INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.739

(11/93)

DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS
OSI MANAGEMENT

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – SYSTEMS MANAGEMENT: METRIC OBJECTS AND ATTRIBUTES

ITU-T Recommendation X.739

(Previously "CCITT Recommendation")

Foreword

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. Some 179 member countries, 84 telecom operating entities, 145 scientific and industrial organizations and 38 international organizations participate in ITU-T which is the body which sets world telecommunications standards (Recommendations).

The approval of Recommendations by the Members of ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, 1993). In addition, the World Telecommunication Standardization Conference (WTSC), which meets every four years, approves Recommendations submitted to it and establishes the study programme for the following period.

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. The text of ITU-T Recommendation X.739 was approved on 16th November 1993. The identical text is also published as ISO/IEC International Standard 10164-II.

NOTE

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

CONTENTS

Summary

This Recommendation | International Standard specifies performance tools to observe characteristics of resources either directly within managed objects or through the use of metric objects. The tools also include provision of statistics, such as mean and percentile calculations, and thresholds to generate notifications.

INTERNATIONAL STANDARD

CCITT RECOMMENDATION

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – SYSTEMS MANAGEMENT: METRIC OBJECTS AND ATTRIBUTES

1 Scope

This Recommendation | International Standard defines the metric objects and attributes function. The metric objects and attributes function is a systems management function which may be used by an application process in a centralised or decentralised management environment to interact for the purpose of systems management, as defined by CCITT Rec. X.700 | ISO/IEC 7498-4. This Recommendation | International Standard defines a function which consists of generic definitions. This function is positioned in the application layer of the OSI reference model, CCITT Rec. X.200 | ISO 7498, and is defined according to the model provided by ISO 9545. The role of systems management functions is described by CCITT Rec. X.701 | ISO/IEC 10040.

This Recommendation | International Standard

- identifies the set of requirements satisfied by the function;
- provides a model for the behaviour of general metric objects;
- provides a model for the behaviour of the metric objects defined in this Recommendation | International Standard;
- specifies the management requirements of the function and how these are realised by specification of managed objects and their behaviour;
- specifies the abstract syntax of the parameters of the MAPDUs that will be used to refer to managed objects and their attributes; and
- defines managed objects.

This Recommendation | International Standard

- does not define the nature of any implementation of this Recommendation | International Standard;
- does not specify the manner in which management is to be accomplished by the user of this Recommendation | International Standard;
- does not define the nature of any interactions which result in the use of this Recommendation | International Standard;
- does not specify the services necessary for the establishment, normal and abnormal release of a management association;
- does not define the interactions which result by the simultaneous use of several management functions; and
- does not define connection establishment or authorization requirements for the use of these functions or for any associated activity.

2 Normative references

The following CCITT 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 editions of the Recommendation and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The CCITT Secretariat maintains a list of the currently valid CCITT Recommendations.

2.1 Identical Recommendations | International Standards

– CCITT Recommendation X.701 (1992) | ISO/IEC 10040:1992, *Information technology – Open Systems Interconnection – Systems management overview*.

– CCITT Recommendation X.720 (1992) | ISO/IEC 10165-1:1993, *Information technology – Open Systems Interconnection – Structure of management information: Management information model.*

– 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*.

– CCITT Recommendation X.730 (1992) | ISO/IEC 10164-1:1992, *Information technology – Open Systems Interconnection – Systems Management: Object management function.*

– CCITT Recommendation X.731 (1992) | ISO/IEC 10164-2:1992, *Information technology – Open Systems Interconnection – Systems Management: State management function.*

– CCITT Recommendation X.733 (1992) | ISO/IEC 10164-4:1992, *Information technology – Open Systems Interconnection – Systems Management: Alarm reporting function*.

2.2 Paired Recommendations | International Standards equivalent in technical content

– CCITT Recommendation X.200 (1989), *Reference model of Open Systems Interconnection for CCITT Applications*.

ISO 7498:1984, *Information processing systems – Open Systems Interconnection – Basic Reference Model.*

– CCITT Recommendation X.208 (1988), *Specification of abstract syntax notation one (ASN.1)*.

ISO/IEC 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1).*

– CCITT Recommendation X.209 (1988), *Specification of Basic Encoding Rules for abstract syntax notation*.

ISO 8825:1990, *Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*

– CCITT Recommendation X.700 (1992), *Management framework definition for Open Systems Interconnection (OSI) for CCITT applications*.

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

– CCITT Recommendation X.710 (1991), *Common management information service definition for CCITT applications*.

ISO/IEC 9595:1991, *Information technology – Open Systems Interconnection – Common management information service definition*.

– CCITT Recommendation X.711 (1991), *Common management information protocol specification for CCITT applications*.

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

2.3 Additional references

– ISO 3534-1:1993, *Statistics – Vocabulary and symbols – Part 1: Probability and general statistical terms.*

– ISO/IEC 9545:1989, *Information processing systems – Open Systems Interconnection – Application Layer structure*.

– CCITT Recommendation M.3100 (1992), *Generic Network Information Model.*

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 CCITT Rec. X.200 | ISO 7498:

a) open system;

b) OSI resource;

c) systems management.

3.2 Management framework definitions

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

a) managed object;

b) management information;

c) OSI management.

3.3 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) agent role;

c) managed system;

d) management application protocol data unit;

e) management operation;

f) manager;

g) manager role;

h) managing system;

i) notification;

j) systems management functional unit.

3.4 Definition of management information definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.721 | ISO/IEC 10165-2:

a) counter;

b) gauge;

c) threshold;

d) counter-threshold;

e) gauge-threshold.

3.5 Standard definitions of statistics

This Recommendation | International Standard makes use of the following terms defined in ISO 3534-1:

a) fractile of a probability distribution;

b) mean of a random variable;

c) variance.

3.6 CMIS definitions

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

a) attribute;

b) Common Management Information Service;

c) invoker;

d) performer.

3.7 Additional definitions

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

3.7.1 capacity: The current amount of resource (e.g octets per second) available to users, including resources already allocated for use, as well as resources available for future allocation.

3.7.2 granularity period: The time between the initiation of two successive scans.

3.7.3 metric: A value calculated from observed attribute values.

3.7.4 metric attribute: An attribute of a metric object whose value is either used as a parameter of one or more metric algorithms or whose value represents the output of such an algorithm.

3.7.5 metric algorithm: The behaviour of a metric object which models a formalized process to calculate specified results.

3.7.6 metric object: A managed object that contains at least one attribute whose value is calculated from values of attributes observed in managed objects.

3.7.7 observed attribute: An attribute of a managed object whose value is being scanned by a metric object or summarization object.

3.7.8 observed object: A managed object with one or more observed attributes.

3.7.9 percentile: The K-percentile of a probability distribution is the K/100 fractile of the distribution.

3.7.10 period synchronization time: The time to which the start of a repeating time period (e.g. granularity period) is synchronized.

3.7.11 rate: The change in a value over a specified interval of time.

NOTE – Instantaneous rate is the derivative of the value with respect to time and cannot generally be measured. The measured rate approaches the instantaneous rate as the specified interval of time approaches zero.

3.7.12 rejection rate: The rate of the number of requests rejected or the rate of the amount of requested resources but rejected due to non-available capacity.

3.7.13 resource request rate: The rate of the number of requests or the rate of the amount of resources requested.

3.7.14 resource utilization: The amount of capacity in use. Resource utilization can be measured as instantaneous or as an estimate of the mean over a time interval. Instantaneous resource utilization is the amount of capacity in use at a given point in time. Estimated mean resource utilization is an estimate of the mean resource utilization, taken over a time interval.

3.7.15 scan: A sampling process of observing attribute values at a specified point in time.

4 Abbreviations

CMIS	Common Management Information Service
EWMA	Exponentially Weighted Moving Average
GP	Granularity Period
MAPDU	Management Application Protocol Data Unit

max	maximum
min	minimum
MOCS	Managed Object Conformance Statement
MTP	Moving Time Period
PCT	A value for which a percentile is calculated
TPDU	Transport Protocol Data Unit
SMTP	Second Moving Time Period
UWMA	Uniformly Weighted Moving Average

5 Conventions

This Recommendation | International Standard does not define any services.

6 Requirements

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

– the definition of statistical monitoring tools to derive metrics to characterize performance;

– the definition of a monitoring function which provides metrics which can be used to determine the resource request rate, resource rejection rate, and resource utilization;

– the specification of mechanisms to obtain these metrics;

– the specification of notifications to be generated when these metrics exceed threshold values, and the ability to include additional performance information into those notifications;

– the specification of mechanisms to control the operation of this function;

– the ability of an external managing system to modify parameters defined and used in this monitoring function;

– the ability to model either physical capacity limitations or those limitations imposed by administrative decisions;

– the scheduling of metric monitoring over a specified interval of time; and

– the ability for a managing system to enable and disable performance measurements independent of the managed objects under observation.

