INTERNATIONAL TELECOMMUNICATION UNION

ITU-T X.746

(02/2000)

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

SERIES X: DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS

OSI management – Management functions and ODMA functions

Information technology – Open Systems Interconnection – Systems management: Scheduling function

ITU-T Recommendation X.746

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEMS INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.369
IP-based networks	X.370–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	X.600–X.629
Efficiency	X.630–X.639
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems Management framework and architecture	X.700–X.709
Management Communication Service and Protocol	X.710–X.719
Structure of Management Information	X.720–X.729
Management functions and ODMA functions	**X.730–X.799**
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, Concurrency and Recovery	X.850–X.859
Transaction processing	X.860–X.879
Remote operations	X.880–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999

For further details, please refer to the list of ITU-T Recommendations.

INTERNATIONAL STANDARD ISO/IEC 10164-15
ITU-T RECOMMENDATION X.746

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – SYSTEMS MANAGEMENT: SCHEDULING FUNCTION

Summary

This Recommendation | International Standard defines the scheduling function. The scheduling function is a systems management function which may be used by an application process in a centralized or decentralized management environment to exchange information and commands for the purpose of systems management.

Source

ITU-T Recommendation X.746 was prepared by ITU-T Study Group 4 (1997-2000) and approved on 4 February 2000. An identical text is also published as ISO/IEC 10164-15.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSC Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

Introduction

ITU-T Rec. X.746 | ISO/IEC 10164-15 is a member of a family of Recommendations | International Standards dealing with *Systems Management*:

- X.730 (1992) | ISO/IEC 10164-1:1993: Object management function.
- X.731 (1992) | ISO/IEC 10164-2:1993: State management function.
- X.732 (1992) | ISO/IEC 10164-3:1993: Attributes for representing relationships.
- X.733 (1992) | ISO/IEC 10164-4:1992: Alarm reporting function.
- X.734 (1992) | ISO/IEC 10164-5:1993: Event report management function.
- X.735 (1992) | ISO/IEC 10164-6:1993: Log control function.
- X.736 (1992) | ISO/IEC 10164-7:1992: Security alarm reporting function.
- X.737 (1995) | ISO/IEC 10164-14:1996: Confidence and diagnostic test categories.
- X.738 (1993) | ISO/IEC 10164-13:1995: Summarization function.
- X.739 (1993) | ISO/IEC 10164-11:1994: Metric objects and attributes.
- X.740 (1992) | ISO/IEC 10164-8:1993: Security audit trail function.
- X.741 (1995) | ISO/IEC 10164-9:1995: Objects and attributes for access control.
- X.742 (1995) | ISO/IEC 10164-10:1995: Usage metering function for accounting purposes.
- X.743 (1998) | ISO/IEC 10164-20:1999: Time management function.
- X.744 (1996) | ISO/IEC 10164-18:1997: Software management function.
- X.745 (1993) | ISO/IEC 10164-12:1994: Test management function.
- X.746 (2000) | ISO/IEC 10164-15:2001: Scheduling function.
- X.748 (1999) | ISO/IEC 10164-22:2000: Response time monitoring function.
- X.749 (1997) | ISO/IEC 10164-19:1998: Management domain and management policy management function.
- X.750 (1996) | ISO/IEC 10164-16:1997: Management knowledge management function.
- X.751 (1995) | ISO/IEC 10164-17:1996: Changeover function.
- X.753 (1997) | ISO/IEC 10164-21:1998: Command sequencer for systems management.

INTERNATIONAL STANDARD
ITU-T RECOMMENDATION

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – SYSTEMS MANAGEMENT: SCHEDULING FUNCTION

1 Scope

This Recommendation | International Standard defines the scheduling function. The scheduling function is a systems management function which may be used by an application process in a centralized or decentralized management environment to exchange information and commands for the purpose of systems management, as defined by CCITT Rec. X.700 | ISO/IEC 7498-4. This Recommendation | International Standard is positioned in the application layer of ITU-T Rec. X.200 | ISO/IEC 7498-1 and is defined according to the model provided by ISO/IEC 9545. The role of systems management functions is described by CCITT Rec. X.701 | ISO/IEC 10040.

This Recommendation | International Standard:

- identifies a set of requirements satisfied by the function;
- provides a model for scheduling;
- specifies the management requirements of the function and how these are realized by specification of managed objects and their behaviour;
- defines the conformance requirements to be met by implementations of this Recommendation | International Standard;
- defines managed objects.

This Recommendation | International Standard does not define:

- the manner in which management is to be accomplished by the user of the scheduling function;
- the nature of any implementation intended to provide the scheduling function;
- the nature of any interactions which result in the use of the scheduling function;
- the interactions which result by the simultaneous use of several management functions;
- the occasions where the use of the scheduling function is appropriate;
- the services necessary for the establishment, normal and abnormal release of a management association.

2 Normative references

The following Recommendations and 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 edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.200 (1994 | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model.*
- ITU-T Recommendation X.210 (1993) | ISO/IEC 10731:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: Conventions for the definition of OSI services.*
- ITU-T Recommendation X.680 (1997) | ISO/IEC 8824-1:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*
- ITU-T Recommendation X.681 (1997) | ISO/IEC 8824-2:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification.*

- ITU-T Recommendation X.682 (1997) | ISO/IEC 8824-3:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification.*
- ITU-T Recommendation X.690 (1997) | ISO/IEC 8825-1:1998, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*
- ITU-T Recommendation X.691 (1997) | ISO/IEC 8825-2:1998, *Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER).*
- CCITT Recommendation X.701 (1992) | ISO/IEC 10040:1992, *Information technology – Open Systems Interconnection – Systems management overview.*
- ITU-T Recommendation X.710 (1997) | ISO/IEC 9595:1998, *Information technology – Open Systems Interconnection – Common management information service.*
- CCITT Recommendation X.721 (1992) | ISO/IEC 10165-2:1992, *Information technology – Open Systems Interconnection – Structure of management information: Definition of management information.*
- CCITT Recommendation X.722 (1992) | ISO/IEC 10165-4:1992, *Information technology – Open Systems Interconnection – Structure of management information: Guidelines for the definition of managed objects*.
- ITU-T Recommendation X.724 (1993) | ISO/IEC 10165-6:1994, *Information technology – Open Systems Interconnection – Structure of management information: Requirements and guidelines for implementation conformance statement proformas associated with OSI management*.
- CCITT Recommendation X.730 (1992) | ISO/IEC 10164-1:1993, *Information technology – Open Systems Interconnection – Systems Management: Object management function*.
- CCITT Recommendation X.731 (1992) | ISO/IEC 10164-2:1993, *Information technology – Open Systems Interconnection – Systems Management: State management function*.
- CCITT Recommendation X.734 (1992) | ISO/IEC 10164-5:1993, *Information technology – Open Systems Interconnection – Systems Management: Event report management function*.
- ITU-T Recommendation X.738 (1993) | ISO/IEC 10164-13:1995, *Information technology – Open Systems Interconnection – Systems Management: Summarization function*.
- ITU-T Recommendation X.739 (1993) | ISO/IEC 10164-11:1994, *Information technology – Open Systems Interconnection – Systems Management: Metric objects and attributes*.
- ITU-T Recommendation X.745 (1993) | ISO/IEC 10164-12:1994, *Information technology – Open Systems Interconnection – Systems Management: Test Management Function*.

2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.291 (1992), *OSI conformance testing methodology and framework for protocol Recommendations for CCITT applications – Abstract test suite specification.*

  ISO/IEC 9646-2:1994, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 2: Abstract test suite specification.*
- CCITT Recommendation X.700 (1992), *Management framework for Open Systems Interconnection (OSI) for CCITT applications*.

  ISO/IEC 7498-4:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 4: Management framework.*

2.3 Additional references

- ITU-T Recommendation M.3100 (1995), *Generic network information model*.
- ISO/IEC 9545:1994, *Information technology – Open Systems Interconnection – Application Layer Structure*.

3 Definitions

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

3.1 Basic reference model definitions

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.200 | ISO/IEC 7498-1.

a) open system;

b) systems management.

3.2 Abstract syntax notation one definitions

This Recommendation | International Standard makes use of the following term defined in ITU-T Rec. X.680 | ISO/IEC 8824-1.

a) object identifier.

3.3 Management framework definitions

This Recommendation | International Standard makes use of the following term defined in CCITT Rec. X.700 | ISO/IEC 7498-4.

a) managed object.

3.4 Common management information service definitions

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.710 | ISO/IEC 9595.

a) attribute;

b) common management information service.

3.5 Systems management overview definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.701 | ISO/IEC 10040.

a) agent;

b) managed object class;

c) manager;

d) notification;

e) systems management operations.

3.6 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) action;

b) actual class;

c) behaviour;

d) characteristic;

e) conditional package;

f) inheritance;

g) instantiation;

h) mandatory package;

i) name binding;

j) package;

k) subclass;

l) superclass.

3.7 Additional definitions

3.7.1 aperiodic scheduling: A type of scheduling that controls the triggering of activities at certain specified times within specified managed object instances.

3.7.2 interval scheduling: A type of scheduling that controls a number of intervals of operation of activities within specified managed object instances.

3.7.3 periodic scheduling: A type of scheduling that controls the repetitive triggering of activities within specified managed object instances.

3.7.4 scheduling: The method of controlling the timing of the execution of a scheduled activity within or represented by a managed object.

3.7.5 scheduled managed object (SMO): The managed object whose activities are to be scheduled.

3.7.6 scheduler object (SO): The managed object that defines the type and values of the schedule to be applied to activities within SMOs.

3.7.7 trigger scheduling: A type of scheduling that controls the triggering of activities within specified managed object instances.

4 Abbreviations

For the purposes of this Recommendation | International standard, the following abbreviations apply:

ASN.1	Abstract Syntax Notation One
CMIS	Common Management Information Service
ICS	Implementation Conformance Statement
MAPDU	Management Application Protocol Data Unit
MCS	Management Conformance Statement
MOCS	Managed Object Conformance Statement
NE	Network Element
OC	Object Class
OS	Operation System
SMO	Scheduled Managed Object
SO	Scheduler Object

5 Conventions

The ICS proformas specified in this Recommendation | International Standard (see Annexes C to G) use the common notations, defined in CCITT Rec. X.291 | ISO/IEC 9646-2 and CCITT Rec. X.296 | ISO/IEC 9646-7.

6 Requirements

In terms of functionality, the requirements to be satisfied are:

- Provide a function that can schedule a number of activities within multiple managed objects according to a single schedule.
- Be able to specify the time duration that the schedule is active.
- For schedules that control the interval of operation of an activity within a managed object, the start and stop time should be defined as the actual time within a 24-hour clock.
- Provide a function that can schedule aperiodic and periodic triggering of an activity.
- Allow that the scheduling information communicated to the scheduled object be independent of the action the scheduled object performs. The scheduler may have no knowledge about this action. As a consequence, the relations between the scheduler, the scheduling information and the corresponding actions to be performed are existing in the scheduled object.
- Several independent schedulers can coexist.
- Scheduling shall be possible on base of type of day (e.g. weekend, Christmas, bank holiday).

Interval scheduling

- Provide a function that controls the scheduled activities of one or more managed objects.
- Provide a configurable schedule that repeats over a specified time period. The specified time period may be a day, a week or a month.
- Provide a user defined number of intervals together with the start and stop times of each of these intervals within the specified period.
- Overlapping intervals shall be allowed. Precedence rules are therefore needed in order to decide which interval is active.

Trigger scheduling

- Provide a function that controls the triggering of an activity of one or more managed objects.
- Provide a configurable period for the repetitions of the triggering.
- Provide a user defined list of trigger times.

7 Model

Scheduling can be modelled as a part of the managed object whose operation or activity is to be scheduled, or as a separate managed object.

Characteristics for the control of a schedule can be imported into a managed object class or can be defined as a separate managed object. These two ways of defining scheduling of a managed object are termed internal and external scheduling, respectively. This Recommendation | International Standard describes models for both internal and external scheduling.

This Recommendation | International Standard also describes four types of scheduling in 7.3 below: interval, trigger (periodic and aperiodic), operations and index ("multi-scheduler") scheduling. These scheduling types can be used with internal and external scheduling mechanisms.

The activities which can be controlled by scheduling are defined as part of the scheduled managed object (SMO) class. There need to be characteristics in the SMO related to these scheduled activities.

7.1 Internal scheduling mechanism

It is appropriate to define the scheduling mechanism within a managed object class if it will not need to be altered in the future and the managed object is to be individually scheduled. The scheduling mechanism can be defined within a managed object class by including the appropriate scheduling components (e.g. attributes and behaviour). If more than one type of scheduling is defined within a managed object class, the conditions for instantiation of each type of scheduling must be defined in the managed object class definition.

When the scheduling mechanism is defined within the managed object whose activity is scheduled, no additional objects are required and the scheduling may be manipulated through the use of systems management operations. However, when multiple activities within a managed object are to be scheduled using this mechanism, separate scheduling characteristics are required for each activity.

Scheduling characteristics for each activity may include more than one type of scheduling (see 7.3) and the conditions for instantiation of each type shall be defined in the managed object class definition.

