeverity ::= PerceivedSeverity (indeterminate | critical | major | minor | warning)
SecurityAlarmDetector::= CHOICE {   mechanism [0]   OBJECT IDENTIFIER,
                                    object [1]      ObjectInstance,
                                    application     [2] AE-title}
ServiceProvider ::= ServiceUser
```

```
ServiceUser ::= SEQUENCE {
                        identifier          OBJECT IDENTIFIER,
                        details  ANY DEFINED BY identifier }
SimpleNameType ::= CHOICE {  number          INTEGER,
                             string          GraphicString}
SpecificIdentifier ::= CHOICE {  OBJECT IDENTIFIER,
                                 INTEGER}
SourceIndicator ::= ENUMERATED { resourceOperation(0), managementOperation(1), unknown(2)}
SpecificProblems ::= SET OF SpecificIdentifier
StandbyStatus ::= INTEGER { hotStandby(0), coldStandby(1), providingService(2)}
StartTime ::= GeneralizedTime
StopTime ::= CHOICE {  specific          GeneralizedTime,
                       continual         NULL}
SupportedFeatures ::= SET OF SEQUENCE {
                        featureIdentifier          OBJECT IDENTIFIER,
                        featureInfo                ANY DEFINED BY featureIdentifier}
SystemId ::= CHOICE {  name          GraphicString,
                       number        INTEGER,
                       nothing       NULL}
SystemTitle ::= CHOICE {        distinguishedName       DistinguishedName,
                                oid                     OBJECT IDENTIFIER,
                                nothing                 NULL}
TideMarkInfo ::= SEQUENCE {     currentTideMark         TideMark,
                                previousTideMark        TideMark,
                                resetTime               GeneralizedTime}
TideMark ::= CHOICE {           maxTideMar      [0] ObservedValue,
                                minTideMark     [1] ObservedValue}
Time24 ::= SEQUENCE {           hour            INTEGER (0..23),
                                minute  INTEGER (0..59) }
ThresholdInfo ::= SEQUENCE {
                        triggeredThreshold      AttributeId,
                        observedValue  ObservedValue,
                        thresholdLevel  [1] ThresholdLevelInd OPTIONAL,
                        -- Indication needed for multilevel thresholds
                        armTime         [2] GeneralizedTime OPTIONAL }

ThresholdLevelInd ::= CHOICE {
                        up      [1] SEQUENCE { high     ObservedValue,
                                               low      ObservedValue OPTIONAL
                        -- up is the only permitted choice for counter --},
                        down    [2] SEQUENCE { high     ObservedValue,
                                               low      ObservedValue}}
TrendIndication ::= ENUMERATED { lessSevere(0), noChange(1), moreSevere(2) }
UnknownStatus ::= BOOLEAN -- TRUE implies unknown status
UnscheduledLogAvailability ::= AvailabilityStatus (WITH COMPONENT (logFull))
UsageState::= ENUMERATED { idle(0), active(1), busy(2) }
WeekMask ::= SET OF SEQUENCE {
                        daysOfWeek      BIT STRING {sunday(0),monday(1),tuesday(2),wednesday(3),
                                        thursday(4), friday(5), saturday(6)} (SIZE(7) ),
                        intervalsOfDay          IntervalsOfDay }

END     -- end of supporting productions
```

## 14.3 Notification types

This clause specifies the ASN.1 syntax for the supporting productions identified in clause 13.

```
Notification-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 2}
DEFINITIONS IMPLICIT TAGS::=
BEGIN
IMPORTS
ProbableCause, SpecificProblems, PerceivedSeverity, BackedUpStatus, TrendIndication, ThresholdInfo,
NotificationIdentifier, CorrelatedNotifications, MonitoredAttributes, ProposedRepairActions,
AdditionalText, AdditionalInformation, SecurityAlarmCause, SecurityAlarmSeverity,
```

```
SecurityAlarmDetector, AttributeValueChangeDefinition, SourceIndicator, AttributeIdentifierList,
AttributeList, ServiceUser, ServiceProvider
FROM Attribute-ASN1Module
        ObjectClass,AttributeId,ObjectInstance FROM CMIP-1 {joint-iso-ccitt ms(9) cmip(1) version1 (1)
protocol(3)} ;