7 Models

This Recommendation | International Standard provides managers of performance with tools to observe characteristics of resources either directly within observable managed objects or through the use of metric objects. The tools include methods to observe resources and to provide statistics. These tools also include thresholds to generate notifications.

7.1 The Metric object model

The metric object model defines the basic concepts of OSI metric objects and their operation in open systems.

7.1.1 Metric monitoring process

The monitoring process for the metric objects in general can be divided into the steps described in 7.1.1.1 through 7.1.1.4.

7.1.1.1 Data capture

The data necessary to compute the observed values are extracted from the observed managed objects. The observed values are monitored by sampling at intervals specified by a granularity period.

7.1.1.2 Data conversion

The extracted data are converted to the forms suitable for their intended use. The conversion algorithm may be specified by metric objects. If the conversion algorithm is absent, conversion is not performed.

Metric monitoring uses counters, gauges, or derivatives of gauges to compare measures against given criteria. The observed values may be converted from counters to gauges or from gauges to derivatives of gauges. A simple method for these two types of conversions is to take two readings of a value (e.g. a counter) over an interval of time and compute their difference (a rate). Every time this is repeated a new value of the rate is obtained. This method, often referred to as the fixed time window, is a simple way of deriving a gauge value from a counter value or computing a derivative of a gauge from a gauge value. In some cases this may be the only processing required on the observed data by the metric object.

7.1.1.3 Data enhancement

If required, the information content of the data may be enhanced by reducing the effect of random variations. Data enhancement is performed according to an arithmetic algorithm. After data enhancement, the enhanced data are visible externally in gauge attributes.

A preliminary form of analysis, data enhancement extracts the information content from the data using an arithmetic algorithm. Data enhancement techniques include smoothing to filter out the stochastic variations of the value of the monitored attribute, to find trends within the data.

7.1.1.4 Data analysis

The information is analysed and used to fulfill a performance requirement. In metric monitoring, the analysis is simple comparison of the calculated gauge values with the thresholds to trigger the generation of alarms.

Metric objects are defined to fulfill some general or specific metric monitoring objectives. The data conversion and data enhancement steps may be absent in a particular metric object. The required monitoring process is set up at the metric object creation time by setting: the observed attribute identifiers (data capture), data conversion algorithm, data enhancement algorithm, and data analysis specification. The process is controlled during the lifetime of the metric managed object by: adjusting the data enhancement algorithm (e.g. changing a moving time period), and adjusting the data analysis algorithm (e.g. adding or modifying threshold levels).

7.1.2 Requirements on metric object classes

One performance management requirement is to provide statistical measurement of the performance of resources. Metric objects have attributes representing such statistical measurements. The basic variables from which the statistics are derived can be defined by reference to attribute definitions corresponding to those variables in managed object definitions representing the resource in question. In this way, metric objects are related (statically or dynamically) to observed managed objects representing the underlying resources. The behaviour of metric objects is controlled by their attributes including a defined class of algorithm. Metric objects are defined consistent with the guidelines of CCITT Rec. X.722 | ISO/IEC 10165-4.

7.1.3 Characteristics of metric objects

The metric objects may have characteristics to provide, for example:

- the identification of the metric object;
- the identification of observed managed objects and one or more of their observed attributes;
- the identification of the metric algorithm used in the observations;
- the number of samples, frequency of observations, and the time of the last observation;
- the scheduling of the observations;
- the indication of the results of its metric algorithms;
- the threshold values at which to report alarms; and
- the management of administrative state.

A managing system may request a managed system to create a metric object. If a metric object is created, a notification indicating the managed object creation may be emitted by the metric object.

A managing system may request a managed system to delete a metric object. If a metric object is deleted, a notification indicating the managed object deletion may be emitted by the metric object.

7.1.4 Relationships of metric objects to other managed objects

The managed system uses internal mechanisms for conveying the observed attribute values to the metric object.

Metric objects have one or more relationship attributes to allow the identification of observed managed objects and their attributes to be observed. The values of each of these relationship attributes shall be read-only and are set at the time of metric object creation. However, additional attributes of a metric object may be defined to be modifiable. The relationship of a metric object with an observed managed object is a one-way asymmetric relationship.

7.1.4.1 Containment relationships

The metric object may either be contained in the observed managed object or within some other managed object.

7.1.5 Use of smoothing algorithms in metric objects

The derived gauge attribute of a metric object may be regarded as a random variable having a probability distribution that is parameterised by time. At a given time, characteristics of the probability distribution can be estimated by applying smoothing algorithms to observations of the derived gauge attribute.

Time-oriented smoothing algorithms for estimating mean, variance and percentiles are described in Annexes B and C. These algorithms are based on weighted averages of observations taken in a moving time period.

The quality of estimates given by these algorithms depends upon:

- the relative variation of the probability distribution during the moving time period;
- the number of observations during the moving time period; and
- the independence of observations.

Optimal values for the moving time period and the number of observations within that period depend upon the characteristics of the probability distribution of the derived gauge attribute.

7.2 Supporting metric objects

This clause describes the metric objects which are defined by this Recommendation | International Standard. These managed objects are the monitor metric, mean monitor, algorithm indicating mean monitor, moving average mean monitor, mean and variance monitor, mean and percentile monitor, and mean and min max monitor metric objects.

The monitor metric managed object is used to observe a counter or a gauge type attribute and update the derived gauge attribute after each observation. If the observed attribute is a counter, the monitor metric managed object may derive a gauge value from the counter. In this case the derived gauge value is the difference between successive observations of the counter. If the observed attribute is a gauge the derived gauge value will be equal to the observed attribute at the time of each observation. The monitor metric managed object’s activity can be scheduled. The monitor metric managed object may have a severity indicating gauge-threshold applied to the derived gauge attribute. The managed object may emit a quality of service alarm notification whenever the threshold value is crossed. The notification may also indicate the specific problem related to the observed attribute (e.g. rejection rate).

The mean monitor managed object class is derived from the monitor metric managed object class. In addition to the characteristics of the monitor metric managed object, the mean monitor managed object has an estimate of mean attribute used to provide the estimate of the mean of the derived gauge attribute value. The severity indicating gauge-threshold value in this managed object is applied to the estimate of the mean attribute. If enabled the managed object will emit a quality of service alarm notification whenever the threshold value is crossed. The notification may also indicate the specific problem related to the observed attribute (e.g. rejection rate).

The algorithm indicating mean monitor managed object class is derived from the mean monitor managed object class. In addition to the characteristics of the mean monitor managed object, the algorithm indicating mean monitor managed object provides the identifier of an algorithm used to derive the estimate of the mean.

The moving average mean monitor managed object class is derived from the mean monitor managed object class. In addition to the characteristics of the mean monitor managed object, the moving average mean monitor managed object specifies the EWMA algorithm used to estimate the mean value.

The mean and variance monitor managed object provides estimates of the mean and the variance of an observed attribute. This managed object class is derived from the moving average mean monitor managed object class. In addition to the characteristics of the moving average mean monitor managed object, the mean and variance monitor managed object has an attribute needed for the algorithm used to evaluate the variance.

The mean and percentile monitor managed object provides estimates of the mean, median, nth percentile, largest in a replication and the smallest in a replication values of an observed attribute. This managed object class is derived from moving average mean monitor managed object class. In addition to the characteristics of the moving average mean monitor managed object, the mean and percentile monitor managed object has attributes needed for the algorithm used to evaluate the percentile.

The mean and min max monitor managed object provides estimates of the mean, as well as the smallest and largest values of an observed attribute. The managed object class is derived from the mean monitor managed object class. In addition to the characteristics of the moving average mean monitor managed object, the mean and min max monitor managed object has attributes needed to provide the largest and smallest values.

7.3 Severity indicating gauge-threshold model

This Recommendation | International Standard defines a severity indicating gauge-threshold attribute type which is used to trigger notifications for monitoring resources. This attribute type has behaviour similar to that of a gauge-threshold attribute type, as defined in 9.3.2 of CCITT Rec. X.721 | ISO/IEC 10165-2. The syntax of the severity indicating gauge-threshold is that of the gauge-threshold enhanced to associate an optional severity indication parameter with each of the notify-high and notify-low submembers of each threshold level member.

If the severity indication parameter is present, its value is used in the perceived severity parameter of all alarms associated with that threshold's submember.

An optional mechanism is provided for determining the value of the specific problems parameter of emitted quality of service alarm notifications.

The example in Figure 1 illustrates three threshold levels. Threshold levels 1 and 2 are triggered in the increasing direction while threshold level 3 is triggered when the gauge value is decreasing. The emission of the notifications associated with the severity indicating gauge-threshold is as follows:

- initially, if notify-high's on/off switch is true and the gauge value becomes equal to or greater than threshold level 1, in a positive going direction, then the defined event notification is triggered; subsequent crossings of notify-high's gauge value shall not cause further generation of event notifications unless the gauge value becomes equal to or less than threshold level 1-clear value.