7.2 External scheduling mechanism

It is beneficial to define an external scheduling mechanism so that schedules may be determined independently of SMOs. Many managed objects may be controlled by a single schedule. If a single scheduler object (SO) provides the schedule, there may be no need for scheduling components in the SMOs: in such a case, this eliminates the need to replicate and coordinate schedules across SMOs.

The scheduling function is represented by SOs which are separate from the SMOs, as shown in Figure 1. One SO may control activities in any number of SMOs. Multiple external schedules are allowed for the same activity. The approach for defining more than one type of scheduling for the same activity is described in 7.3.

The scheduler object provides a schedule to a SMO. SMOs shall have attributes which identify the SOs providing schedules. Each of these attributes shall have and be associated with behaviour which describes the effect of the schedule upon the SMO. It may not be necessary to use several SOs to provide this, when using the index SO.

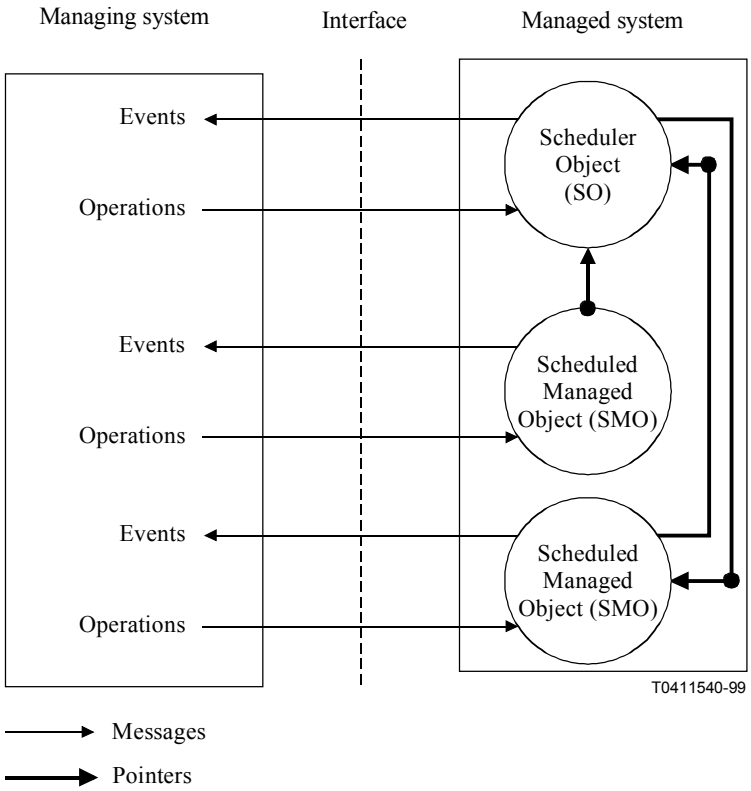


Figure 1 – Scheduler Object model

7.3 Types of scheduling

There are four specific types of scheduling: interval scheduling, trigger scheduling, operations scheduling and index scheduling. This Recommendation | International Standard describes four types of interval scheduling – daily, weekly, monthly and type-of-day interval scheduling (see 7.3.1); two types of trigger scheduling – periodic and aperiodic scheduling (see 7.3.2), operations scheduling (see Operations scheduling) and index scheduling (see Index scheduling). These types of scheduling are defined by packages which may be included in managed objects for the purpose of internal scheduling (except for the operations scheduling) or in a scheduler object for external scheduling.

NOTE – Other scheduling packages are defined in CCITT Rec. X.734 | ISO/IEC 10164-5.

If a combination of interval and trigger scheduling is required for one activity, the triggering is effective only within the intervals defined by the interval schedule.

7.3.1 Interval scheduling

Interval scheduling is used to define a schedule that controls a sequence of transitions of an activity of a SMO between the active and inactive state. The schedule may repeat in one of the following ways: a given number of days with specified intervals for each day, a given number of weeks with specified intervals for specified days of each week or a given number of months with specified intervals for specified days of each month. Each of these types of interval scheduling, daily, weekly and monthly is specified by selecting the intervals of day parameter for the day, week or month mask attribute in the appropriate scheduler object class.

The duration over which interval scheduling affects the operation of the SMOs may be controlled by the specified duration start time and duration stop time (date and time).

The intervals of operation are specified by a set of interval start and interval stop times.

The operation of the interval schedulers can be suspended by setting their administrative state attribute to locked and resumed by setting their administrative state attribute to unlocked.

7.3.2 Trigger scheduling

7.3.2.1 Periodic scheduling

Periodic scheduling is used to define a schedule that repetitively triggers specified activities at regular time intervals within specified managed object instances. The time duration over which the activities, specified in the SMOs, can be triggered, may be controlled by the specified duration start time and duration stop time (date and time). When a periodic scheduler is created, it either triggers at the specified duration start time (which may be the object creation time) or it synchronizes the first triggering point to a specified synchronization time. It then synchronizes the period to this initial triggering point.

The operation of a scheduler can be suspended and resumed by setting its administrative state attribute. Two methods of synchronization of the triggering points can be used when the operation of the scheduler is resumed, either period synchronization time or resynchronize mode. If a period synchronization time is specified, the triggering will always be synchronized to that time. If a resynchronize mode has been specified in the SO, the triggering may be synchronized to the specified duration start time, or it may be synchronized to the time of resumption of the SO, depending on the resynchronize mode selected. If period synchronization time and resynchronize mode are absent, the period will always be synchronized to the specified duration start time.

7.3.2.2 Aperiodic scheduling

An activity in a managed object can be triggered at scheduled times. This is achieved by specifying a set of trigger times for the activity rather than specifying an interval for the operation of that activity. This mechanism allows activities in a managed object to be triggered at absolute times as opposed to the triggering of activities at regular intervals relative to a start time as defined for periodic scheduling (see 7.3.2.1).

An aperiodic trigger schedule may repeat in one of the following ways: a given number of days with specified trigger times for each day, a given number of weeks with specified trigger times for specified days of each week or a given number of months with specified trigger times for specified days of each month. Each of these types of aperiodic scheduling, daily, weekly and monthly is specified by selecting the trigger times parameter for the day, week or month mask attribute in the appropriate scheduler object class.

7.3.3 Operations scheduling

In accordance with its schedule, a scheduling object which uses the operation scheduling approach determines operations performed upon SMOs.

In this case the SO may have notifications to report success and failure in the execution of the operations. A scheduling object which uses the operation scheduling approach has attributes to identify a schedule, the SMOs which are being scheduled and the operations and parameters which are to be requested in accord with the schedule. When the result notification is issued, the managed object class and managed object instance parameters shall be present in the operation result(s).

NOTE – The sending of messages between managed objects in the same system, either expressed on implied in this model, does not imply any need for conformance testing of these inter-object interactions.

7.3.4 Index scheduling

This index scheduling functions is an enhancement to the previous schedulings.

The multiScheduler scheduling extends the functionality of the interval scheduling and aperiodic scheduling:

- for interval scheduling, the multiScheduler allows the transition of an activity between several (two or more) states. Each such state is associated with an index value. The multiScheduler also allows overlapping intervals;
- for aperiodic scheduling, the multiScheduler allows the triggering of (possibly different) activities depending on an index value. Aperiodic scheduling in the multiScheduler can also be used for operation scheduling.

For both types of scheduling, the index scheduling allows to schedule activities depending on the type of day, e.g. holidays, weekdays. For this purpose, a type-Of-Day Controller is used to group days into categories according to their type, e.g. 1st January can be classified as specialDay1.

7.4 Relationships between SOs and SMOs

A SMO may be scheduled by more than one SO. In order to be scheduled by an external interval or trigger scheduler, a SMO shall have an attribute which points at the SO (the external scheduler name attribute). The SO may optionally have an attribute which points at the SMO (the scheduled managed objects attribute). SMOs which have multiple activities to

be scheduled shall have an attribute associated with each activity that points to the appropriate SOs. A single SO may provide a schedule for many SMOs. See Figure 1.

If a SMO is deleted, the entry for that object in the scheduled managed objects attribute in the related SO(s) will be deleted. If there are no remaining entries in the scheduled managed objects attribute, the SO will continue to exist. If the SO is deleted, the state of the activities of the SMO shall be as defined by the behaviour of the SMO.

Changes in the administrative and operational state of the SMO will have no effect on the SO. If the administrative state of the SO is changed to locked or the operational state is changed to disabled, the state of the activity in the SMO becomes inactive. This state may be represented by an attribute of the SMO associated with this activity. If the administrative state of the SO is changed to unlocked or the operational state is changed to enabled, the SMO is set to the status as indicated by the schedule defined for the SO.

The relationship between the SO and the SMO is established at the creation time of the SMO or when the identifier of the SO is added to the external scheduler name attribute of an existing SMO. When the SMO is created with the identifier of the SO included in the external scheduler name attribute, the identifier of the SMO instance is added to the scheduled managed objects attribute of the SO (if the SO instance supports it). The relationship may be terminated by deleting either of the objects as described above, by removing the identifier of the SO from the scheduled managed objects attribute of the SMO.

8 Generic definitions

8.1 Management information required for internal scheduling

To define a managed object class including a scheduling mechanism, the appropriate scheduling packages (i.e. periodic scheduling, daily scheduling, weekly scheduling, multiple-daily scheduling, multiple-weekly scheduling, or multiple-monthly scheduling) can be imported into the managed object class definition and tied to the appropriate activities within the behaviour clause. (The daily and weekly scheduling packages are defined in CCITT Rec. X.734 | ISO/IEC 10164-5.)

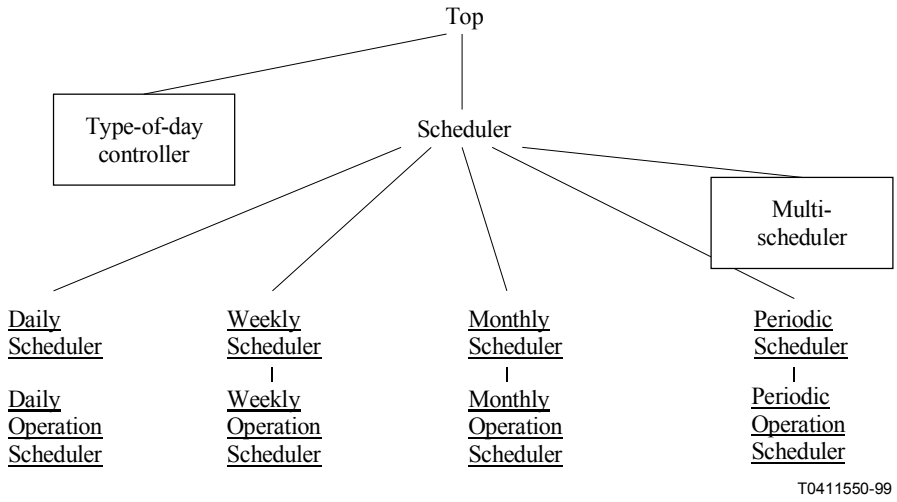
8.2 Managed objects

This Recommendation | International Standard defines a set of scheduling managed object classes. The inheritance structure of these managed object classes is shown in Figure 2.

The managed object classes shown in Table 1 are defined to fulfil the requirements of the scheduling types described in 7.3.

Table 1 – Scheduling types vs managed object classes

Type of scheduling	Managed object class(es)
Interval scheduling	Daily scheduler, weekly scheduler, monthly scheduler
Periodic scheduling (trigger)	Periodic scheduler
Aperiodic scheduling (trigger)	Daily scheduler, weekly scheduler, monthly scheduler
Operations scheduling	Daily operation scheduler, weekly operation scheduler, monthly operation scheduler, periodic operation scheduler
Index and type-of-day scheduling	Type-of-day controller, multi-scheduler



NOTE – Instantiable objects are underlined.

Figure 2 – Inheritance structure of scheduling objects

8.2.1 Scheduler

8.2.1.1 Overview

The scheduler object class is a superclass from which other scheduler object classes are derived.

8.2.1.2 Packages of the scheduler

The scheduler managed object class has the following mandatory packages:

- scheduler object package; and
- duration as defined in CCITT Rec. X.734 | ISO/IEC 10164-5.

The scheduler managed object class has the following conditional package:

- scheduled managed objects package.

8.2.2 Daily scheduler

8.2.2.1 Overview

The daily scheduler object class is a subclass of the scheduler object class. It is used to schedule intervals of activity or the aperiodic triggering of an activity of a SMO on a daily basis.

8.2.2.2 Packages of the daily scheduler

The daily scheduler managed object class has the following mandatory package:

- multiple daily scheduling.

8.2.3 Weekly scheduler

8.2.3.1 Overview

The weekly scheduler object class is a subclass of the scheduler object class. It is used to schedule intervals of activity or the aperiodic triggering of an activity of a SMO on a weekly basis.

8.2.3.2 Packages of the weekly scheduler

The weekly scheduler managed object class has the following mandatory package:

- multiple weekly scheduling.

8.2.4 Monthly scheduler

8.2.4.1 Overview

The monthly scheduler object class is a subclass of the scheduler object class. It is used to schedule intervals of activity or the aperiodic triggering of an activity of a SMO on a monthly basis.

8.2.4.2 Packages of the monthly scheduler

The monthly scheduler managed object class has the following mandatory package:

– multiple monthly scheduling.

8.2.5 Periodic scheduler

8.2.5.1 Overview

The periodic scheduler object class is a subclass of the scheduler object class. It is used to schedule triggering of an activity of a SMO on a regular periodic basis.