--EXPORTS everything
smi2Notification OBJECT IDENTIFIER ::= { joint-iso-ccitt ms (9) smi(3) part2(2) notification(10)}

AlarmInfo ::= SEQUENCE {
        probableCause                   ProbableCause,
        specificProblems                [1]SpecificProblems OPTIONAL,
        perceivedSeverity               PerceivedSeverity,
        backedUpStatus                  BackedUpStatus OPTIONAL,
        backUpObject                    [2]ObjectInstance OPTIONAL,
        trendIndication                 [3]TrendIndication OPTIONAL,
        thresholdInfo                   [4]ThresholdInfo OPTIONAL,
        notificationIdentifier          [5]NotificationIdentifier OPTIONAL,
        correlatedNotifications         [6]CorrelatedNotifications OPTIONAL,
        stateChangeDefinition           [7]AttributeValueChangeDefinition OPTIONAL,
        monitoredAttributes             [8]MonitoredAttributes OPTIONAL,
        proposedRepairActions           [9]ProposedRepairActions OPTIONAL,
        additionalText                  AdditionalText OPTIONAL,
        additionalInformation           [10]AdditionalInformation OPTIONAL}

AttributeValueChangeInfo ::= SEQUENCE {
        sourceIndicator                 SourceIndicator OPTIONAL,
        attributeIdentifierList         [1]AttributeIdentifierList OPTIONAL,
        attributeValueChangeDefinition  AttributeValueChangeDefinition,
        notificationIdentifier          NotificationIdentifier OPTIONAL,
        correlatedNotifications         [2]CorrelatedNotifications OPTIONAL,
        additionalText                  AdditionalText OPTIONAL,
        additionalInformation           [3]AdditionalInformation OPTIONAL}
ObjectInfo ::= SEQUENCE {
        sourceIndicator                 SourceIndicator OPTIONAL,
        attributeList                   AttributeList OPTIONAL,
        notificationIdentifier          NotificationIdentifier OPTIONAL,
        correlatedNotifications         [1]CorrelatedNotifications OPTIONAL,
        additionalText                  AdditionalText OPTIONAL,
        additionalInformation           [2]AdditionalInformation OPTIONAL}

RelationshipChangeInfo ::= SEQUENCE {
        sourceIndicator                 SourceIndicator OPTIONAL,
        attributeIdentifierList         [1]AttributeIdentifierList OPTIONAL,
        relationshipChangeDefinition    AttributeValueChangeDefinition,
        notificationIdentifier          NotificationIdentifier OPTIONAL,
        correlatedNotifications         [2]CorrelatedNotifications OPTIONAL,
        additionalText                  AdditionalText OPTIONAL,
        additionalInformation           [3]AdditionalInformation OPTIONAL}

StateChangeInfo ::= SEQUENCE {
        sourceIndicator                 SourceIndicator OPTIONAL,
        attributeIdentifierList         [1]AttributeIdentifierList OPTIONAL,
        stateChangeDefinition           AttributeValueChangeDefinition,
        notificationIdentifier          NotificationIdentifier OPTIONAL,
        correlatedNotifications         [2]CorrelatedNotifications OPTIONAL,
        additionalText                  AdditionalText OPTIONAL,
        additionalInformation           [3]AdditionalInformation OPTIONAL}

SecurityAlarmInfo ::= SEQUENCE {
        securityAlarmCause              SecurityAlarmCause,
        securityAlarmSeverity           SecurityAlarmSeverity,
        securityAlarmDetector           SecurityAlarmDetector,
        serviceUser                     ServiceUser,
        serviceProvider                 ServiceProvider,
        notificationIdentifier          NotificationIdentifier OPTIONAL,
```

```
        correlatedNotifications     [1]CorrelatedNotifications OPTIONAL,
        additionalText              AdditionalText OPTIONAL,
        additionalInformation       [2]AdditionalInformation OPTIONAL}

END
```

### 14.4 Parameter types

This clause specifies the ASN.1 syntax for the supporting productions identified in clause 12.

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

DEFINITIONS IMPLICIT TAGS::=
BEGIN
smi2Parameter OBJECT IDENTIFIER ::= {joint-iso-ccitt ms(9) smi(3) part2(2) parameter(5)}
MiscellaneousError ::= NULL

END
```

## 15 Conformance and compliance

### 15.1 Conformance

This Recommendation | International Standard does not specify any conformance requirements placed on open systems.