- initially if notify-low's on/off switch is true and the gauge value becomes equal to or less than threshold level 1-clear, in a negative going direction and the gauge value has been greater than or equal to threshold level 1, then the defined event notification is triggered if there is an outstanding event notification for threshold level 1; subsequent crossings of notify-low's gauge value shall not cause further generation of event notifications unless the gauge value becomes equal to or greater than threshold level 1 value.

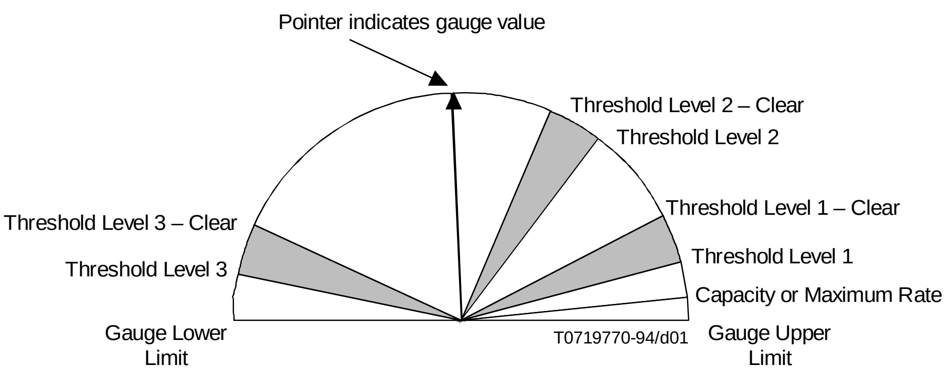


Figure 1 – Example severity indicating gauge-threshold

If the notification switch for threshold level 3 is set to "true" a notification will be emitted as follows:

– initially, if notify-low's on/off switch is true and the gauge value becomes equal to or less than threshold level 3, in a negative going direction, then the defined event notification is triggered; subsequent crossings of notify-low's gauge value shall not cause further generation of event notifications unless the gauge value becomes equal to or greater than threshold level 3-clear value.

– initially, if notify-high's on/off switch is true and the gauge value becomes equal to or greater than threshold level 3-clear, in a positive going direction and the gauge value has been less than or equal to threshold level 3, then the defined event notification is triggered; subsequent crossings of notify-high's gauge value shall not cause further generation of event notifications unless the gauge value becomes equal to or less than threshold level 3 value.

7.4 Metric objects and models for workload monitoring

This Recommendation | International Standard provides for resource monitoring. Monitoring of resources includes monitoring of resource utilization, rejection rate, and resource request rate. Additionally, this Recommendation | International Standard provides for setting of threshold values, reporting of early warning and severe warning conditions, and the clearing of those conditions.

NOTE 1 – The unit for utilization and capacity can be the amount of resources in use or a rate derived from the cumulative amount of resources used.

Three models are defined to fulfill OSI workload monitoring requirements:

– the resource utilization model provides monitoring of the amount of resource in use;

– the rejection rate model provides monitoring of service request rejection rate; and

– the resource request rate model provides monitoring of requests for usage of resources.

In some cases, it is possible that the service requests are rejected before the capacity is fully utilized. For example, this happens when the amount of resource requested exceeds the remaining available capacity.

Systems can support one or more models to fulfill their management requirements. When more than one model is supported there are relationships among them. These relationships exist because each model is associated with the use of the same resource. The resource utilization model indicates the variation of usage of the resource. When the request cannot be fulfilled, the rejection rate model provides information concerning rejection rate conditions. The resource request rate model provides information concerning request rates.

Users of this management service must be aware that capacity could decrease (e.g. because of a system fault).

The relationship between the resource utilization, rejection rate, and resource request rate of the same (hypothetical) resource is illustrated in Figure 2. Together, these three quantities provide an estimate of the workload for that resource.

Each of these quantities is modelled as a gauge. A severity indicating gauge-threshold, illustrated in the figure below, is defined in Annex A. Each gauge has a severity indicating gauge-threshold, which is a set of threshold levels. Each threshold level contains a pair of (high and low) threshold values.

For example, threshold level 1 in Figure 2 may be associated with a more severe condition than threshold level 2, which may be regarded as an early warning. Then, each of the high values is used to trigger a notification indicating the corresponding condition and the low values are used to trigger a notification indicating the clearing of the condition.

Although not shown in Figure 2, there may be instances when the manager needs to be aware of the under-utilization of resources, possibly to allocate them for service for other use. In such a case, the manager may be made aware by a downward triggered threshold level event notification.

In addition to resource utilization gauge-thresholds, similar gauge-thresholds are defined for the rejection rate and resource request rate. These operate as described for the resource utilization gauge-threshold, although, in general, only the increasing gauge-thresholds would be required.

To support resource utilization, resource request rate and rejection rate monitoring using metric objects the observed managed object must provide gauge or counter attributes for deriving the resource utilization, the resource request rate and the rejection rate respectively.

NOTE 2 – There are several possible ways to handle a user request if the request cannot be fully satisfied:

a) the request is totally rejected;

b) the request is partially satisfied;

c) the request is queued and will eventually be satisfied;

d) the request is queued for a maximum amount of waiting time.

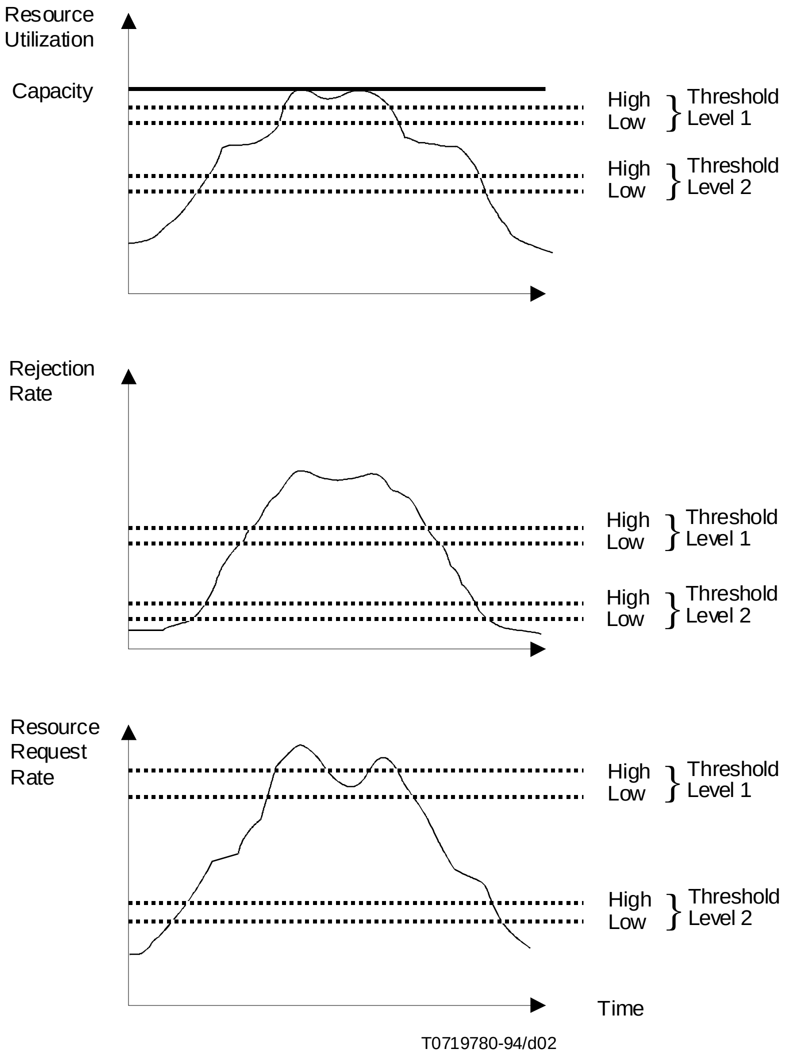


Figure 2 – Resource utilization, rejection rate and resource request rate

In the first case the semantics of resource request rate and rejection rate are obvious and no further explanation is needed. However in all other cases the precise semantics of resource request rate and rejection rate is dependent on the behaviour of the observed managed object.

Gauges can be used to model the values of resource utilization, rejection rate, and resource request rate. A generic model of such a gauge is illustrated in Figure 1.

7.4.1 Resource utilization gauge model

In addition to the common characteristics specified in 7.4, the resource utilization gauge has the following specific characteristics:

a) The value of the gauge indicates the resource utilization which is the result of calculations over a time interval that may be specified by the manager.

b) The utilization will tend to increase as resources are supplied to users.

c) The utilization will tend to decrease as resources are released by users.

d) There are smoothing algorithms for calculating the value of estimated means. These algorithms are defined in Annexes B and C. If the mean monitor managed object class or any of its subclasses are used, an algorithm is selected for calculating the value of the estimated mean utilization. If the data enhancement process is not required, the monitor metric managed object class can be used.

e) Resource utilization is measured as an estimate of the mean utilization over a time interval. Estimated mean resource utilization is an estimate of the mean amount of either services supplied or workload (selection depends on units of capacity) utilization.