8.2.5.2 Packages of the periodic scheduler

The periodic scheduler managed object class has the following mandatory package:

– periodic scheduling package.

The periodic scheduler managed object class has the following conditional packages which may not both be present in an instance of the Periodic Scheduler object:

– resynchronize mode package;

– period synchronization package as defined in ITU-T Rec. X.738 | ISO/IEC 10164-13.

8.2.6 Daily operation scheduler

8.2.6.1 Overview

The daily operation scheduler object class is a subclass of the daily scheduler object class. It is used to schedule get, set and action operations on a SMO on a daily basis.

8.2.6.2 Packages of the daily operation scheduler

The daily operation scheduler managed object class has the following mandatory package:

– operations scheduling package.

The daily operation scheduler managed object class has the following conditional package:

– operation result package.

The scheduled managed objects package inherited from the scheduler object class shall not be instantiated for this object class. The value for trigger times of the sequence of days attribute shall be specified.

8.2.7 Weekly operations scheduler

8.2.7.1 Overview

The weekly operation scheduler object class is a subclass of the weekly scheduler object class. It is used to schedule get, set and action operations on a SMO on a weekly basis.

8.2.7.2 Packages of the weekly operation scheduler

The weekly operation scheduler managed object class has the following mandatory package:

– operations scheduling package.

The weekly operation scheduler managed object class has the following conditional package:

– operation result package.

The scheduled managed objects package inherited from the scheduler object class shall not be instantiated for this object class. The value for trigger times of the sequence of weeks attribute shall be specified.

8.2.8 Monthly operation scheduler

8.2.8.1 Overview

The monthly operation scheduler object class is a subclass of the monthly scheduler object class. It is used to schedule get, set and action operations on a SMO on a monthly basis.

8.2.8.2 Packages of the monthly operation scheduler

The monthly operation scheduler managed object class has the following mandatory package:

– operations scheduling package.

The monthly operation scheduler managed object class has the following conditional package:

– operation result package.

The scheduled managed objects package inherited from the scheduler object class shall not be instantiated for this object class. The value for trigger times of the sequence of months attribute shall be specified.

8.2.9 Periodic operation scheduler

8.2.9.1 Overview

The periodic operation scheduler object class is a subclass of the periodic scheduler object class. It is used to schedule operations on a SMO on a regular periodic basis.

8.2.9.2 Packages of the periodic operation scheduler

The periodic operation scheduler managed object class has the following mandatory package:

– operations scheduling package.

The periodic operation scheduler managed object class has the following conditional package:

– operation result package.

The scheduled managed objects package inherited from the scheduler object class shall not be instantiated for this object class.

8.2.10 Multi-scheduler

8.2.10.1 Overview

The multi-scheduling scheduler provides the ability to control activities for which more information is required than a simple on/off scheduling. It allows the definition of multiple independent schedules, each of which is associated with an activity. These activities are associated to index values or are triggered by operations. In the case of index values, the association between a particular index value and an activity is defined within the managed object.

Object Class: multiScheduler			
Attributes	**M/C**	**Value Set**	**Operation**
schedulingData	M	Set	GET-REPLACE ADD-REMOVE
defaultIndex	C	Single	GET-REPLACE REPLACE-WITH-DEFAULT
typeOfDayControllerInstance	C	Single	GET-REPLACE
Notifications			
"ITU-T Recommendation X.746": operationNotificationPackage	C		

The column M/C indicates whether the information presented by the attributes/actions/notifications is mandatory (M) or conditional (C).

The column "Value Set" indicates whether the attribute is single-valued or set-valued.

The column "Operation" indicates the operations that are possible on the attribute.

The following ER diagram (Figure 3) and naming tree are related to the multi-scheduler and type of day controller managed objects.

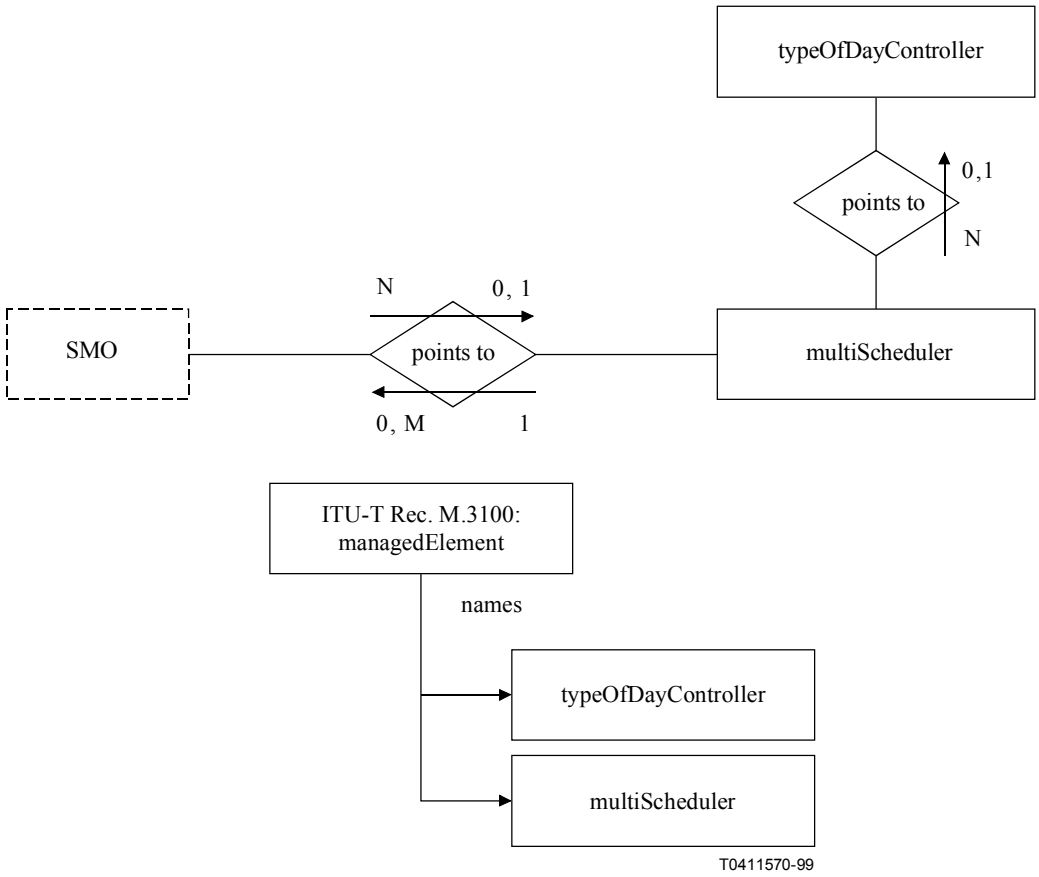


Figure 3 – Scheduled selection ER diagram and Naming Relations

8.2.10.2 Packages of the multi-scheduling scheduler

The multi-scheduling scheduler managed object class has the following mandatory package:

- multiSchedulerPackage.

The multi-scheduling scheduler managed object class has the following conditional packages:

- intervalSchedulingPackage;
- triggerSchedulingPackage;
- typeOfDayControllerInstancePackage.

The multi-scheduling scheduler managed object class has the following notification:

- OperationNotification Package.

The scheduled managed objects package inherited from the scheduler object class shall not be instantiated for this object class.

8.2.11 Type-of-day controller

8.2.11.1 Overview

The type of day controller provides management information needed to map a specific date or a weekday to a type of day and contains the currently valid TypeOfDay.

Object Class: typeOfDayController			
Attributes	**M/C**	**Value Set**	**Operation**
typeOfDayControllerId	M	Single	GET
currentTypeOfDay	M	Single	GET
dateTranslationList	M	Set	GET-REPLACE REPLACE-WITH-DEFAULT ADD-REMOVE
weekDayTranslationList	M	Set	GET-REPLACE REPLACE-WITH-DEFAULT
Notifications			
"ITU-T Recommendation M.3100: (1995)": objectManagementNotificationsPackage	M		

8.2.11.2 Packages of the type of day controller

The typeOfDayController managed object class has the following mandatory package:

– typeOfDayControllerPackage.

The typeOfDayController managed object class has the following mandatory notification:

– objectManagementNotification.

8.3 Packages

Mandatory packages must be present in all managed object instances of a managed object class while the presence of the conditional packages are determined at the time of managed object creation.

8.3.1 Scheduler object package

8.3.1.1 Overview

The Scheduler object package comprises the mandatory characteristics of the scheduler object.

8.3.1.2 Attributes of the scheduler object package

The scheduler object package has the following attributes:

a) Scheduler ID: This attribute contains a value which identifies an instance of the scheduler managed object class (used for naming).

b) Administrative state: This attribute is defined in CCITT Rec. X.731 | ISO/IEC 10164-2.

c) Operational state: This attribute is defined in CCITT Rec. X.731 | ISO/IEC 10164-2.

8.3.1.3 Notifications of the scheduler object package

The scheduler object package contains the following notifications:

– attribute value change as defined in CCITT Rec. X.730 | ISO/IEC 10164-1;

– state change as defined in CCITT Rec. X.731 | ISO/IEC 10164-2;

– object creation as defined in CCITT Rec. X.731 | ISO/IEC 10164-2; and

– object deletion as defined in CCITT Rec. X.731 | ISO/IEC 10164-2.

8.3.1.4 Behaviour of the scheduler object package

The scheduler object package provides the naming attribute for the SO using the scheduler ID attribute. It provides the ability to suspend and resume the functioning of the SO by changing the administrative state. The administrative state attribute exhibits the locked, unlocked and shutting down states. The operational state attribute exhibits the enabled and disabled operational states. The scheduler object is active if the administrative state is unlocked and the operational state is enabled.

When the administrative state is changed to locked during a scheduled interval, the interval is terminated immediately and the administrative state attribute is changed to locked. When the administrative state is changed to shut down during a scheduled interval, the interval is continued until its normal end time at which time the administrative state is changed to locked and the schedule is terminated.

Attribute value change notifications are generated when the start time and end time attributes are changed.

State change notifications are generated when the administrative state and operational state attributes are changed.

The object creation notification is generated when an instance of a managed object class which contains the Scheduler object package is instantiated.

The object deletion notification is generated when an instance of a managed object class which contains the scheduler object package is deleted.

8.3.2 Scheduled managed objects package

8.3.2.1 Overview

The scheduled managed objects package contains a list of the SMOs which the SO is currently scheduling.

8.3.2.2 Attributes of the scheduled managed objects package

The scheduled managed objects package has the following attribute:

- scheduled managed objects: This attribute identifies the SMOs which are using this SO to schedule their activities and optionally, the identifier of the attribute in the SMO that describes the activity being scheduled by the SO. Attribute value change notifications are generated when the scheduled managed objects attribute is changed.

8.3.3 Behaviour common to interval schedulers

An interval schedule comprises a collection (constructed as a sequence or a sequence of set) of schedules for a single day. Each schedule for a single day comprises a set of distinct (i.e. non-overlapping) intervals. Each of these intervals is specified as a sequence of a start time and a stop time, the values of which represent a twenty-four hour clock coordinated with the time base specified for the start time in the duration package. The stop time shall not be less than the start time. An interval may be continued into the next day by specifying a stop time of 24:00 and by specifying an interval with a start time of 0:00 for the next day.

If an activity in a managed object needs to be scheduled using a time zone base other than the local time, then the values of the start time and end time attributes shall be specified using the UTC time format of generalized time and the value of the time intervals shall be synchronized to the time specified in these attributes.

8.3.4 Multiple daily scheduling package

8.3.4.1 Overview

The multiple daily scheduling package comprises the mandatory characteristics of the daily interval scheduler object.

8.3.4.2 Attributes of the multiple daily scheduling package

The multiple daily scheduling package has the following attribute:

- Sequence of days: This attribute defines a sequence of time intervals for a day by specifying interval start and interval end times. A value of (hours = 0, minutes = 0) for start time means start of day, and the value of (hours = 0, minutes = 0) for end time means end of day (i.e. 24 hours, 0 minutes). If the value of this attribute is not specified in the create request, its value defaults to a single interval encompassing the entire 24-hour period of a day. An interval end time of hours = 0, minutes = 0 implies that the interval may continue into the following day. If the first interval start time of the next day is hours = 0, minutes = 0 then the interval is continued, otherwise it ends at the end of the day.

8.3.4.3 Multiple daily scheduling package behaviour

The multiple daily scheduling package provides the ability to automatically control an activity within a managed object. It provides the capability of scheduling the operation of an activity with a periodicity of 24 hours. A sequence of daily schedules can be defined that repeat continuously. Time intervals can be specified for specific days of a sequence.

The schedule identified by the first element in the sequence shall be implemented when the object becomes active. Each succeeding schedule shall be implemented in turn until the sequence is exhausted, when the sequence shall be repeated.

The intervals of day component within the sequence of days attribute defines the list of time intervals (interval-start and interval-end times of day) for which the scheduled activity will be operable. During excluded intervals the scheduled activity will be inactive.

If the value of the sequence of days attribute is not specified in the create request, its value is set to the default specified. This value results in the activity in the SMO being continually active.

Attribute value change notifications are generated when the sequence of days attribute is changed.