### 15.2 Compliance

Where other CCITT Recommendations | International Standards refine any of the managed object class definitions contained in this Recommendation | International Standard via the refinement and referencing mechanisms defined in CCITT Rec. X.722 | ISO/IEC 10165-4, the managed object classes that refine these definitions shall comply with the behavioural and syntactic aspects of the superclass as defined in this Recommendation | International Standard.

Where other CCITT Recommendations | International Standards import any of the definitions contained in this Recommendation | International Standard into attribute type or managed object class or notification type definitions via the referencing mechanism defined in CCITT Rec. X.722 | ISO/IEC 10165-4, the attribute type or managed object class or notification type definitions that reference these definitions shall comply with the behavioural and syntactic aspects of those types as specified in this Recommendation | International Standard.

# Annex A
# Counters and counter threshold attribute

(This Annex forms an integral part of this Recommendation | International Standard )

## A.1 Counter

This Annex defines a number of counter types which are useful for importing in other standards.

### A.1.1 Corrupted PDUs received counter

The attribute type **corruptedPDUsReceivedCounter** specifies the total number of corrupted PDUs received.

```
corruptedPDUsReceivedCounter    ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 72};
```

### A.1.2 Incoming connection reject error counter

The attribute type **IncomingConnectionRejectErrorCounter** specifies the total number of incoming connection requests which were received by the managed object but rejected due to protocol errors.

```
incomingConnectionRejectErrorCounter      ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 73};
```

### A.1.3 Incoming connection requests counter

The attribute type **incomingConnectionRequestsCounter** specifies the total number of incoming connection requests.

```
incomingConnectionRequestsCounter         ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 74};
```

### A.1.4 Incoming disconnect counter

The attribute type **incomingDisconnectCounter** specifies the total number of incoming disconnect requests received by the managed object.

```
incomingDisconnectCounter         ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 75};
```

### A.1.5 Incoming disconnect error counter

The attribute type **incomingDisconnectErrorCounter** specifies the total number of incoming disconnect requests received by the managed object due to protocol errors.

```
incomingDisconnectErrorCounter    ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 76};
```

### A.1.6 Incoming protocol error counter

The attribute type **incomingProtocolErrorCounter** specifies the total number of error report or reset PDUs which were received by the managed object due to protocol errors.

```
incomingProtocolErrorCounter      ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 77};
```

### A.1.7 Octets received counter

The attribute type **octetsReceivedCounter** specifies the total number of user data octets received by the managed object.

```
octetsReceivedCounter      ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 78};
```

### A.1.8 Octets retransmitted error counter

The attribute type **octetsRetransmittedErrorCounter** specifies the total number of octets retransmitted by the managed object.

```
octetsRetransmittedErrorCounter    ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 79};
```

### A.1.9 Octets sent counter

The attribute type **octetsSentCounter** specifies the total number of user data octets sent by the managed object.

```
octetsSentCounter ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 80};
```

### A.1.10 Outgoing connection reject error counter

The attribute type **outgoingConnectionRejectErrorCounter** specifies the total number of outgoing connection requests which were sent by the managed object but rejected due to protocol errors.

```
outgoingConnectionRejectErrorCounter      ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 81};
```

### A.1.11 Outgoing connection requests counter

The attribute type **outgoingConnectionRequestsCounter** specifies the total number of outgoing connection requests.

```
outgoingConnectionRequestsCounter         ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 82};
```

### A.1.12 Outgoing disconnect counter

The attribute type **outgoingDisconnectCounter** specifies the total number of outgoing disconnect requests processed by the managed object.

```
outgoingDisconnectCounter ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 83};
```

### A.1.13 Outgoing disconnect error counter

The attribute type **outgoingDisconnectErrorCounter** specifies the total number of outgoing disconnect requests sent by the managed object due to protocol errors.

```
outgoingDisconnectErrorCounter    ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 84};
```

### A.1.14 Outgoing protocol error counter

The attribute type **outgoingProtocolErrorCounter** specifies the total number of error report or reset PDUs which were sent by the managed object due to protocol errors.

```
outgoingProtocolErrorCounter       ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 85};
```