NOTE – By using the appropriate metric object class, resource utilization can be reported as either instantaneous utilization values or an estimate of the mean utilization over a time interval.

7.4.2 Rejection rate gauge model

In addition to the common characteristics specified in 7.4, the rejection rate gauge has the following characteristics:

a) The value of the gauge indicates the rejection rate which is the result of calculations over a set of time intervals that may be specified by the manager.

b) The rejection rate will tend to increase as user requests for resources are denied and will tend to decrease as user requests are granted or users submit less requests for resources.

c) There are associated algorithms for calculating the value of the rate. The algorithms are defined in Annexes B and C.

d) Rejection rate is measured as an estimate of the mean rejection rate over a time interval that may be specified by the manager. Estimated mean rejection rate is an estimate of the mean amount of either service requests or workload (selection depends on units of capacity) rejected per unit time due to non-available capacity, taken over a time interval that may be specified by the manager.

7.4.3 Resource request rate gauge model

In addition to the common characteristics specified in 7.4, the resource request rate gauge has the following characteristics:

a) The value of the gauge indicates the resource request rate which is the result of calculations over a set of time intervals that may be specified by the manager.

b) The resource request rate will tend to increase as user requests for resources arrive and will tend to decrease as users submit less requests for resources.

c) There are associated algorithms for calculating the value of the rate. The algorithms are defined in Annexes B and C.

d) Resource request rate is measured as an estimate of the mean resource request rate over a time interval that is specified by the manager. Estimated mean resource request rate is an estimate of the mean amount of either service requests or workload (selection depends on units of capacity) requested per unit time taken over a time interval that is specified by the manager.

7.5 Use of metric objects for workload monitoring

The monitor metric managed object or any of its subclasses may be used for generating notifications related to resource utilization when it is desired to apply a threshold directly to the derived gauge representing resource utilization. The monitor metric managed object may also be used for generating notifications related to resource request rate and rejection rate when it is desired to apply a threshold to a gauge derived from a counter that counts requests or rejections for the resource. If a counter difference package is included in the monitor metric managed object, the underlying gauge is treated as a counter.

NOTE – The unit for utilization and capacity can be the amount of resources in use or a rate derived from the cumulative amount of resources used.

The mean monitor managed object class or its subclasses may be used for generating notifications related to the average value of the derived gauge derived over time for resource utilization, resource request rate or rejection rate. If the resource request rate and rejection rate are to be calculated from a counter that counts requests or rejections for the resource, or if the resource utilization is derived from a cumulative usage counter, then a counter difference package shall be included in the instances of the mean monitor managed object class or its subclasses.

The moving average mean monitor managed object may also be used for workload monitoring. When the moving average mean monitor managed object is used for workload monitoring it is possible to obtain information about and manage the algorithm used in measurements.

Other managed objects may generate notifications related to resource utilization, resource request rate or rejection rate if their definition includes the appropriate generic definitions defined in this Recommendation | International Standard.

8 Generic definitions

8.1 Managed objects

The managed object classes defined in this Recommendation | International Standard are, with the exception of scanner, metric objects whose inheritance structure is shown in Figure 3.

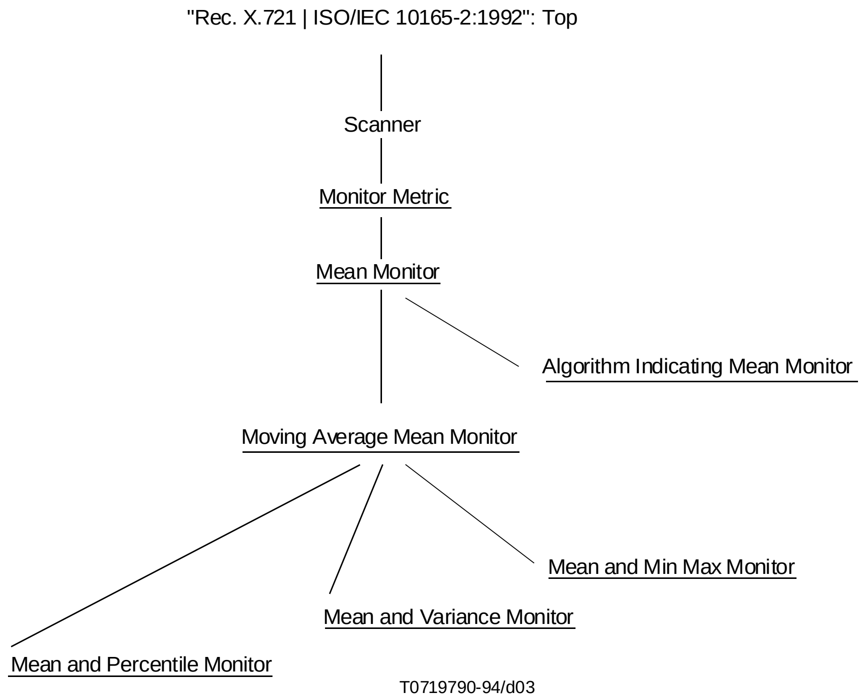


Figure 3 – Inheritance structure of metric objects

Note – Instantiable managed objects are underlined.

To report threshold crossing events metric objects use the Alarm reporting service defined in CCITT Rec. X.733 | ISO/IEC 10164-4.

At least one threshold level shall be specified on those metric objects which have severity indicating gauge-threshold attributes. If only one threshold level is specified, it shall represent a severe threshold level as defined for workload monitoring.

Attribute values of managed objects can be read or modified in accordance with the operations described in CCITT Rec. X.720 | ISO/IEC 10165-1.

8.1.1 Scanner

8.1.1.1 Overview

The scanner managed object class is a non-instantiable superclass from which other managed objects (e.g. metric objects) are derived. It defines the facilities for periodically sampling the values of a specified set of attributes within specified managed objects. The intervals during which the periodic scans may occur can be controlled according to a schedule.

8.1.1.2 Attributes and packages of the scanner

The scanner managed object class has the following attributes:

a) *Scanner Id*

This attribute contains a value used to identify an instance of the scanner managed object class (i.e. the scanner Id is used for naming).

b) *Operational state*

This attribute is defined in CCITT Rec. X.731 | ISO/IEC 10164-2.

c) *Administrative state*

This attribute is defined in CCITT Rec. X.731 | ISO/IEC 10164-2.

d) *Granularity period*

This attribute contains the granularity period indicating the time between scans.

The scanner managed object class has the following conditional packages:

- availability status package (present if the scanner can be scheduled);
- duration package (present if the managed object function is scheduled to start at a specified time and stop at either a specified time or function continuously);
- daily scheduling package (present if both the weekly scheduling package and external scheduler packages are not present in an instance and daily scheduling is required);
- weekly scheduling package (present if both the daily scheduling package and external scheduler packages are not present in an instance and weekly scheduling is required);
- external scheduling package (present if both the daily scheduling package and weekly scheduling packages are not present in an instance and a reference to an external scheduler is required);
- period synchronization package (present if clock synchronization for the granularity period is required. If this package is not present then clock synchronization is a local matter.);
- create delete notifications package (present if notification of managed object creation and deletion events is required);
- attribute value change notification package (present if notification of attribute value change events is required); and
- state change notification package (present if notification of state change events is required).

NOTES

1 The first five packages are defined in CCITT Rec. X.721 | ISO/IEC 10165-2.

2 The period synchronization package is defined in 8.1.9.9.

3 The last three packages are defined in CCITT Recommendation M.3100.

8.1.1.3 Behaviour of the scanner

A managed object of this class represents the ability to retrieve values of attributes of managed objects and produce summary information from those values. This summary information may be made available in attributes, notifications, action replies, or some combination of these. Summary information may consist of observed attribute values or statistics calculated from these values (either over time or over managed objects).

Observed attribute values are retrieved during a “scan”, which is initiated periodically, at the end of each granularity period, provided that the granularity period is non-zero.

The granularity period attribute indicates the length of the granularity period. The granularity period in the scanner managed object class shall not be modified unless the value of the administrative state is “locked”. If the period synchronization package is not present, the time at which a granularity period starts after the scanner is unlocked is a local matter.

The administrative state attribute is used to suspend or resume the scanning function. If administrative state has the value “unlocked”, the scanner is administratively permitted to perform scans. If administrative state has the value “locked”, the scanner is administratively prohibited from performing scans.

The operational state attribute represents the operational capability of the scanner to perform its functions.

If present, the availability status attribute indicates whether the schedule inhibits the scanning function, consistent with the description of the availability status attribute in CCITT Rec. X.731 | ISO/IEC 10164-2.