8.3.5 Multiple weekly scheduling package

8.3.5.1 Overview

The multiple weekly scheduling package comprises the mandatory characteristics of the weekly interval scheduler object.

8.3.5.2 Attributes of the multiple weekly scheduling package

The multiple weekly scheduling package has the following attribute:

– Sequence of weeks: This attribute defines a sequence of time intervals for each day of the week, as defined by a sequence of week masks. Each week mask is a set of mask components, each specifying a set of time intervals on a 24-hour time-of-day clock, pertaining to selected days of the week.

  The days of week component within the sequence of weeks attribute type defines the days of the week on which the scheduling mechanism operates. This component, if not present in a create request, will default to all seven days of the week.

  The intervals of day component within the sequence of weeks attribute type defines a list of time intervals (interval-start and interval-end times of day). A value of (hours = 0, minutes = 0) for start time means start of day, and the value of (hours = 0, minutes = 0) for end time means end of day (i.e. 24 hours, 0 minutes). If the value of this attribute is not specified in the create request, its value defaults to a single interval encompassing the entire 24-hour period of a day. An interval end time of hours = 0, minutes = 0 implies that the interval may continue into the following day. If the first interval start time of the next day is hours = 0, minutes = 0 then the interval is continued, otherwise it ends at the end of the day.

8.3.5.3 Multiple weekly scheduling package behaviour

The multiple weekly scheduling package provides the ability to automatically control an activity within a managed object. It provides the capability of scheduling the operation of an activity with a periodicity of one week. Time intervals can be specified for specified days of each week. A sequence of weekly schedules can be defined that repeat continuously.

The schedule identified by the first element in the sequence shall be implemented when the object becomes active. Each succeeding schedule shall be implemented in turn until the sequence is exhausted, when the sequence shall be repeated. A schedule for a single week comprises of a set of a sequence comprising an element which identifies days of the week and an element which identifies a schedule for a single day. Taken as a whole, this set identifies a disjoint collection of intervals spanning a whole week, beginning at 12 am on Sunday relative to the time base specified for the start time of the duration package.

The intervals of day component within the sequence of weeks attribute defines the list of time intervals (interval-start and interval-end times of day) for which the scheduled activity will be operable. During excluded intervals the scheduled activity will be inactive.

If the value of the sequence of weeks attribute is not specified in the create request, its value is set to the default specified. This value results in the activity in the SMO being continually active.

Attribute change notifications are generated when the sequence of weeks attribute is changed.

8.3.6 Multiple monthly scheduling package

8.3.6.1 Overview

The multiple monthly scheduling package comprises the mandatory characteristics of the monthly scheduler object.

8.3.6.2 Attributes of the multiple monthly scheduling package

The multiple monthly scheduling package has the following attribute:

– Sequence of months: This attribute defines a sequence of time intervals for each day of the month, as defined by a sequence of month masks. Each month mask is a set of mask components, each specifying a set of time intervals on a 24-hour time-of-day clock, pertaining to selected days of the month.

The days of month component within the sequence of months attribute type defines the days of the month on which the scheduling mechanism operates. This attribute allows for selection of days of the month forward from the first day of the month and backwards from the last day of the month. The component consists of two bit strings. The days from first bit string selects the days of the month starting from the first day of the month (i.e. the first bit in the bit string represents the first day of the month, etc.). The days from last bit string selects days of the month starting from the end of the month and working backwards from the end of the month (i.e. the first bit in this bit string represents the 30th day of the month which has 30 days while the second bit of this bit string represents the 29th day of the month which has 30 days). A day of the month is selected if either of the corresponding bits in the days from first or days from last is set.

This component, if not present in a create request, will default to every day of the month.

The intervals of day component within the Sequence of months attribute type defines a list of time intervals (interval-start and interval-end times of day). A value of (hours = 0, minutes = 0) for start time means start of day, and the value of (hours = 0, minutes = 0) for end time means end of day (i.e. 24 hours, 0 minutes). If the value of this attribute is not specified in the create request, its value defaults to a single interval encompassing the entire 24-hour period of a day. An interval end time of hours = 0, minutes = 0 implies that the interval may continue into the following day. If the first interval start time of the next day is hours = 0, minutes = 0 then the interval is continued, otherwise it ends at the end of the day.

8.3.6.3 Multiple monthly scheduling package behaviour

The multiple monthly scheduling package provides the ability to automatically control an activity within a managed object. It provides the capability of scheduling the operation of an activity with a periodicity of one month. Time intervals can be specified for specified days of each month. A sequence of monthly schedules can be defined that repeat continuously.

The schedule identified by the first element in the sequence shall be implemented when the object becomes active. Each succeeding schedule shall be implemented in turn until the sequence is exhausted, when the sequence shall be repeated. A schedule for a single month comprises of a set of a sequence comprising an element which identifies days of the month and an element which identifies a schedule for a single day. Taken as a whole, this set identifies a disjoint collection of intervals spanning a whole month, beginning at 12 am on the first of the month relative to the time base specified for the start time of the duration package. Extraneous days are ignored.

The intervals of day component within the sequence of months attribute defines the list of time intervals (interval-start and interval-end times of day) for which the scheduled activity will be operable. During excluded intervals the scheduled activity will be inactive.

If the value of the sequence of months attribute is not specified in the create request, its value is set to the default specified. This value results in the activity in the SMO being continually active.

Attribute change notifications are generated when the sequence of months attribute is changed.

8.3.7 Periodic scheduling package

8.3.7.1 Overview

The periodic scheduling package comprises the mandatory characteristics of the periodic scheduler object.

8.3.7.2 Attributes of the periodic scheduling package

The periodic scheduling package has the following attribute:

- Time period: This attribute defines the length of the time period for the periodic triggering of an activity in a SMO.

8.3.7.3 Periodic scheduling package behaviour

The periodic scheduling package provides the capability of scheduling the triggering of activities within a SMO based on a defined schedule. An activity within a SMO will be triggered by the periodic scheduler object.

If the value of the time period attribute is not specified in the create request, its value defaults to zero seconds. This means that triggering does not take place.

The operation of a scheduler can be suspended and resumed by setting its administrative state. Two methods of synchronization of the triggering points can be used when the operation of the scheduler is resumed, either period synchronization time or resynchronize mode. If a period synchronization time is specified in the period synchronization attribute the triggering will always be synchronized to that time. If the resynchronize mode attribute is present in the SO,

the triggering will be synchronized to the specified duration start time, if the resynchronize mode is false, or it will be synchronized to the time of resumption of the SO, if the resynchronize mode is true.

If the period synchronization package and the resynchronize mode package are not present in the object instance, the time period is synchronized to the start time in the duration package. If the start time in the duration package is not specified the time period is synchronized to the object creation time.

Attribute value change notifications are generated when the time period, period synchronization, or resynchronize mode attributes are changed.

8.3.8 Resynchronize mode package

8.3.8.1 Overview

The resynchronize mode package is used to specify how a periodic scheduler object is to resynchronize the triggering of the periodic schedule when the SO is resumed.

8.3.8.2 Attributes of the resynchronize mode package

The resynchronize mode package has the following attribute:

- Resynchronize mode: This attribute defines the way in which the time period is defined or redefined when the operation of the periodic scheduler is activated (i.e. the administrative state is changed to unlocked with the operational state enabled, or the operational state is changed to enabled with the administrative state unlocked).

8.3.8.3 Resynchronize mode behaviour

The resynchronize mode attribute provides the ability to control the mode of synchronization of a periodic scheduler's triggering periods upon suspension and activation or reactivation of the scheduler managed object. If the value is false, it implies that the triggering period will be synchronized to the initial triggering point or the pre-suspended triggering points when the operation of the scheduler managed object has been activated or reactivated respectively. If the value is true, it implies that, when the scheduler managed object has been created in a suspended state or placed in a suspended state after creation (i.e. locked), it triggers on resumption and synchronizes the time period to the resumption time.

8.3.9 Period synchronization package

The period synchronization package, as defined in ITU-T Rec. X.738 | ISO/IEC 10164-13, specifies the synchronization time for periods. The start for each period is at a time which is an integral number of periods before or after the period synchronization time.

8.3.10 Operations scheduling package

8.3.10.1 Overview

The operations scheduling package identifies the specific operations to be scheduled in the scheduled managed object. An SO containing this package shall determine the performance of the specified operations upon the specified SMOs in accord with the schedule supported by the SO.

8.3.10.2 Attributes of the operation scheduling package

The operations scheduling package has the following attribute:

- Operation specifications: The operation specifications attribute identifies the specific operations to be scheduled in the scheduled managed object. This read-write and set-operable (add/remove) attribute identifies SMOs and the operations to be performed upon the SMOs in accord with the schedule.

8.3.11 Operation notification package

8.3.11.1 Overview

The operation notification package contains the operation result notification which contains the results of the operation performed on the SMO.

8.3.11.2 Notifications of the operation notification package

The operation scheduling package has the following notification:

- Operation result: The operation result notification identifies the specific SMO instances and the results of the operations that were performed on the scheduled managed object. This information is contained in the operation result parameter of the notification.

8.3.12 Multi-scheduler package

8.3.12.1 Overview

The multi-scheduler package provides the ability to control activities for which more information is required than a simple on/off scheduling. It allows the definition of multiple independent schedules, each of which is associated with an activity.

8.3.12.2 Attributes of the multi-scheduler package

The multi-scheduler package has the following attribute:

- schedulingData: It contains a set of schedules and related data controlling the activities. It is a set-valued attribute and the allowed operations are GET, REPLACE, ADD, REMOVE.

8.3.13 Interval scheduling package

8.3.13.1 Overview

The interval scheduling package provides the ability to schedule activities depending on intervals and the relation of these intervals to indexes. When the interval scheduler is created or resumed at a time that is within the scheduler's defined duration, the activity within the SMO will be set according to the index defined by the interval scheduler's schedule. When the interval scheduler is deleted or suspended, or the scheduler exists at a time outside the scheduler's defined duration, the activity within the SMO will be set according to the default index.

8.3.13.2 Attributes of the interval scheduling package

The interval scheduling package has the following attribute:

- defaultIndex: The value indicated by the default index is applicable when none of the intervals are valid. This comes from the fact that, because the schedules defined in the attribute schedulingData are independent, the possibility exists that none of the intervals are valid.

8.3.14 Trigger scheduling package

8.3.14.1 Overview

The trigger scheduling package provides the ability to schedule activities based on a trigger mechanism. It is used to define restrictions to the schedulingData attribute. When the trigger scheduler is created or resumed at a time that is within the scheduler's defined duration, the first triggering of an activity within the SMO will occur according to the schedule. When the trigger scheduler is deleted or suspended, or the scheduler exists at a time outside the scheduler's defined duration, the scheduled activities within the SMO will not be triggered.

The restrictions to the schedulingData attribute are:

- TimesOfDayWps shall have the component triggerTimes;
- the optional component priority shall be absent.

8.3.14.2 Attributes of the trigger scheduling package

The trigger scheduling package has no attributes.

8.3.15 Type of day controller instance package

8.3.15.1 Overview

This type of day controller instance package provides the ability to identify the instance of a typeOfDayController OC which is valid for the instance of the multi-scheduler OC it is contained into.

8.3.15.2 Attributes of the type of day controller instance package

The type of day controller instance package has the following attribute:

- typeOfDayControllerInstance: It identifies the type of day controller instance relevant to the current multi-scheduler.

8.3.16 Type of day controller package

8.3.16.1 Overview

The type of day controller package provides the ability to define the information needed to map a specific date or a weekday to a type of day and it contains the currently valid typeOfDay.

8.3.16.2 Attributes of the type of day controller package

The type of day controller package has the following attributes:

- typeOfDayControllerId: It identifies the instance of the type of day controller OC (RDN).
- currentTypeOfDay: It indicates the currently valid typeOfDay.
- dateTranslationList: It is a table that contains the mapping of dates to typeOfDay values.
- WeekDayTranslationList: It is a table that contains the mapping of weekdays to typeOfDay values.

8.4 Properties of SMOs

8.4.1 Packages and attributes used in SMOs

The relationship of the SMO to the scheduler is represented by either:

- the external scheduler name attribute;
- an attribute derived from the external scheduler name attribute; or
- the external scheduler package,

in the SMO. The external scheduler package is defined in CCITT Rec. X.734 | ISO/IEC 10164-5. The external scheduler name attribute is defined in 8.4.1.1.

The status of the activity may be specified in a SMO using either:

- the availability status as defined in CCITT Rec. X.731 | ISO/IEC 10164-2; or
- the on duty attribute defined in 8.4.1.2.

8.4.1.1 External scheduler name attribute

The external scheduler name attribute is incorporated in a managed object definition when an activity within a SMO is to be scheduled by an external scheduler. It specifies the name of one or more external scheduler managed objects that are related to an activity within a SMO. This relationship implies that the activity will be controlled by the external scheduler object(s). If multiple activities are to be scheduled in the SMO other attributes derived from this attribute must be included in the object.

The empty set shall indicate that no scheduler object is being specified (e.g. if the SO has been deleted).

Attribute value change notifications are generated when this attribute is changed.