### A.1.15 PDUs received counter

The attribute type **pdusReceivedCounter** specifies the total number of PDUs received by the managed object.

```
pdusReceivedCounter       ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 86};
```

### A.1.16 PDUs retransmitted error counter

The attribute type **pdusRetransmittedErrorCounter** specifies the total number of PDUs retransmitted by the managed object.

```
pdusRetransmittedErrorCounter    ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 87};
```

### A.1.17 PDUs sent counter

The attribute type specifies **pdusSentCounter** the total number of PDUs sent by the managed object.

```
pdusSentCounter   ATTRIBUTE
        DERIVED FROM  counter;

REGISTERED AS  { smi2AttributeID 88};
```

## A.2 Counter-threshold

### A.2.1 Corrupted PDUs received threshold

The attribute type **corruptedPDUsReceivedThreshold** specifies a counter-threshold which is associated with the PDUs Corrupted PDUs Received counter attribute type.

```
corruptedPDUsReceivedThreshold  ATTRIBUTE
        DERIVED FROM  counter-Threshold;
```

REGISTERED AS { smi2AttributeID 89};

### A.2.2 Incoming connection reject error threshold

The attribute type **incomingConnectionRejectErrorThreshold** specifies a counter-threshold which is associated with the Incoming ConnectionRejectError counter attribute type.

```
incomingConnectionRejectErrorThreshold    ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 90};
```

### A.2.3 Incoming connection requests threshold

The attribute type **incomingConnectionRequestsThreshold** specifies a counter-threshold which is associated with the Incoming Connection Requests counter attribute type.

```
incomingConnectionRequestsThreshold
        ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 91};
```

### A.2.4 Incoming disconnect error threshold

The attribute type **incomingDisconnectErrorThreshold** specifies a counter-threshold which is associated with the Incoming Disconnect Error counter attribute type.

```
incomingDisconnectErrorThreshold
        ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 92};
```

### A.2.5 Incoming protocol error threshold

The attribute type **incomingProtocolErrorThreshold** specifies a counter-threshold which is associated with the Incoming Protocol Error counter attribute type.

```
incomingProtocolErrorThreshold    ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 93};
```

### A.2.6 Octets received threshold

The attribute type **octetsReceivedThreshold** specifies a counter-threshold which is associated with the Octets Received counter attribute type.

```
octetsReceivedThreshold    ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 94};
```

### A.2.7 Octets retransmitted threshold

The attribute type **octetsRetransmittedThreshold** specifies a counter threshold which can be associated with the octects retransmitted counter attribute type.

```
octetsRetransmittedThreshold
        ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 95};
```

### A.2.8 Octets sent threshold

The attribute type **octetsSentThreshold** specifies a counter-threshold which is associated with the Octets Sent counter attribute type.

```
octetsSentThreshold          ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 96};
```

### A.2.9 Outgoing connection reject error threshold

The attribute type **outgoingConnectionRejectErrorThreshold** specifies a counter-threshold which is associated with the Outgoing Connection Reject Error counter attribute type.

```
outgoingConnectionRejectErrorThreshold
        ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 97};
```

### A.2.10 Outgoing connection requests threshold

The attribute type **outgoingConnectionRequestsThreshold** specifies a counter-threshold which is associated with the Outgoing Connection Requests counter attribute type.

```
outgoingConnectionRequestsThreshold
        ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 98};
```

### A.2.11 Outgoing disconnect error threshold

The attribute type **outgoingDisconnectErrorThreshold** specifies a counter-threshold which is associated with the Outgoing Disconnect Error counter attribute type.

```
outgoingDisconnectErrorThreshold
        ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 99};
```

### A.2.12 Outgoing protocol error threshold

The attribute type **outgoingProtocolErrorThreshold** specifies a counter-threshold which is associated with the Outgoing Protocol Error counter attribute type.

```
outgoingProtocolErrorThreshold
        ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 100};
```

### A.2.13 PDUs received threshold

The attribute type **pdusReceivedThreshold** specifies a counter-threshold which is associated with the PDUs Received counter attribute type.

```
pdusReceivedThreshold      ATTRIBUTE
        DERIVED FROM  counter-Threshold;

REGISTERED AS  { smi2AttributeID 101};
```