The scanning function can be suspended or resumed according to a schedule. If the schedule suspends the scanning function, any scan that is in progress continues.

If the period synchronization package is present, the period synchronization time attribute, contained in this package, is used to determine the time to which the granularity period is synchronized. The period synchronization time attribute shall not be modified unless the value of the administrative state is locked. The start of each granularity period is an integral number of granularity periods before or after the time specified by this attribute.

If the attribute value change notification package is present, then changes to the granularity period or to the period synchronization time shall cause attribute value change notifications to be emitted. If the state change notification package is present, then changes to the operational state or administrative state shall cause state change notifications to be emitted.

NOTES

1 It is assumed that the observed attributes can be scanned within the granularity period. If a scan is initiated when another scan is in progress, then how the conflict is resolved is a local issue.

2 The scan may occur at slightly different times for each observed attribute, but the time skew between the start of a scan and the scan time for a given attribute in the scan should be approximately equal between successive scans.

8.1.2 Monitor metric

8.1.2.1 Overview

The monitor metric managed object class is defined as a subclass of the scanner managed object class. The monitor metric managed object monitors values of an attribute in an observed managed object at intervals specified by the granularity period. A gauge value is derived from the values of the observed attribute.

8.1.2.2 Attributes and packages of the monitor metric

The monitor metric managed object class has the following attributes:

a) *Observed object instance*

   This attribute is used to identify the instance of the managed object that contains the observed attribute. Its value shall be specified when the monitor metric managed object is created and may not be modified.

b) *Observed attribute identifier*

   This attribute is used to identify the observed attribute of the observed managed object. Its value shall be specified when the monitor metric managed object is created and may not be modified.

c) *Derived gauge*

   This attribute contains the gauge value derived from the values of the observed attribute. If there is neither a counter difference nor a gauge difference package present, the derived gauge attribute value contains the last observed value of the observed attribute. If either of the difference packages is present, the derived gauge value contains the value of the difference between two successive observations of the observed attribute value.

The monitor metric managed object class has the following conditional packages:

– counter difference package (present if counter to gauge conversion is required and the gauge difference package is not present);

– counter overflow package (present if the counter difference package is present and modulo arithmetic is required to calculate the new value of the derived gauge on counter overflow);

– gauge difference package (present if gauge to derivative conversion is required and the counter difference package is not present);

– derived gauge-threshold package (present if a threshold on the derived gauge is required);

– specific problems indication (present if specific problems indication behaviour is required and the derived gauge-threshold package is present); and

– derived gauge timestamp package (present if timestamping associated with updating the derived gauge is required).

Further description of these packages is contained in 8.1.9.

8.1.2.3 Behaviour of the monitor metric

The monitor metric managed object monitors values of an attribute in an observed managed object. The observed attribute is monitored at intervals specified by the granularity period. A gauge value (derived gauge) is derived from the values of the observed attribute. The value of the derived gauge attribute shall remain the same until the next observation.

The presence or absence of the difference packages (i.e. counter difference and gauge difference) determines how the output of the data conversion process is derived from the observed attribute. If neither package is present, the derived gauge attribute value contains the last observed value of the observed attribute. If either of the difference packages is present, the derived gauge value contains the value of the difference between two successive observations of observed attribute value.

If a difference package is not present, the units of the derived gauge are the same as the units of the observed attribute.

If a difference package is present, the units of the derived gauge is related to the interval between observations (granularity period). For example, if the derived gauge has a value of 60 and the observed attribute is a counter of the number of messages, with the granularity period at 15 minutes, the derived gauge represents a value of 60 messages per 15 minutes.

The presence of the derived gauge timestamp package requires monitor metric instances to update the derived gauge timestamp attribute with the current time, whenever the derived gauge attribute value is updated.

If the derived gauge-threshold package is present, the severity indicating gauge-threshold in the package is applied to the derived gauge attribute, and a threshold crossed quality of service alarm notification will be emitted when the derived gauge value crosses the threshold values. In addition, the presence of the derived gauge timestamp package requires managed object instances to report the derived gauge timestamp attribute as part of the monitored attributes parameter of any quality of service alarm notification.

The quality of service alarm notification parameters shall be as specified in 8.2.1. The value of the derived gauge attribute shall be in the observed value element of the threshold information parameter of the quality of service alarm notification. The values of the observed object instance and observed attribute identifier shall be in the monitored attributes parameter of the quality of service alarm notification.

If the conditional specific problems indication package is present, the specific problems indicator attribute value (used to specify a specific type of metric monitoring) shall be placed into the specific problems parameter of the threshold crossed quality of service alarm notification.

The observed attribute is specified by the observed attribute identifier of the managed object identified by observed object instance. If the counter difference package is absent, the observed attribute is treated as a gauge. If the gauge difference package is present, the observed attribute is treated as a gauge.

When the monitor metric managed object is created, the following shall be specified:

- the scanner Id, observed object instance, observed attribute identifier;
- the granularity period (i.e. time between observations of the observed attribute);
- if the counter difference package is present, then the data conversion process is calculated according to the counter difference package behaviour;
- if the counter overflow package is present, then the metric object shall ensure that the value of the modulus value attribute is initialized before entering the on-duty condition.;
- if the gauge difference package is present, then the data conversion process is calculated according to the gauge difference package behaviour;
- a choice of at most one of the available scheduling packages. The scheduling packages are defined in CCITT Rec. X.721 | ISO/IEC 10165-2;
- if the derived gauge-threshold package is present, then the severity indicating gauge-threshold (i.e. the threshold levels to be applied to the derived gauge attribute value for generation of quality of service alarm notifications) shall be specified; and
- if the observed attribute identifier references an attribute that is not of real or integer type, the creation shall fail and the CMIS error “invalid attribute value” shall be returned.

If the state change notification package is present, then changes to the administrative state, operational state, and availability status shall cause state change notifications to be emitted.

The lifetime of the metric object may be controlled by the managing system by requesting the managed system to delete the metric object. If the managed system deletes a metric object and the create delete notifications package is present, a notification indicating managed object deletion is emitted. The notification shall include the values of the attributes of the metric object at the time of deletion.

Attributes of the counter difference, counter overflow, gauge difference, derived gauge-threshold, and specific problems indication packages may only be modified if the administrative state is locked. If the attribute value change notification package is present, then modifications of these attributes shall result in an attribute value change notification.

8.1.3 Mean monitor

8.1.3.1 Overview

The mean monitor managed object class is defined as a subclass of the monitor metric managed object class. The mean monitor managed object provides the estimate of the mean of the derived gauge attribute. The severity indicating gauge-threshold is applied to the estimate of the mean attribute for the purpose of generating quality of service alarm notifications.

8.1.3.2 Attributes and packages of the mean monitor

The mean monitor managed object class has the following attributes:

a) *Estimate of mean*

This attribute contains the estimate of the mean calculated by the algorithm. The initial value of this attribute shall be supplied upon creation of the metric object for use in initialising the algorithm. The attribute may be modified after creation of the metric object in order to re-initialise the algorithm.

b) *Moving time period*

This attribute contains the effective time interval over which values are scanned to calculate an estimate of the mean. This attribute shall be initialised when the metric object is created and may be modified.

NOTE – The value of this attribute should be greater than or equal to the value of the granularity period attribute.

The mean monitor managed object class has the following conditional packages:

- estimate of mean threshold package (present if threshold on estimate of mean is required);
- specific problems indication package (present if specific problems indication behaviour is required and either the estimate of mean threshold package is present or the derived gauge-threshold package is present).

Further description of these packages is contained in 8.1.9.

8.1.3.3 Behaviour of the mean monitor

The estimate of mean attribute has the same units as the derived gauge attribute.

The estimate of the mean of the derived gauge attribute is provided by the estimate of mean attribute. The algorithm used to evaluate the estimate of the mean value is not specified. However, the time period that is used in the evaluation of the mean value is specified by the moving time period attribute.

NOTE – Examples of two algorithms that may be used to do this are EWMA and UWMA as specified in Annexes B and C.

There are two conditional packages used to apply thresholds to the value of derived gauge and estimate of mean attribute values. If the derived gauge-threshold package (inherited from the monitor metric managed object class) is present, the severity indicating gauge-threshold is applied to the value of the derived gauge attribute. If the estimate of the mean threshold package is present, the estimate of mean severity indicating gauge-threshold is applied to the value of the estimate of mean attribute.

The managed object will emit a quality of service alarm notification whenever the value of the derived gauge or the estimate of the mean attribute triggers the corresponding threshold.

If either of the threshold packages is present, then the conditional specific problems indication package may also be present. If the specific problems indication package is present then the specific problems indicator attribute value shall be placed in the specific problems parameter of the quality of service notifications that are emitted, whether they are triggered by either the derived gauge attribute or by the estimate of mean attribute.