8.4.1.2 On duty attribute

The on duty attribute is read only and is used to indicate the status of a scheduled activity within a SMO. This attribute is used to identify the specific activity within the SMO to which the schedule applies. Its identifier is included in the scheduled managed objects attribute in the SO (see A.4.4). When the value of this attribute is true it indicates that the activity is scheduled to be active, and when the value is false it indicates that the activity is scheduled to be inactive. For each individual activity of a SMO that requires separate scheduling an activity-specific on duty attribute shall be specified which is derived from the on duty attribute (see A.4.2).

8.4.1.3 Requested window package

The requested window package is defined in ITU-T Rec. X.745 | ISO/IEC 10164-12. This package may be imported into a SMO if there is a requirement for controlling the time window within which an activity is to be performed.

8.4.2 SMO behaviour for interval scheduling

Each interval scheduler object instance can control any number of managed object instances.

When an interval scheduler is created and the scheduling relationship is set up, the activities within the SMOs will be set to no duty or off duty as defined by the interval scheduler's schedule for that particular time. If the interval scheduler is created at a time outside the schedule's defined intervals of operation, the activity within the SMO will become off-duty. The behaviour of each activity under these conditions will be defined in the behaviour clause in the SMO's class definition. One option is that any activity being performed at this time will continue to completion but no other activities will be started.

When the operation of an interval scheduler is suspended the scheduled activities within the SMOs will be off-duty. If the operation of an interval scheduler is suspended during any of the schedule's defined intervals of operation the scheduled activities within the SMOs will become off-duty. The behaviour of each activity under these conditions will be

defined in the behaviour clause in the SMO's class definition. One option is that any activity being performed at this time will continue to completion, but no other operations will be started.

For each activity that is scheduled in a SMO, a status attribute may be defined to indicate that the activity is scheduled. For an activity scheduled by one SO, if the administrative state of the corresponding SO is set to locked, or the operational state of the corresponding SO changes to disabled, the status attribute for the selected activity in the SMO is set to indicate that it is not scheduled. If there is only one activity in a SMO then the availability status defined in CCITT Rec. X.731 | ISO/IEC 10164-2 may be used to indicate the scheduled status. When the administrative state of the SO is set to unlocked or the operational state changes to enabled, the status attribute for this activity in the SMO is modified depending on the schedule defined for the SO. The definition of the SMO needs to specify how it is affected by the SOs. For example, for a SMO with one activity, the availability status is set to off-duty when the schedule in the SO indicates off duty.

When an interval scheduler is resumed the scheduled activities within the SMOs will be set to on duty or off duty as defined by the interval scheduler's schedule.

If a SMO contains its own periodic schedule it may be scheduled by an interval scheduler. In this case the periodic schedule in the SMO is only active during the intervals specified in the SO, an example is shown in Figure 4.

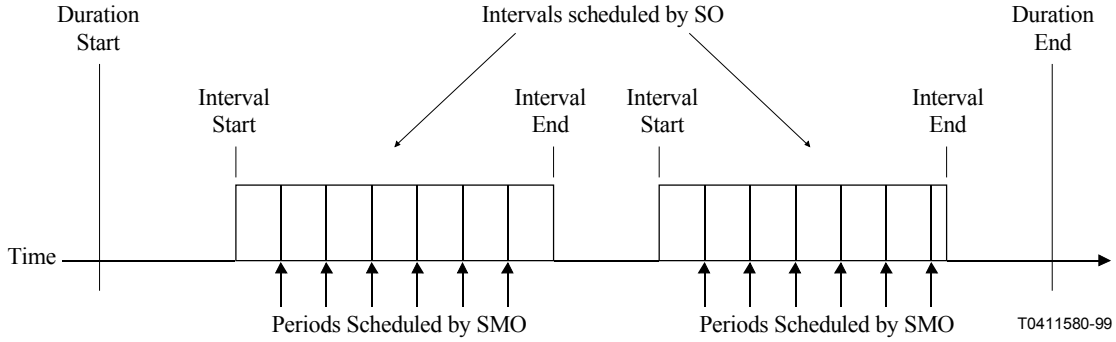


Figure 4 – Example of periodic scheduling within a scheduled interval

8.4.3 SMO behaviour for periodic and aperiodic trigger scheduling

When a trigger (periodic or aperiodic) scheduler managed object is created and the scheduling relationship is set up, the scheduled activities of the SMOs will be triggered when the scheduler starts functioning and at the appropriate periodicity or trigger times according to their schedule.

When the operation of a periodic or aperiodic scheduler is suspended, the scheduled activities within the SMOs will not be triggered.

8.4.4 SMO behaviour for index scheduling

In order to use the index scheduling mechanism, the SMO shall contain a scheduling attribute that provides the mapping of each index value to the appropriate activity in the SMO. The general form for the scheduling attribute is a SET OF SEQUENCE where the SEQUENCE contains an index component and another component (describing the activity) as simple as a numerical value or as complicated as a series of operations.

Here follows an example:

OC A (SMO)
namingAttributeA
schedulingAttribute
```
        {    -- SET OF
             {    -- SEQUENCE
                  index: 1,
                  scheduledAttribute: B1
             },
             {    -- SEQUENCE
                  index: 2,
                  scheduledAttribute: B2
             },
```

```
            {   -- SEQUENCE
                index: 3,
                scheduledAttribute: B3
            }
        }
activeScheduledAttribute
```

OC B (pointed at OC)
namingAttributeB (with possible values B1, B2, B3)

OC multiScheduler
.../... from X.746
schedulingData

```
        {   -- SET OF
            {   -- SEQUENCE
                indexOrOperSpec: index: 1,...
```

This attribute will contain scheduling information for three index values.

8.5 Compliance

Managed object class definitions support the functions defined in this Recommendation | International Standard by incorporating the specification of the management information through reference to the management templates defined in Annex A. The reference mechanism is defined in CCITT Rec. X.722 | ISO/IEC 10165-4.

8.6 Generic definitions from the object management function

This Recommendation | International Standard makes use of the following generic definitions in CCITT Rec. X.730 | ISO/IEC 10164-1:

- attribute value change notification;
- object creation notification;
- object deletion notification.

8.7 Generic definitions from the state management function

This Recommendation | International Standard makes use of the following generic definitions in CCITT Rec. X.731 | ISO/IEC 10164-2:

- administrative state;
- operational state;
- state change notification;
- availability status.

8.8 Generic definitions from the event report management function

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

- duration package;
- external scheduler package.

8.9 Generic definitions from the test management function

This Recommendation | International Standard makes use of the following generic definition in ITU-T Rec. X.745 | ISO/IEC 10164-12:

- requested time window package.

8.10 Generic definitions from the summarization function

This Recommendation | International Standard makes use of the following generic definition in ITU-T Rec. X.738 | ISO/IEC 10164-13:

- period synchronization package.

9 Service definition

This Recommendation | International Standard does not define any services. The use of services defined in other functions is listed below.

This Recommendation | International Standard uses:

- the PT-EVENT service defined in CCITT Rec. X.730 | ISO/IEC 10164-1;
- the PT-GET service defined in CCITT Rec. X.730 | ISO/IEC 10164-1;
- the PT-SET service defined in CCITT Rec. X.730 | ISO/IEC 10164-1;
- the PT-CREATE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1;
- the PT-DELETE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1;
- the object creation reporting service defined in CCITT Rec. X.730 | ISO/IEC 10164-1;
- the object deletion reporting service defined in CCITT Rec. X.730 | ISO/IEC 10164-1;
- the attribute value change reporting service defined in CCITT Rec. X.730 | ISO/IEC 10164-1; and
- the state change reporting service defined in CCITT Rec. X.731 | ISO/IEC 10164-2.

10 Functional units

The following functional units defined in CCITT Rec. X.730 | ISO/IEC 10164-1 may be negotiated for the purpose of managing scheduler objects:

- all events;
- control;
- monitor; and
- object events.

The following functional unit defined in CCITT Rec. X.731 | ISO/IEC 10164-2 may be negotiated for the purpose of managing scheduler objects:

- state change reporting.

11 Protocol and abstract syntax

11.1 Managed objects

This Recommendation | International Standard defines the following scheduler objects, the abstract syntax of which is specified in Annex A:

- scheduler;
- daily scheduler;
- weekly scheduler;
- monthly scheduler;
- periodic scheduler;
- daily operation scheduler;
- weekly operation scheduler;
- monthly operation scheduler;
- periodic operation scheduler;
- operation result;
- multi-scheduler;
- type-of-day controller.

11.2 Management attributes

This Recommendation | International Standard defines the following attributes, the abstract syntax of which is specified in Annex A:

- external scheduler name;
- on duty;
- operation specifications;
- operation result;
- resynchronize mode;
- scheduled managed objects;
- scheduler ID;
- sequence of days;
- sequence of months;
- sequence of weeks;
- time period;
- scheduling data;
- default index;
- type-of-day controller instance;
- type-of-day controller id;
- current type of day;
- date translation list;
- weekday translation list.

11.3 Management actions

There are no specific management actions defined for this systems management function.

11.4 Management notifications

This Recommendation | International Standard defines the following notification, the abstract syntax of which is specified in Annex A:

- operation result.

This Recommendation | International Standard references the following management notifications:

- attribute value change;
- state change;
- object creation;
- object deletion.

12 Relationships with other functions

This Recommendation | International Standard uses services defined in CCITT Rec. X.730 | ISO/IEC 10164-1 for the creation and deletion of managed objects, the retrieval of attributes and the notification of attribute changes, the definition in CCITT Rec. X.731 | ISO/IEC 10164-2 for the notification of state changes, the definition in ITU-T Rec. X.739 | ISO/IEC 10164-11 for the syntax of the time period attribute. It also references a number of objects, packages and attributes defined in CCITT X.721 | ISO/IEC 10165-2.

13 Conformance

Implementations claiming to conform to this Recommendation | International Standard shall comply with the conformance requirements as defined in the following subclauses.

13.1 Static conformance

The implementation shall conform to the requirements of this Recommendation | International Standard in the manager role, the agent role, or both roles. A claim of conformance to at least one role shall be made in Table C.1. If a claim of conformance is made for support in the manager role, the implementation shall support at least one of the attributes, notifications, actions, or managed objects described in Table C.2. A claim of conformance in the manager role requires the support of at least one management operation or notification as specified by those management definitions.

If a claim of conformance is made for support in the agent role, the implementation shall support at least one of the attributes, actions, or managed objects described in Table C.3. A claim of conformance in the agent role requires the support of all the mandatory operations and mandatory notifications specified by those management definitions.

The implementation shall support the transfer syntax derived from the encoding rules specified in ITU-T Rec. X.690 | ISO/IEC 8825-1 named {joint-iso-ccitt asn1(1) basicEncoding(1)} for the abstract data types referenced by the definitions for which support is claimed.

13.2 Dynamic conformance

Implementations claiming to conform to this Recommendation | International Standard shall support the elements of procedure and definitions of semantics corresponding to the definitions for which support is claimed.

13.3 Management implementation conformance statement requirements

Any MCS proforma, PICS proforma, MOCS proforma, and MIDS proforma which conforms to this Recommendation | International Standard shall be technically identical to the proformas specified in Annexes C, D, E and F preserving table numbering and the index numbers of items, and differing only in pagination and page headers.

The supplier of an implementation which is claimed to conform to this Recommendation | International Standard shall complete a copy of the management conformance summary (MCS) provided in Annex C as part of the conformance requirements together with any other ICS proformas referenced as applicable from that MCS. An MCS, MIDS, MOCS, MRCS and PICS which conform to this Recommendation | International Standard shall:

- describe an implementation which conforms to this Recommendation | International Standard;
- be completed in accordance with the instructions for completion given in ITU-T Rec. X.724 | ISO/IEC 10165-6;
- include the information necessary to uniquely identify both the supplier and the implementation.

Claims of conformance to the management information defined in this Recommendation | International Standard in managed object classes defined elsewhere shall include the requirements of the MIDS proforma in the MOCS proforma for the managed object class.

Annex A
Definition of management information

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

A.1 Object class definitions

A.1.1 Scheduler object definition

```
scheduler  MANAGED OBJECT CLASS
DERIVED FROM            "Rec. X.721 | ISO/IEC 10165-2 : 1992":top;
CHARACTERIZED BY
                schedulerObjectPackage,
                "Rec. X.721 | ISO/IEC 10165-2 : 1992":duration;
CONDITIONAL PACKAGES
                scheduledManagedObjectsPackage
                        PRESENT IF "An instance supports it.";
REGISTERED AS    {schedMo 1};
```

A.1.2 Daily scheduler object definition

```
dailyScheduler  MANAGED OBJECT CLASS
DERIVED FROM                scheduler;
CHARACTERIZED BY
            multipleDailyScheduling;
REGISTERED AS    {schedMo 2};
```

A.1.3 Weekly scheduler object definition

```
weeklyScheduler     MANAGED OBJECT CLASS
DERIVED FROM                scheduler;
CHARACTERIZED BY
            multipleWeeklyScheduling;
REGISTERED AS    {schedMo 3};
```

A.1.4 Monthly scheduler object definition

```
monthlyScheduler    MANAGED OBJECT CLASS
DERIVED FROM                scheduler;
CHARACTERIZED BY
            multipleMonthlyScheduling;
REGISTERED AS    {schedMo 4};
```

A.1.5 Periodic scheduler object definition

```
periodicScheduler    MANAGED OBJECT CLASS
DERIVED FROM                scheduler;
CHARACTERIZED BY
            periodicSchedulingPackage;
CONDITIONAL PACKAGES
            resynchronizeModePackage      PRESENT IF "an instance supports it
                and the periodSynchronizationPackage package is not
                present",
            "Rec. X.739 (1993) | ISO/IEC 10164-11:1994":periodSynchronizationPackage
                PRESENT IF "Synchronization to a specified time other than
                the duration start time is required, and the
                resynchronizeMode package is not present.";
REGISTERED AS    {schedMo 5};
```

A.1.6 Daily operation scheduler object definition

```
dailyOperationScheduler         MANAGED OBJECT CLASS
DERIVED FROM                dailyScheduler;
CHARACTERIZED BY
            operationsSchedulingPackage;
CONDITIONAL PACKAGES
            operationNotificationPackage    PRESENT IF "the results of the operation need to be
                reported or the operation performed is a GET operation";
REGISTERED AS    {schedMo 6};
```

A.1.7 Weekly operation scheduler object definition

```
weeklyOperationScheduler MANAGED OBJECT CLASS
DERIVED FROM                weeklyScheduler;
CHARACTERIZED BY
            operationsSchedulingPackage;
CONDITIONAL PACKAGES
            operationNotificationPackage    PRESENT IF "the results of the operation need to be
                reported or the operation performed is a GET operation";
REGISTERED AS    {schedMo 7};
```

A.1.8 Monthly operation scheduler object definition

```
monthlyOperationScheduler       MANAGED OBJECT CLASS
DERIVED FROM                monthlyScheduler;
CHARACTERIZED BY
            operationsSchedulingPackage;
CONDITIONAL PACKAGES
            operationNotificationPackage    PRESENT IF "the results of the operation need to be reported or
                the operation performed is a GET operation";
REGISTERED AS    {schedMo 8};
```

A.1.9 Periodic operation scheduler object definition

```
periodicOperationScheduler      MANAGED OBJECT CLASS
DERIVED FROM                    periodicScheduler;
CHARACTERIZED BY
        operationsSchedulingPackage;
CONDITIONAL PACKAGES
        operationNotificationPackage    PRESENT IF "the results of the operation need to be reported or
            the operation performed is a GET operation";
REGISTERED AS   {schedMo 9};
```

A.1.10 Operation result record object definition

```
operationResultRecord           MANAGED OBJECT CLASS
DERIVED FROM                    "Rec. X.721 | ISO/IEC 10165-2" : eventLogRecord;
CHARACTERIZED BY
        operationResultRecordPackage        PACKAGE
BEHAVIOUR operationResultRecordBehaviour BEHAVIOUR
            DEFINED AS   "see 8.3.11.2";;
        ATTRIBUTES
            operationResult GET;;;
REGISTERED AS   {schedMo 10};
```

A.1.11 multischeduler object definition

```
multiScheduler      MANAGED OBJECT CLASS
DERIVED FROM scheduler;
CHARACTERIZED BY
        multiSchedulerPackage PACKAGE
        BEHAVIOUR multiSchedulerPackageBeh BEHAVIOUR
        DEFINED AS "See section 8.2.10      Multi-scheduler.";;
        ATTRIBUTES
            schedulingData
            GET-REPLACE
            ADD-REMOVE;;;
        CONDITIONAL PACKAGES
            intervalSchedulingPackage PACKAGE
            BEHAVIOUR intervalSchedulingPackageBeh BEHAVIOUR
            DEFINED AS "Because the schedules defined in the attribute schedulingData are independent, the possibility
    exists that none of the intervals are valid. In that situation, the value indicated by the default index is applicable. The
    following restrictions apply to the schedulingData attribute:
     - index is the only valid component of IndexOrOperSpec,
     - intervalsOfDayWps is the only valid component of TimeOfDayWps.
            When the interval scheduler is created or resumed at a time that is within the scheduler's defined duration, the activity
    within the SMO will be set according to the index defined by the interval scheduler's schedule. When the interval scheduler
    is deleted or suspended, or the scheduler exists at a time outside the scheduler's defined duration, the activity within the
    SMO will be set according to the default index.";;
            ATTRIBUTES
            defaultIndex
            REPLACE-WITH-DEFAULT
            DEFAULT VALUE Schedulerev1-ASN1Module.defaultDefaultIndex
            GET-REPLACE;
            REGISTERED AS { schedPkg 1};
            PRESENT IF "Instance provides interval scheduling",
                triggerSchedulingPackage PACKAGE
                BEHAVIOUR triggerSchedulingPackageBeh BEHAVIOUR
                DEFINED AS "The following restrictions apply to the schedulingData attribute:
            - TimesOfDayWps shall have the component triggerTimes;
            - the optional component priority shall be absent.
            When the trigger scheduler is created or resumed at a time that is within the scheduler's defined duration, the first
    triggering of an activity within the SMO will occur according to the schedule. When the trigger scheduler is deleted or
    suspended, or the scheduler exists at a time outside the scheduler's defined duration, the scheduled activities within the SMO
    will not be triggered.";;
                REGISTERED AS { schedPkg 2};
                PRESENT IF "Instance provides trigger scheduling",
                TypeOfDayControllerInstancePackage PACKAGE
                BEHAVIOUR typeOfDayControllerInstancePackageBeh BEHAVIOUR
                DEFINED AS "The attribute typeOfDayControllerInstance identifies the instance of OC
     typeOfDayController that is relevant for this instance of OC multiScheduler.";;
                ATTRIBUTES
                typeOfDayControllerInstance
```

```
                GET-REPLACE;
                REGISTERED AS { schedPkg 3};
        PRESENT IF "More than one instance of OC typeOfDayController can exist in managedElement and if scheduler
    instance uses 'type of day' scheduling",
        "ITU-T Recommendation X.746":operationNotificationPackage
        PRESENT IF "the results of an operation need to be reported or an operation performed is a GET
    operation.";
REGISTERED AS { schedMo 11};
```

A.1.12 typeOfDayController

```
typeOfDayController MANAGED OBJECT CLASS
        DERIVED FROM "ITU-T Recommendation X.721:1992":top;
        CHARACTERIZED BY
            typeOfDayControllerPackage PACKAGE
            BEHAVIOUR   typeOfDayControllerPackageBeh BEHAVIOUR
        DEFINED AS "As in section Type of day controller. The mapping is maintained in the attributes
    'dateTranslationList' and 'weekDayTranslationList'. The typeOfDay value that is specified for specific dates (via
dateTranslationList attribute) has precedence on the typeOfDay value for a week day (via weekDayTranslationList
attribute). ";;
        ATTRIBUTES
            typeOfDayControllerId
            GET,
            currentTypeOfDay
            INITIAL VALUE DERIVATION RULE
                currentTypeOfDayAlgorithm
            GET,
            dateTranslationList
            REPLACE-WITH-DEFAULT
            DEFAULT VALUE Schedulerev1-ASN1Module.defaultDateTranslationList
            GET-REPLACE
            ADD-REMOVE,
            weekDayTranslationList
            REPLACE-WITH-DEFAULT
            DEFAULT VALUE Schedulerev1-ASN1Module.defaultWeekDayTranslationList
            GET-REPLACE;;,
        "ITU-T Recommendation M.3100: 1995":objectManagementNotificationsPackage;
REGISTERED AS { schedMo 11};
```

A.2 Name bindings

Additional Name Bindings may be defined and registered for each scheduler object class.

A.2.1 Scheduler name binding

The following NAME-BINDING templates provide the name binding currently defined for naming instances of the scheduler managed object classes.

```
scheduler-system      NAME BINDING
        SUBORDINATE OBJECT CLASS
        scheduler         AND SUBCLASSES;
        NAMED BY
        SUPERIOR OBJECT CLASS
            "Rec. X.721 | ISO/IEC 10165-2 : 1992":system
                    AND SUBCLASSES;
        WITH ATTRIBUTE
            schedulerID;
        CREATE
        WITH-REFERENCE-OBJECT,
        WITH-AUTOMATIC-INSTANCE-NAMING
        "Rec. X.738|ISO/IEC 10164-13:1992":conflictingPackagesRequestedError;
        DELETE
        ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS {schedNb 1};
scheduler-managedElement NAME BINDING
        SUBORDINATE OBJECT CLASS
        scheduler         AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS
            "ITU-T Recommendation M.3100": managedElement
                    AND SUBCLASSES;
        WITH ATTRIBUTE
```

```
            schedulerID;
        CREATE
        WITH-REFERENCE-OBJECT,
        WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE;
REGISTERED AS { schedNb 2};
```

A.2.2 type of day controller name-binding

The following NAME-BINDING templates provide the name binding currently defined for naming instances of the typeOfDayController managed object classes.

```
typeOfDayController-managedElement NAME BINDING
        SUBORDINATE OBJECT CLASS
        typeOfDayController AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS
            "ITU-T Recommendation M.3100": managedElement
                        AND SUBCLASSES;
        WITH ATTRIBUTE
            typeOfDayControllerId;
        CREATE
        WITH-REFERENCE-OBJECT,
        WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE;
REGISTERED AS { schedNb 3};

typeOfDayController-system NAME BINDING
        SUBORDINATE OBJECT CLASS
        TypeOfDayController     AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS
            "Rec. X.721 | ISO/IEC 10165-2 : 1992":system AND SUBCLASSES;
        WITH ATTRIBUTE
            typeOfDayControllerId;
        CREATE
        WITH-REFERENCE-OBJECT,
        WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE;
REGISTERED AS { schedNb 4};
```

A.3 Packages

A.3.1 Multiple daily scheduling package

```
multipleDailyScheduling     PACKAGE
        BEHAVIOUR       multipleDailySchedulingBehaviour    BEHAVIOUR
        DEFINED AS      "See 8.3.3 and 8.3.4.3";;
        ATTRIBUTES
        sequenceOfDays          DEFAULT VALUE
                        Schedulerev1-ASN1Module.defaultSequenceOfDays
            GET-REPLACE;
REGISTERED AS {schedPkg 10};
```

A.3.2 Multiple monthly scheduling package

```
multipleMonthlyScheduling   PACKAGE
        BEHAVIOUR       multipleMonthlySchedulingBehaviour      BEHAVIOUR
        DEFINED AS      "See 8.3.3 and 8.3.6.3";;
        ATTRIBUTES
        sequenceOfMonths        DEFAULT VALUE
            Schedulerev1-ASN1Module.defaultSequenceOfMonths
            GET-REPLACE;
REGISTERED AS {schedPkg 11};
```

A.3.3 Multiple weekly scheduling package

```
multipleWeeklySchedulingPACKAGE
        BEHAVIOUR       multipleWeeklySchedulingBehaviour BEHAVIOUR
        DEFINED AS      "See 8.3.3 and 8.3.5.3";;
        ATTRIBUTES
        sequenceOfWeeks         DEFAULT VALUE
                        Schedulerev1-ASN1Module.defaultSequenceOfWeeks
```

```
            GET-REPLACE;
REGISTERED AS {schedPkg 12};
```

A.3.4 Periodic scheduling package

```
periodicSchedulingPackage   PACKAGE
        BEHAVIOUR           periodicSchedulingBehaviour    BEHAVIOUR
        DEFINED AS          "See 8.3.7.3";;
        ATTRIBUTES
        timePeriod          DEFAULT VALUE
                            Schedulerev1-ASN1Module.defaultTimePeriod
            GET-REPLACE;
REGISTERED AS {schedPkg 4};
```

A.3.5 Resynchronize mode package

```
resynchronizeModePackage    PACKAGE
        BEHAVIOUR           resynchronizeModePackageBehaviourBEHAVIOUR
        DEFINED AS          "See 8.3.8.3";;
        ATTRIBUTES
        resynchronizeMode       GET-REPLACE;
REGISTERED AS {schedPkg 5};
```

A.3.6 Scheduled managed objects package

```
scheduledManagedObjectsPackage    PACKAGE
        BEHAVIOUR           scheduledManagedObjectBehaviour   BEHAVIOUR
        DEFINED AS          "See 8.3.2";;
        ATTRIBUTES          scheduledManagedObjects GET;
REGISTERED AS {schedPkg 6};
```

A.3.7 Scheduler object package

```
schedulerObjectPackage      PACKAGE
        BEHAVIOUR           schedulerObjectBehaviour BEHAVIOUR
        DEFINED AS          "See 8.3.1.4";;
        ATTRIBUTES
        schedulerID     GET,
        "Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState
                            GET-REPLACE,
        "Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState     GET;
        NOTIFICATIONS
        "Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange,
        "Rec. X.721 | ISO/IEC 10165-2 : 1992":stateChange,
        "Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,
        "Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion;
REGISTERED AS {schedPkg 7};
```

A.3.8 Operations scheduling package

```
operationsSchedulingPackage PACKAGE
        BEHAVIOUR           operationsSchedulingBehaviour BEHAVIOUR
        DEFINED AS          "See 8.3.10";;
        ATTRIBUTES
        operationSpecifications     GET-REPLACE ADD-REMOVE;
REGISTERED AS {schedPkg 8};
```

A.3.9 Operation notification package

```
operationNotificationPackage    PACKAGE
        BEHAVIOUR           operationNotificationBehaviour  BEHAVIOUR
        DEFINED AS          "See 8.3.11";;
        NOTIFICATIONS
        operationResultNotification;
REGISTERED AS {schedPkg 9};
```