When the mean monitor metric object is created, the moving time period required by the mean estimation algorithm shall be specified.

When the mean monitor metric object is created, the estimate of mean severity indicating gauge-threshold levels (i.e. the threshold levels to be applied to the estimate of mean attribute value for generation of quality of service alarm notifications) may be specified.

In addition, the presence of the derived gauge timestamp package requires metric object instances to report the derived gauge timestamp attribute as part of the monitored attributes parameter of the quality of service alarm notification.

The use of the quality of service alarm notification parameters is given in 8.2.1.

The value of the estimate of mean attribute will be in the observed value element of the threshold information parameter of the quality of service alarm notification.

The values of the observed object instance and observed attribute identifier will be in the monitored attributes parameter of the quality of service alarm notification.

The moving time period attribute, estimate of the mean attribute and attributes of the estimate of mean threshold and specific problems indication packages may only be modified if the administrative state is locked. If the attribute value change notification package is present, modifications of these attributes shall result in an attribute value change notification.

8.1.4 Algorithm indicating mean monitor

8.1.4.1 Overview

The algorithm indicating mean monitor managed object class is a subclass of the mean monitor managed object class. The algorithm indicating mean monitor managed object class provides an indication of the algorithm used to calculate the estimate of the mean.

8.1.4.2 Attributes and packages of the algorithm indicating mean monitor

The algorithm indicating mean monitor has the following attribute:

– *Algorithm identifier*

  This attribute contains the identification of the algorithm used to derive the estimate of the mean. The value of this attribute is provided by the managed system at the time of object creation and is read-only. This attribute may not be specified by the manager on a create request.

  This Recommendation | International Standard registers two algorithms for use in this attribute in . Other identifiers for use in this attribute may be defined outside this Recommendation | International Standard and registered using the procedures defined for ASN.1 object identifier values in CCITT Rec. X.208 | ISO 8824.

8.1.4.3 Behaviour of the algorithm indicating mean monitor

This managed object class has the same behaviour as the mean monitor managed object class. In addition, it provides the identifier of the algorithm used to derive the value of the estimate of the mean.

8.1.5 Moving average mean monitor

8.1.5.1 Overview

The moving average mean monitor managed object class is a subclass of the mean monitor managed object class. The EWMA algorithm is used for smoothing of the derived gauge value.

8.1.5.2 Attributes and packages of the moving average mean monitor

No new attributes or packages have been added.

8.1.5.3 Behaviour of the moving average mean monitor

An estimate of the mean of the derived gauge attribute is calculated using the exponentially weighted moving average (EWMA) algorithm as defined in B.2.2 and its value is placed into the estimate of mean attribute. The value of the estimate of mean remains the same until the next observation.

When the moving average mean monitor metric object is created, the initial value for the estimate of mean shall be specified.

8.1.6 Mean and variance monitor

8.1.6.1 Overview

The mean and variance monitor managed object is a subclass of the moving average mean monitor managed object class. The mean and variance monitor managed object provides estimates of the mean and variance of the derived gauge attribute.

8.1.6.2 Attributes and packages of the mean and variance monitor

The mean and variance monitor managed object class has the following attributes:

a) *Second moving time period*

This attribute is used for calculating the variance using the EWMA algorithm. It represents the effective time interval over which values are scanned to calculate an estimate of the variance. (It is used in the calculation of "g" for variance as described in B.2.1 and B.2.3.) This attribute shall be initialised when the metric object is created and may be modified.

NOTE – The value of this attribute should be greater than or equal to the value of the granularity period attribute.

b) *Estimate of variance*

This attribute contains the calculated estimate of variance.

The mean and variance monitor managed object class has one mandatory package and no conditional packages.

8.1.6.3 Behaviour of the mean and variance monitor

This managed object calculates estimates of the mean and variance on the values of the derived gauge.

The estimated mean of the derived gauge (in estimate of mean attribute inherited from the mean monitor), and variance (estimate of variance) are calculated using the exponentially weighted moving average (EWMA) algorithm. (See B.2.2 and B.2.3.)

The estimate of variance attribute is placed in the monitored attributes parameter of the quality of service alarm notification.

When the mean and variance monitor metric object is created, the following shall be specified:

– The moving time period and second moving time period required by the EWMA algorithm;

– The EWMA gauge mean and EWMA gauge variance behaviours are included and the initial value for the estimate of mean and estimate of variance shall be specified.

The second moving time period attribute may only be modified if the administrative state is locked. If the attribute value change notification package is present, modification of this attribute shall result in an attribute value change notification.

8.1.7 Mean and percentile monitor

8.1.7.1 Overview

The mean and percentile monitor managed object class is a subclass of the moving average mean monitor managed object class. The mean and percentile monitor managed object provides estimates of the mean, median, PCTth percentile, (100-PCTth) percentile, largest in a replication, and smallest in a replication values of either a gauge type attribute, or a gauge derived from a counter type attribute. PCT used in the PCTth percentile is a positive integer from 1 to 49 inclusive.

8.1.7.2 Attributes and packages of the mean and percentile monitor

The mean and percentile monitor managed object class has the following attributes:

a) *Second moving time period*

This attribute is used in calculating the time constant for calculating the percentiles used in the EWMA. It represents the effective time interval over which values are scanned to calculate an estimate of the percentile. (It is used in the calculation of "h" in the calculations for percentiles as described in B.2.1 and B.2.4.) This attribute shall be initialised when the metric object is created and may be modified.

NOTE – The value of this attribute should be greater than or equal to the value of the granularity period attribute.

b) *Estimate of largest in replication*

This attribute contains the calculated estimate of the largest value in a replication of size equal to the value of the number of replications attribute.

c) *Estimate of smallest in replication*

This attribute contains the calculated estimate of the smallest value in a replication of size equal to the value of the number of replications attribute.

d) *Estimate of median*

This attribute contains the calculated estimate of the median value.

e) *Estimate of 100-PCT percentile*

This attribute contains the calculated estimate of the one hundred minus PCTth percentile.

f) *Estimate of PCT percentile*

This attribute contains the calculated estimate of the PCTth percentile.

g) *Number of Replications*

This attribute contains the number of repeated statistically independent samples (replications) to be used for the calculations. The number of replications is represented by "M" in equations defining percentile calculations. See Annex B.

The mean and percentile monitor metric object has the following conditional package:

– configurable percentile package, as defined in 8.1.9.8 (present if configurable percentile calculation is required.)

8.1.7.3 Behaviour of the mean and percentile monitor

This metric object calculates the estimate of the mean and percentiles of the values of the derived gauge.

PCT is an integer with a value from 1 to 49 inclusive. If the configurable percentile package is not present, PCT shall be 25, and the number of replications attribute shall be set by the manager to 8, 16 or 32.

The mean of the derived gauge (in the estimate of mean attribute inherited from the mean monitor), estimate of median, estimate of largest in a replication, estimate of smallest in a replication, estimate of 100-PCTth percentile, and estimate of the PCTth percentile are calculated using the exponentially weighted moving average (EWMA) algorithm as described in B.2.2 and B.2.4.

The percentile estimates are only updated after a number of replications are observed, with one replication per granularity period. The estimate of mean is updated and the thresholds (if present) are applied at each granularity period.

When the mean and percentile monitor metric object is created, the following shall be specified:

– The moving time period (MTP) and second moving time period (SMTP) required by the EWMA algorithms;

– The EWMA gauge mean and EWMA gauge percentile behaviours are included and the initial value for the estimate of mean, estimate of largest in a replication, estimate of smallest in a replication, estimate of median, estimate of 100-PCT percentile, estimate of PCT percentile, number of replications, and time between replications attributes shall be specified.

Estimate of largest in replication, estimate of smallest in replication, estimate of median, estimate of 100-PCT percentile, estimate of PCT percentile, number of replications, and granularity period attribute values are placed in the monitored attributes parameter of the quality of service alarm notification.

The second moving time period attribute and attributes of the configurable percentile package may only be modified if the administrative state is locked. If the attribute value change notification package is present, modifications of these attributes shall result in an attribute value change notification.

8.1.8 Mean and min max monitor

8.1.8.1 Overview

The mean and min max monitor managed object is a subclass of the moving average mean monitor managed object. In addition to providing estimates of the mean, it also includes the capability of providing the minimum value of the derived gauge attribute and the maximum value of the derived gauge attribute.

8.1.8.2 Attributes and packages of the mean and min max monitor

The mean and min max monitor managed object class has the following attributes:

a) *Estimate of largest*

This attribute contains the maximum value of the observed gauge.

b) *Estimate of smallest*

This attribute contains the minimum value of the observed gauge.

The mean and min max monitor metric object has no conditional packages.

8.1.8.3 Behaviour of the mean and min max monitor

The mean and min max monitor managed object inherits the properties of the moving average mean monitor managed object class.

The mean and min max monitor managed object calculates estimates of the mean, largest, and smallest values of the derived gauge.