A.4 Attributes

A.4.0.1 current type of day attribute

```
currentTypeOfDay ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.TypeOfDay;
```

```
        MATCHES FOR EQUALITY;
REGISTERED AS {schedAtt 12};
```

A.4.0.2 date translation list attribute

```
dateTranslationList ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.DateTranslationList;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR dateTranslationListBeh BEHAVIOUR
                DEFINED AS
"The translation list is a set that contains the mapping of a particular date into a type of day. The typeOfDay can be either a weekday type ('workday' or 'weekend'), or one of the special days that are defined. A specific date value may only occur once in dateTranslationList attribute.";;
REGISTERED AS {schedAtt 13};
```

A.4.0.3 default index attribute

```
defaultIndex ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.DefaultIndex;
        MATCHES FOR EQUALITY;
        BEHAVIOUR defaultIndexBeh BEHAVIOUR
                DEFINED AS
"As in section 8.2.10  Multi-scheduler.";;
REGISTERED AS {schedAtt 14};
```

A.4.1 External scheduler name attribute

This attribute is included in Scheduled Managed objects. It specifies the SO instance that controls the activity in the SMO.

```
externalSchedulerName     ATTRIBUTE
                WITH ATTRIBUTE SYNTAX
                  Schedulerev1-ASN1Module.ExternalSchedulerName;
                MATCHES FOR EQUALITY;
                BEHAVIOUR  externalSchedulerNameBehaviour BEHAVIOUR
                    DEFINED AS "See 8.4.1.1";;
REGISTERED AS {schedAtt 1};
```

A.4.2 On duty attribute

This attribute is included in SMOs. It specifies the current status of the activity in the SMO.

```
onDuty     ATTRIBUTE
                WITH ATTRIBUTE SYNTAX
                  Schedulerev1-ASN1Module.OnDuty;
                MATCHES FOR EQUALITY;
                BEHAVIOUR onDutyBehaviour BEHAVIOUR
                    DEFINED AS "See 8.4.1.2";;
REGISTERED AS {schedAtt 2};
```

A.4.3 Resynchronize mode attribute

This attribute defines the mode of synchronization of a periodic scheduler's triggering periods.

```
resynchronizeMode   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.ResynchronizeMode;
        MATCHES FOR EQUALITY;
        BEHAVIOUR   resynchronizeModeBehaviour    BEHAVIOUR
        DEFINED AS   "See 8.3.8.2";;
REGISTERED AS {schedAtt 3};
```

A.4.4 Scheduled managed objects attribute

This attribute is defined in the Scheduler object to specify the SMO instances and the attribute identifier of the attribute associated with the activities within those instances that are controlled by the SO.

```
scheduledManagedObjects ATTRIBUTE
        WITH ATTRIBUTE SYNTAX
            Schedulerev1-ASN1Module.ScheduledManagedObjectsList;
        MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
        BEHAVIOUR          scheduledManagedObjectsBehaviour  BEHAVIOUR
        DEFINED AS         "See 8.3.2.2";;
REGISTERED AS {schedAtt 4};
```

A.4.5 Scheduler ID attribute

This attribute is the distinguished attribute for naming instances of a SO.

```
schedulerID      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SimpleNameType;
        MATCHES FOR EQUALITY, SUBSTRINGS;
        BEHAVIOUR   schedulerIDBehaviour       BEHAVIOUR
        DEFINED AS   "See 8.3.1.2";;
REGISTERED AS {schedAtt 5};
```

A.4.6 scheduling data attribute

```
schedulingData ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SchedulingData;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
        BEHAVIOUR schedulingDataBeh BEHAVIOUR
            DEFINED AS "As in section 8.2.10    Multi-scheduler.
        Intervals within one member of the set shall be non-overlapping, but intervals defined in different members of the set can
overlap. In that case, the (optional) sequence member 'priority' shall be present in each member of the set containing an interval that
overlaps. The priority associated with each member of the set determines which activity will be scheduled. The higher priority
numerical value has precedence on the lower one.";;
REGISTERED AS {schedAtt 15};
```

A.4.7 Sequence of days attribute

This structured attribute defines a sequence of intervals of day.

```
sequenceOfDays       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SequenceOfDays;
        MATCHES FOR EQUALITY;
        BEHAVIOUR   sequenceOfDaysBehaviourBEHAVIOUR
        DEFINED AS   "See 8.3.4.2";;
REGISTERED AS {schedAtt 6};
```

A.4.8 Sequence of months attribute

This structured attribute defines a sequence of month masks.

```
sequenceOfMonths    ATTRIBUTE
        WITH ATTRIBUTE SYNTAX
                Schedulerev1-ASN1Module.SequenceOfMonths;
        MATCHES FOR EQUALITY;
        BEHAVIOUR   sequenceOfMonthsBehaviour     BEHAVIOUR
        DEFINED AS   "See 8.3.6.2";;
REGISTERED AS {schedAtt 7};
```

A.4.9 Sequence of weeks attribute

This structured attribute defines a sequence of week masks.

```
sequenceOfWeeks     ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SequenceOfWeeks;
        MATCHES FOR EQUALITY;
        BEHAVIOUR   sequenceOfWeeksBehaviour      BEHAVIOUR
        DEFINED AS   "See 8.3.5.2";;
REGISTERED AS {schedAtt 8};
```

A.4.10 Time period attribute

This attribute defines the length of the time period for the periodic triggering of an activity in a SMO by the Periodic scheduling object.

```
timePeriod       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.TimePeriod;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR   timePeriodBehaviour BEHAVIOUR
        DEFINED AS   "See 8.3.7.2";;
REGISTERED AS {schedAtt 9};
```

A.4.11 Operation specifications attribute

This attribute defines the operations which may be scheduled for a SMO by an operations scheduling object.

```
operationSpecifications      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX
            Schedulerev1-ASN1Module.OperationSpecifications;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
        BEHAVIOUR   operationSpecificationsBehaviour      BEHAVIOUR
        DEFINED AS   "See 8.3.10.2";;
REGISTERED AS {schedAtt 10};
```

A.4.12 Operation result attribute

This attribute is included in operation result record objects. It specifies the result of operations performed by operations schedulers.

```
operationResult ATTRIBUTE
        WITH ATTRIBUTE SYNTAX
          Schedulerev1-ASN1Module.OperationResult;
        MATCHES FOR EQUALITY;
        BEHAVIOUR operationResultBehaviour BEHAVIOUR
        DEFINED AS "See 8.3.11.2";;
REGISTERED AS {schedAtt 11};
```

A.4.13 type of day controller instance attribute

```
typeOfDayControllerInstance ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.ObjectInstance;
        MATCHES FOR EQUALITY;
BEHAVIOUR typeOfDayControllerInstanceBeh BEHAVIOUR
                DEFINED AS
                    "This attribute points to an instance of OC typeOfDayController.";;
REGISTERED AS {schedAtt 16};
```

A.4.14 type of day controller id attribute

```
typeOfDayControllerId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.NameType;
        MATCHES FOR EQUALITY;
REGISTERED AS {schedAtt 17};
```

A.4.15 week day translation list attribute

```
weekDayTranslationList ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.
WeekDayTranslationList;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR weekDayTranslationListBeh BEHAVIOUR
                DEFINED AS
"The translation list is a set that contains the mapping of a particular day of the week into a type of day. The typeOfDay can be either a weekday type ('workday' or 'weekend'), or one of the special days that are defined. In weekDayTranslationList attribute, there shall be one mapping for every week day and a weekday value shall only occur once.";;
REGISTERED AS {schedAtt 18};
```

A.4.16 Behaviour for current TypeOfDay initial value

```
currentTypeOfDayAlgorithm BEHAVIOUR
            DEFINED AS
"The currentTypeOfDay attribute is determined in accordance to the value of the attributes 'dateTranslationList' and
'weekDayTranslationList'.";
```

A.5 Notifications

A.5.1 Operation result notification

```
operationResultNotification NOTIFICATION
        BEHAVIOUR operationResultNotificationBehaviour BEHAVIOUR
            DEFINED AS "See 8.3.11.2";;
        WITH INFORMATION SYNTAX Schedulerev1-ASN1Module.OperationResult;
        REGISTERED AS {schedNotif 1};
```

A.6 ASN.1 definitions

Schedulerev1-ASN1Module { joint-iso-itu-t ms(9) function(2) part15(15) modules (2)}

```
DEFINITIONS IMPLICIT TAGS ::=

BEGIN
-- EXPORTS everything

IMPORTS

CMIP-ATTRIBUTE, AttributeSet,
Attribute,AttributeId,ObjectClass,ObjectInstance,ActionResult,SetResult,SetListError,ActionError,ModifyOperator,ActionI
nfo,GetResult,GetListError,GetArgument FROM CMIP-1
        { joint-iso-itu-t ms(9) cmip(1) modules(0) protocol(3) }

DMI-TYPE-IDENTIFIER, SimpleNameType,defaultStopTime FROM Attribute-ASN1Module
        { joint-iso-itu-t ms(9) smi(3) part2(2) asn1Module(2) 1}

TimePeriod FROM MetricModule
{ joint-iso-itu-t ms(9) function(2) part11(11) asn1Module(2) 0};

schedMo  OBJECT IDENTIFIER  ::=
        {joint-iso-itu-t ms(9) function(2) part15(15) managedObjectClass(3)}

schedAtt  OBJECT IDENTIFIER  ::=
        {joint-iso-itu-t ms(9) function(2) part15(15) attribute(7)}

schedNotif      OBJECT IDENTIFIER  ::=
        {joint-iso-itu-t ms(9) function(2) part15(15) notification(10)}

schedPkg OBJECT IDENTIFIER  ::=
        {joint-iso-itu-t ms(9) function(2) part15(15) package(4)}

schedNb  OBJECT IDENTIFIER  ::=
        {joint-iso-itu-t ms(9) function(2) part15(15) nameBinding(6)}
-- default value definitions

defaultDateTranslationList     DateTranslationList ::= {}

defaultDaysOfMonth     DaysOfMonth ::= {
        daysFromFirst '11111111111111111111111111111111'B,
                daysFromLast '11111111111111111111111111111111'B }

defaultDefaultIndex        DefaultIndex ::=0
defaultIntervalsOfDayWps     IntervalsOfDayWps ::= {{
                             intervalStart {hour 0, minute 0, second 0},
                             intervalEnd {hour 0, minute 0, second 0}}}

defaultResynchronizeMode     ResynchronizeMode::=   FALSE

defaultSequenceOfDays   SequenceOfDays ::= {  intervalsOfDayWps:defaultIntervalsOfDayWps }

defaultSequenceOfMonths      SequenceOfMonths ::= {{{
                             daysOfMonth  defaultDaysOfMonth,
                             timesOfDayWps      intervalsOfDayWps:defaultIntervalsOfDayWps }}}

defaultSequenceOfWeeksSequenceOfWeeks ::= {defaultWeekMaskWps}

defaultTimePeriod          TimePeriod ::= seconds: 0

defaultWeekDayTranslationList      WeekDayTranslationList ::= {
        {weekDay sunday,   typeOfDay      workday},
        {weekDay monday,   typeOfDay      workday},
        {weekDay tuesday,  typeOfDay      workday},
        {weekDay wednesday,      typeOfDay      workday},
        {weekDay thursday, typeOfDay      workday},
        {weekDay friday,   typeOfDay      workday},
        {weekDay saturday, typeOfDay      weekend}}
defaultWeekMaskWps           WeekMaskWps      ::=   {{
                             daysOfWeek         '1111111'B,
                              timesOfDayWps     intervalsOfDayWps:defaultIntervalsOfDayWps }}
-- supporting productions

Date                       ::= VisibleString (SIZE(8))
-- i.e. YYYYMMDD
-- according to example in ITU-T Rec. X.680 | ISO/IEC 8824-1
DateTranslationList                ::= SET OF SEQUENCE {
                           date            [1]  Date,
                           typeOfDay           [2] TypeOfDay }
```

```
DaysOfMonth ::=    SEQUENCE {
                        daysFromFirst [0]  BIT STRING (SIZE(31)),
                        daysFromLast  [1]  BIT STRING (SIZE(31))}

DaysOfWeek   ::=   BIT STRING   {
                   sunday(0),monday(1),tuesday(2),wednesday(3),thursday(4),
                   friday(5),saturday(6)}      (SIZE(7))

DefaultIndex              ::= INTEGER
ExternalSchedulerName ::=    SET OF    ObjectInstance

IndexOrOperSpec                ::= CHOICE {
                        index                  [0] INTEGER,
                        operation [1] OperationSpecifications}

IntervalsOfDayWps ::=   SET OF SEQUENCE {
                        intervalStart    Time24Wps,
                        intervalEnd           Time24Wps}

MonthMask ::=           SET OF SEQUENCE {
                        daysOfMonth   DaysOfMonth,
                        timesOfDayWps      TimesOfDayWps}

-- Specifies the TimesOfDayWps for different days of the same month if
-- they have different intervals.

ModificationList ::= SET OF SEQUENCE{
          modifyOperator          [2] IMPLICIT ModifyOperator DEFAULT replace,
          attributeId             CMIP-ATTRIBUTE.&id({AttributeSet}),
          attributeValue          CMIP-ATTRIBUTE.&Value ({AttributeSet} {@.attributeId}) OPTIONAL
                                  -- absent for setToDefault -- }

OnDuty ::=     BOOLEAN

OperationSpecifications ::=    SET OF SEQUENCE{
                scheduledObjects          SET OF ObjectInstance,
                scheduledOperations       CHOICE{
                set                       [0] IMPLICIT ModificationList,
                action                    [1] IMPLICIT ActionInfo,
                get                       [2] IMPLICIT GetArgument}}

OperationResult ::= SET OF CHOICE{
                getResult                 [0] IMPLICIT GetResult,
                getListError              [1] IMPLICIT GetListError,
                setResult                 [2] IMPLICIT SetResult,
                setListError              [3] IMPLICIT SetListError,
                actionResult              [4] IMPLICIT ActionResult,
                commonError               [5] IMPLICIT CommonError}

SCHED-ERROR ::= CLASS {
          &id       ErrorId UNIQUE,
          &Value                    }
CommonError ::=   SEQUENCE{
                managedObjectClass              ObjectClass,
                managedObjectInstance           ObjectInstance,
                errorId                         SCHED-ERROR.&id ({CommonErrorSet}),
                errorValue                      SCHED-ERROR.&Value ({CommonErrorSet} {@.errorId}) OPTIONAL}
CommonErrorSet SCHED-ERROR ::= {...}
ErrorId ::= CHOICE {
          localValue          INTEGER,
          globalValue    OBJECT IDENTIFIER}

Schedule                       ::= CHOICE {
                        daily      [0]   SequenceOfDays,
                        weekly          [1]   SequenceOfWeeks,
                        monthly    [2]   SequenceOfMonths,
                        typeOfDay       [3]   SequenceOfTypeOfDays}

ScheduledManagedObjectsList ::=         SET OF SEQUENCE {
                        objectInstance ObjectInstance,
                        activity    AttributeId OPTIONAL}
```

```
SchedulingData              ::= SET OF SEQUENCE {
                            indexOrOperSpec   [0]   IndexOrOperSpec,
                            schedule  [1]   Schedule,
                            priority          [2]   INTEGER OPTIONAL}
ResynchronizeMode ::=  BOOLEAN

SequenceOfDays ::=SEQUENCE OF TimesOfDayWps
SequenceOfWeeks ::=     SEQUENCE OF WeekMaskWps
SequenceOfMonths ::=    SEQUENCE OF MonthMask

SequenceOfTypeOfDays        ::= SEQUENCE SIZE(1) OF TypeOfDaysMaskWps
-- SIZE(1) because more has no meaning because there is no implied periodicity, contrary to
-- SequenceOfDays, SequenceOfWeeks, SequenceOfMonths.

Time24Wps ::=       SEQUENCE {
                    hour                [1]   INTEGER  (0..23),
                    minute              [2]   INTEGER  (0..59) OPTIONAL,
                    second              [3]   INTEGER  (0..59) OPTIONAL,
                    milliseconds        [4]   INTEGER  (0..999) OPTIONAL,
                    microseconds        [5]   INTEGER  (0..999999) OPTIONAL,
                    nanoseconds         [6]   INTEGER  (0..999999999) OPTIONAL,
                    picoseconds         [7]   INTEGER  (0..999999999999) OPTIONAL}

TimesOfDayWps ::=       CHOICE {
                        intervalsOfDayWps [0] IMPLICIT IntervalsOfDayWps,
                    triggerTimes    [1] IMPLICIT TriggerTimes}

TriggerTimes ::=        SET OF    Time24Wps

TypeOfDay                   ::= INTEGER {
                            workday  (0),
                            weekend  (1),
                            specialDay1     (2),
                            specialDay2     (3),
                            specialDay3     (4),
                            specialDay4     (5),
                            specialDay5     (6),
                            specialDay6     (7),
                            specialDay7     (8),
                            specialDay8     (9),
                            specialDay9     (10),
                            specialDay10    (11),
                            specialDay11    (12),
                            specialDay12    (13),
                            specialDay13    (14),
                            specialDay14    (15),
                            specialDay15    (16),
                            specialDay16    (17) }

TypeOfDaysMaskWps               ::= SET OF SEQUENCE {
                            typeOfDay       TypeOfDay,
                            timesOfDay      TimesOfDayWps}

WeekDay                     ::= ENUMERATED {
            sunday          (0),
            monday          (1),
            tuesday         (2),

            wednesday       (3),
            thursday        (4),
            friday          (5),
            saturday        (6)}
WeekDayTranslationList      ::= SET OF SEQUENCE {
                            weekDay  [1]    WeekDay,
                            typeOfDay               [2] TypeOfDay}
WeekMaskWps ::=             SET OF SEQUENCE {
                                daysOfWeek          DaysOfWeek,
                                timesOfDayWps       TimesOfDayWps}
END     -- end of supporting productions
```

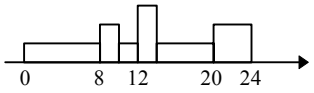
Annex B
Example of schedulingData values

(This annex does not form an integral part of this Recommendation | International Standard)

B.1 Example 1: Shows several index values use

B.1.1 Interval solution

Every day from 8:00 to 10:00 and 20:00 to 24:00, use index 1. From 12:00 to 14:00, use index 2. Otherwise, use index 0.

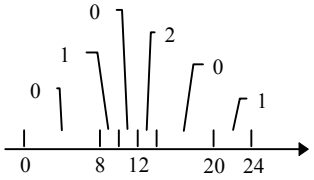


The attribute schedulingData would look like:

```
schedulingData:
        { -- SET OF
            { -- SEQUENCE
                indexOrOperSpec: index: 1,
                schedule: daily: -- daily, so
                { -- SEQUENCE OF TimesOfDayWps
                    intervalsOfDayWps:
                    { -- SET OF
                        {
                            intervalStart: { hour: 8 },
                            intervalEnd: { hour: 10 }
                        },
                        {
                            intervalStart: { hour: 20 },
                            intervalEnd: { hour: 0 }
                        }
                    }
                }
                -- no priority
            },
            { -- SEQUENCE
                indexOrOperSpec: index: 2,
                schedule: daily: -- daily, so
                { -- SEQUENCE OF TimesOfDayWps
                    intervalsOfDayWps:
                    { -- SET OF
                        {
                            intervalStart: { hour: 12 },
                            intervalEnd: { hour: 14 }
                        }
                    }
                },
                -- no priority
            }
        }
defaultIndex: 0
```

NOTE – Priority is not required as intervals are not overlapping.

B.1.2 Trigger solution



The value of the attribute schedulingData would look like:

```
        {   -- SET OF
            {   -- SEQUENCE
                indexOrOperSpec: index: 0,
                schedule: daily: -- daily, so
```

```
                {     -- SEQUENCE OF TimesOfDayWps
                      triggerTimes:
                      {     -- SET OF Times24Wps
                            {hour: 0},
                            {hour : 10},
                            {hour : 14}
                      }
                }
                -- no priority
          },
          {     -- SEQUENCE
                indexOrOperSpec: index: 1,
                schedule: daily: -- daily, so
                {     -- SEQUENCE OF TimesOfDayWps
                      triggerTimes:
                      {     -- SET OF Times24Wps
                            {hour : 8},
                            {hour : 20}
                      }
                }
                -- no priority
          },
          {     -- SEQUENCE
                indexOrOperSpec: index: 2,
                schedule: daily: -- daily, so
                {     -- SEQUENCE OF TimesOfDayWps
                      triggerTimes:
                      {     -- SET OF Times24Wps
                            {hour : 12}
                      }
                }
                -- no priority
          }
    }
```

B.2 Example 2: Shows typeOfDay use

Index 3 for every workday from 0:00 to 6:00 and from 20:00 to 00:00, and for the weekends the entire day. Otherwise, use index 0.

B.2.1 Interval solution

```
schedulingData:
      { -- SET OF
            { -- SEQUENCE
                  indexOrOperSpec: index: 3,
                  schedule: typeOfDay: -- typeOfDay, so
                  { -- SEQUENCE OF TypeOfDaysMaskWps
                        { -- SET OF
                              {
                                    typeOfDays: { workday },
                                    timesOfDay: intervalsOfDayWps:
                                    { -- SET OF
                                          {
                                                intervalStart: { hour: 0 },
                                                intervalEnd: { hour: 6 }
                                          },
                                          {
                                                intervalStart: { hour 20 },
                                                intervalEnd: { hour: 0 }
                                          }
                                    }
                              },
                              {
                                    typeOfDays: { weekend },
                                    timesOfDay: intervalsOfDayWps:
                                    { -- SET OF
                                          {
                                                intervalStart: { hour: 0 },
                                                intervalEnd: { hour: 0 }
```

```
                                        }
                                    }
                                }
                            }
                        },
                        -- no priority
                    }
                },
defaultIndex: 0
```

NOTE – Priority is not required as intervals are not overlapping.

B.2.2 Trigger solution

Trigger for index = 3 on workday 0:00 and 20:00 and trigger for index = 0 on workday 6:00. (Implicitly entire weekend will have index 3).

The value of the attribute schedulingData would look like:

```
        {   -- SET OF
            {   -- SEQUENCE
                indexOrOperSpec: index: 3,
                schedule: typeOfDay -- typeOfDay, so
                {   -- SEQUENCE OF TypeOfDaysMaskWps
                    {   -- SET OF
                        {   -- SEQUENCE
                        typeOfDays: {workday},  --workday,
                        timesOfDay : triggerTimes:
                            {   -- SET OF Time24Wps
                                {hour: 20}
                            }
                        }
                    }
                }
                -- no priority
            },
            {   -- SEQUENCE
                indexOrOperSpec: index: 0,
                schedule: typeOfDay -- typeOfDay, so
                {   -- SEQUENCE OF TypeOfDaysMaskWps
                    {   -- SET OF
                        {   -- SEQUENCE
                        typeOfDays: {workday},  --workday,
                        timesOfDay : triggerTimes:
                            {   -- SET OF Time24Wps
                                {hour: 6}
                            }
                        }
                    }
                }
                -- no priority
            }
    }
```

B.3 Example 3: Shows overlapping intervals, implicit repetition (months periodicity) and mixed scheduler types

Use index 2 from the second day of the month starting at 8:00 until the fifth day of the month ending at 18:00, for every third month (March, June, September, December). Otherwise, use index 3 on Mondays and Fridays from 12:00 to 16:00. Otherwise, use index 0. The Monday/Friday setting has higher priority than the setting of index 2.

```
schedulingData:
        { -- SET OF
            { -- SEQUENCE
                indexOrOperSpec: index: 2,
                schedule: monthly:
                { -- SEQUENCE OF Monthmask
                    { -- SET OF
                        {
                            daysOfMonth:
                            {
```

```
                        daysFromFirst: { ''B },
                        daysFromLast: { ''B }
                    }
                    timesOfDaysWps: intervalsOfDayWps: {} -- empty set
                }
            }, -- January, April, July, and October
            { -- SET OF
                {
                    daysOfMonth:
                    {
                        daysFromFirst: { ''B },
                        daysFromLast: { ''B }
                    }
                    timesOfDaysWps: intervalsOfDayWps: {} -- empty set
                }
            }, -- February, May, August, and November
            { -- SET OF
                {
                    daysOfMonth:
                    {
                        daysFromFirst: {'01'B }, -- 2nd day
                        daysFromLast: { ''B }
                    }
                    timesOfDaysWps: intervalsOfDayWps:
                    { -- SET OF
                        {
                            intervalStart: { hour: 8 },
                            intervalEnd: { hour: 0 }
                        }
                    }
                },
                {
                    daysOfMonth:
                    {
                        daysFromFirst: {'0011'B }, -- 3rd, 4th day
                        daysFromLast: { ''B }
                    }
                    timesOfDaysWps: intervalsOfDayWps:
                    { -- SET OF
                        {
                            intervalStart: { hour: 0 },
                            intervalEnd: { hour: 0 }
                        }
                    }
                },
                {
                    daysOfMonth:
                    {
                        daysFromFirst: {'00001'B }, -- 5th day
                        daysFromLast: { ''B }
                    }
                    timesOfDaysWps: intervalsOfDayWps:
                    { -- SET OF
                        {
                            intervalStart: { hour: 0 },
                            intervalEnd: { hour: 18 }
                        }
                    }
                }
            }
        }
        priority: 1
    },
    { -- SEQUENCE
        indexOrOperSpec: index: 3,
        schedule: weekly:
        { -- SEQUENCE OF WeekMaskWps
            { -- SET OF
                {
                    daysOfWeek: { '0100010'B } -- Monday and Friday
```

```
                                    timesOfDayWps: intervalsOfDayWps:
                                    { -- SET OF
                                        {
                                            intervalStart: { hour: 12 },
                                            intervalEnd: { hour: 16 }
                                        }
                                    }
                                }
                            }
                        }
                        priority: 2
                    }
                },
defaultIndex: 0
```

Annex C
MCS proforma

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

This annex is for further study.

Annex D
PICS proforma

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

This annex is for further study.

Annex E
MOCS proforma

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

This annex is for further study.

Annex F
MIDS proforma

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

This annex is for further study.

Annex G
MRCS proforma

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

This annex is for further study.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks and open system communications**
Series Y	Global information infrastructure and Internet protocol aspects
Series Z	Languages and general software aspects for telecommunication systems

Geneva, 2002