The initial value for the estimate of largest shall be set equal to the initial value of the estimate of the mean. The initial value for the estimate of the smallest is also set equal to the initial value for the estimate of the mean.

For each granularity period, if the derived gauge value is less than the current estimate of the smallest, then the estimate of the smallest is set equal to this new value. If the derived gauge value is greater than the current estimate of the largest, then the estimate of the largest is set equal to this new value.

The estimate of largest and estimate of smallest attribute values are placed in the monitored attributes parameter of the quality of service alarm notification.

8.1.9 Packages supporting metric objects

The conditional packages which are present are determined at the time of managed object creation, and they are used to control the behaviour of metric object instances.

8.1.9.1 Counter difference package

8.1.9.1.1 Overview

This package defines the behaviour for deriving a gauge value from an observed attribute that is a wrap counter (i.e. a counter as defined in CCITT Rec. X.721 | ISO/IEC 10165-2). The gauge value derived is the difference between the observed counter values for two successive observations.

NOTE – It is recommended that a gauge not be derived from a resettable counter which may be reset frequently. The resetting of a counter during a granularity period will result in an invalid gauge value for that granularity period. Therefore, gauges should only be derived from resettable counters if the time between resets is significantly greater than the granularity period.

8.1.9.1.2 Attributes of the counter difference package

The counter difference package contains the following attributes:

a) *Previous scan counter value*

This attribute is used in calculating counter differences for counters. It contains the value of the counter captured by the latest observation.

b) *Procedural status*

This attribute is defined in CCITT Rec. X.731 | ISO/IEC 10164-2. It is used to convey the initialization status of the counter difference package. An empty value indicates that the counter difference package is fully initialized.

8.1.9.1.3 Behaviour of the counter difference package

If this package is included in a managed object, the observed attribute shall be a counter. The gauge value derived is initially calculated using the following equation:

$$V[t] = [counter[t] - counter[t\text{-}GP]]$$

where

V[t]	is the difference between successive observations of the counter;
counter[t]	is the value of the counter at the current time t (not saved in any attribute);
counter[t-GP]	is the previous value of the counter at time t-GP (i.e. previous scan counter value);
GP	is the sampling interval in seconds, minutes, hours, or days (i.e. granularity period).

The initial value of previous scan counter value determines the initial difference value.

The value of the previous scan counter value attribute is considered valid if it was obtained at the end of the previous granularity period. Thus, for example, it is not considered valid in the first scan interval subsequent to creation, in the first scan interval subsequent to transition from “unlocked” to “locked” states or subsequent to transition from “off-duty” to “on-duty” states.

During the period in which the value of the previous scan counter value attribute is not valid, the value of the procedural status attribute shall have the value “Not initialized”. During this period, the value of the derived gauge attribute is not usable as input to subsequent data enhancement or data analysis algorithms, and any attempt to read either the derived gauge attribute or the previous scan counter value attribute will result in the retrieval of invalid information.

If the value of the previous scan counter value attribute is valid, and

– if V[t] (i.e. the difference between successive observations) is not negative then the value of derived gauge is set to V[t];

– if V[t] is negative and the counter overflow package (see 8.1.9.2) is present in the managed object then the derived gauge value is calculated according to the behaviour specified in 8.1.9.2.3;

– if V[t] is negative and the counter overflow package is not present in the managed object, then the derived gauge is set to V[t]. This negative value of the derived gauge attribute is not usable as input to subsequent data enhancement or data analysis algorithms.

The observed value is retained by the metric object to be used in calculating the next difference, i.e. it becomes the value of the previous scan counter value attribute.

During the period after which the previous scan counter value attribute is initialized but before the end of next granularity period, the value of the procedural status attribute shall have the value “Initializing”. During this period, the value of the derived gauge attribute is not usable as input to subsequent data enhancement or data analysis algorithms, and any attempt to read the derived gauge attribute will result in the retrieval of invalid information.

8.1.9.2 Counter overflow package

8.1.9.2.1 Overview

This package defines the modulus value attribute which is used when an observed counter overflows and counter differences are to be calculated (see 8.1.9.1).

8.1.9.2.2 Attributes of the counter overflow package

The counter overflow package contains the following attribute:

– *Modulus value*

This attribute holds the value to be used as the modulus value when an observed counter overflows.

8.1.9.2.3 Behaviour of the counter overflow package

The gauge value derived (V[t]) is calculated using the following method:

If [counter[t] – counter[t-GP]] is positive

V[t] = counter[t] – counter[t-GP].

If [counter[t] – counter[t-GP]] is negative

V[t] = [counter[t] – counter[t-GP] + MOD],

where MOD is the value of the modulus attribute and counter[t-GP] is the value of the previous scan counter value attribute.

If the value of the modulus value attribute is zero the actual modulus value used to evaluate V[t] is a local matter.

8.1.9.3 Gauge difference package

8.1.9.3.1 Overview

This package defines the behaviour for deriving a derivative of gauge from an observed attribute that is a gauge. The gauge value derived is the difference between the observed gauge values for two successive observations. (This difference may be negative.) The observed value retained by the metric object is to be used in calculating the next difference.

8.1.9.3.2 Attributes of the gauge difference package

The gauge difference package contains the following attributes:

a) *Previous scan gauge value*

This attribute is used in calculating gauge differences. It contains the value of the gauge captured by the latest observation.

b) *Procedural status*

This attribute is defined in CCITT Rec. X.731 | ISO/IEC 10164-2. It is used to convey the initialization status of the gauge difference package. An empty value indicates that the gauge difference package is fully initialized.

8.1.9.3.3 Behaviour of the gauge difference package

The derived gauge value is initially calculated using the following equation:

$$V[t] = gauge[t] - gauge[t\text{-}GP]$$

where

V[t]	is the difference between successive observations of the gauge;
gauge[t]	is the value of the gauge at the current time t (not saved in any attribute);
gauge[t-GP]	is the value of the gauge at time t minus GP (i.e. the value of the previous scan gauge value attribute);
GP	is the sampling interval in seconds, minutes, hours, or days (i.e. granularity period).

The initial value of the previous scan gauge value attribute determines the initial difference value.

The value of the previous scan gauge value attribute is considered valid if it was obtained at the previous granularity period. Thus, for example, it is not considered valid in the first scan interval subsequent to creation, in the first scan interval subsequent to transition from “unlocked” to “locked” states or subsequent to transition from “off-duty” to “on-duty” states.

During the period in which the value of the previous scan gauge value attribute is not valid, the value of the procedural status attribute shall have the value “Not initialized”. During this period, the value of the derived gauge attribute is not usable as input to subsequent data enhancement or data analysis algorithms, and any attempt to read either the derived gauge attribute or the previous scan gauge value attribute will result in the retrieval of invalid information.

The observed value is retained by the metric object to be used in calculating the next difference, i.e., it becomes the value of the previous scan gauge value attribute.

During the period after which the previous scan gauge value attribute is initialized but before the end of next granularity period, the value of the procedural status attribute shall have the value "Initializing". During this period, the value of the derived gauge attribute is not usable as input to subsequent data enhancement or data analysis algorithms, and any attempt to read the derived gauge attribute will result in the retrieval of invalid information.

8.1.9.4 Derived gauge-threshold package

8.1.9.4.1 Overview

This package is used to trigger notifications when the value of the derived gauge attribute (in the managed object in which the derived gauge-threshold package is included) crosses threshold levels. The threshold levels are specified in the severity indicating gauge-threshold attribute of the derived gauge-threshold package.

8.1.9.4.2 Attributes of the derived gauge-threshold package

The derived gauge-threshold package contains the following attribute:

– *Severity indicating gauge-threshold*

  This attribute contains the threshold levels which are to be applied to the derived gauge attribute. It shall be initialised when the managed object in which it is included is created and may be modified. The attribute is set-valued to allow multiple threshold levels to be specified. An optional parameter is used to associate the threshold level to the severity parameter of the emitted notification. The generation of the notification can be switched off using the boolean parameter notify-on-off. The severity parameter is mandatory if notify-on-off is "true".

  This attribute has similar behaviour to the gauge-threshold attribute defined in CCITT Rec. X.721 | ISO/IEC 10165-2 for triggering the related notifications. The syntax has an added parameter for indicating the associated severity to the notification triggered by the crossing of the corresponding threshold level. As an enhancement to the syntax of the gauge-threshold attribute type it adds an optional severity indication parameter to the syntax of both the notify-high and notify-low submembers within each threshold level member. This attribute type has additional behaviour associated with these optional perceived severity indication parameters, which is defined as follows:

  – If the notify-high's switch is on (true), the notify-high's severity indication value shall be reported in the perceived severity parameter of a notification triggered by the gauge value crossing the notify-high's gauge-threshold value in the positive going direction.

  – If the notify-low's switch is on (true), the notify-low's severity indication value shall be reported in the perceived severity parameter of a notification triggered by the gauge value crossing the notify-low's gauge-threshold value in the negative going direction.

    If both switches are on (true) for a single threshold level, one of the severity indication values shall be "clear". The severity indicating gauge-threshold shall only emit a clear event notification if the corresponding threshold level (either notify-high or notify-low) notification has been emitted and no other clear notification for this threshold level pair has been emitted since the previous corresponding threshold level notification has been emitted.

8.1.9.4.3 Behaviour of the derived gauge-threshold package

The derived gauge-threshold package has no additional behaviour beyond the behaviour of the severity indicating gauge-threshold attribute.

8.1.9.5 Estimate of mean threshold package

8.1.9.5.1 Overview

This package is used to trigger notifications when the value of the estimate of mean attribute (in the managed object in which the estimate of mean threshold package is included) crosses threshold levels. The threshold levels are specified in the estimate of mean severity indicating gauge-threshold attribute of the estimate of mean threshold package.

8.1.9.5.2 Attributes of the estimate of mean threshold package

The estimate of mean threshold package contains the following attribute:

– Estimate of mean severity indicating gauge-threshold

  This attribute contains the threshold levels which are to be applied to the Estimate of the Mean attribute. It has the same syntax and behaviour as the severity indicating gauge-threshold of the derived gauge-threshold package (see 8.1.9.4.2) except that the threshold levels are applied to the estimate of mean attribute (instead of the derived gauge attribute) of the managed object in which it is included.

8.1.9.5.3 Behaviour of the estimate of mean threshold package

The estimate of mean threshold package has no additional behaviour beyond the behaviour of the estimate of mean severity indicating gauge-threshold attribute.

8.1.9.6 Derived gauge timestamp package

8.1.9.6.1 Overview

The derived gauge timestamp package enables timestamping associated with updates to derived gauge values.

8.1.9.6.2 Attributes of the derived gauge timestamp package

The derived gauge timestamp package contains the following attribute:

– *Derived gauge timestamp*

  This attribute contains the time that the observation was taken. The derived gauge timestamp attribute is read-only.

8.1.9.6.3 Behaviour of the derived gauge timestamp package

The derived gauge timestamp package enables timestamping associated with updates to derived gauge values by including the time of the last observation in an attribute.

8.1.9.7 Specific problems indication package

8.1.9.7.1 Overview

The specific problems indication package enables the indication of a specific type of metric monitoring.

8.1.9.7.2 Attributes of the specific problems indication package

The specific problems indication package contains the following attribute:

– *Specific problems indicator*

  This attribute is used to indicate a specific type of metric monitoring. The attribute is specified at the metric object creation time. The value of the attribute is used to set the specific problems parameter of the quality of service alarm notification. Three specific problems are registered for workload monitoring: resource utilization, resource request rate and rejection rate.

8.1.9.7.3 Behaviour of the specific problems indication package

The specific problems indication package enables the indication of a specific type of metric monitoring. A specific type of problem can be associated with a threshold crossing.

8.1.9.8 Configurable percentile package

8.1.9.8.1 Overview

The configurable percentile package is used to allow a change from the default value of PCT. A particular value for PCT can be set so that the PCTth and 100-PCTth percentiles can be calculated.

8.1.9.8.2 Attributes of the configurable percentile package

The configurable percentile package contains the following attribute:

– *Configurable PCT*

  This attribute contains the value of PCT used in the percentile calculations.

8.1.9.8.3 Behaviour of the configurable percentile package

Estimates of the percentiles are obtained by taking a set of statistically independent samples of the observations of V[t] within a specified time interval and sorting these statistically independent observations of V[t] from smallest to largest. Repeated statistically independent samples are called replications.

If the configurable percentile package is not present, PCT shall be 25, and the number of replications attribute shall be 8, 16 or 32. If the configurable percentile package is present, PCT shall take the value of the configurable PCT attribute, and the number of replications shall admit arbitrary values (typically greater than 32 for PCT < 25, with a larger number of replications required for good performance for smaller PCT values).

8.1.9.9 Period synchronization package

8.1.9.9.1 Overview

The period synchronization package provides a mechanism for specifying that synchronization of repeating time periods is desired, and it specifies the time to which the repeating time periods is synchronized.

8.1.9.9.2 Attributes of the period synchronization package

The period synchronization package contains the following attribute:

– Period synchronization time

  This attribute contains the time to which a repeating time period is synchronized. The attribute is of ASN.1 type GeneralizedTime. The start of each time period is an integral number of periods before or after the time specified by this attribute.

  NOTE – Clock synchronization may be needed across multiple systems.

8.1.9.9.3 Behaviour of the period synchronization package

The period synchronization package has no additional behaviour beyond the behaviour of the period synchronization time attribute.

8.2 Imported generic definitions

8.2.1 Use of quality of service alarm notification

Parameters corresponding to alarm information in the alarm reporting service (see CCITT Rec. X.733 | ISO/IEC 10164-4) and their semantics are presented below. These parameters shall be present in the notification unless otherwise stated. This Recommendation | International Standard may use all of the parameters defined in the alarm report:

– Event type: Quality of service alarm;

– Probable cause: Threshold crossed;

– Specific problems: For use in further refining the threshold crossed quality of service alarm notification the following three have been registered in this Recommendation | International Standard: Resource utilization, Rejection rate, and Resource request rate;

– Perceived severity: The severity associated with a specified threshold level is defined on a managed object basis;

– Threshold information: The threshold information is used for the derived gauge-threshold of the monitor metric or for the estimate of the mean threshold of the mean monitor;

– Monitored attributes: This parameter shall be used to send the observed object instance and the observed attribute identifier attributes. In addition, all attributes whose values are metrics, other than the one that triggered the alarm notification, shall be included;

– State change: If present the parameter shall indicate the change in the operational state, administrative state or availability status attributes of the metric object.

8.3 Compliance

Managed object class definitions may import the appropriate specification of managed objects, notifications, actions, and/or attribute types defined in this standard. This is achieved by reference to the templates defined in this Recommendation | International Standard and in CCITT Rec. X.721 | ISO/IEC 10165-2. The reference mechanism is defined in CCITT Rec. X.722 | ISO/IEC 10165-4.

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-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;

– the State change reporting service defined in CCITT Rec. X.731 | ISO/IEC 10164-2; and

– the Alarm reporting service defined in CCITT Rec. X.733 | ISO/IEC 10164-4.

10 System management functional units

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

– control;

– monitor;

– objectEvents.

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

– stateChangeReporting

The following functional units defined in CCITT Rec. X.732 | ISO/IEC 10164-4 may be negotiated for the purpose of managing metric objects:

– alarmReporting

11 Protocol and abstract syntax

11.1 Abstract syntax

11.1.1 Managed objects

11.1.1.1 Referenced managed objects

This Recommendation | International Standard references the following support managed object for which the abstract syntax is specified in CCITT Rec. X.721 | ISO/IEC 10165-2:

– top

11.1.1.2 Defined managed objects

Table 1 identifies the relationship between the managed objects defined in 8.1 and the managed object class specifications in Annex A.

Table 1 – Reference labels for managed objects defined in this Recommendation | International Standard

Managed object name	Reference label
Algorithm indicating mean monitor	algorithmIndicatingMeanMonitor
Mean and min max monitor	meanAndMinMaxMonitor
Mean and percentile monitor	meanAndPercentileMonitor
Mean and variance monitor	meanAndVarianceMonitor
Mean monitor	meanMonitor
Monitor metric	monitorMetric
Moving average mean monitor	movingAverageMeanMonitor
Scanner	scanner

11.1.2 Attributes

11.1.2.1 Attributes imported from the definition of management information

This Recommendation | International Standard references the following management attributes, whose abstract syntax are specified in CCITT Rec. X.721 | ISO/IEC 10165-2:

a) administrativeState;

b) allomorphs

c) availabilityStatus;

d) counter;

e) gauge;

f) objectClass;

g) operationalState;

h) packages;

i) proceduralStatus.

11.1.2.2 Attributes defined in this Recommendation | International Standard

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

a) algorithmIdentifier;

b) configurablePCT;

c) derivedGauge;

d) derivedGaugeTimestamp;

e) estimateOfLargest;

f) estimateOfLargestInReplication;

g) estimateOfMean;

h) estimateOfMeanSeverityIndicatingGaugeThreshold;

i) estimateOfMedian;

j) estimateOfPCTPercentile;

k) estimateOfSmallest;

l) estimateOfSmallestInReplication;

m) estimateOfVariance;

n) estimateOf100-PCTPercentile;

o) granularityPeriod;

p) periodSynchronizationTime;

q) previousScanCounterValue;

r) previousScanGaugeValue;

s) modulusValue;

t) movingTimePeriod;

u) numberOfReplications;

v) observedAttributeId;

w) observedObjectInstance;

x) secondMovingTimePeriod;

y) severityIndicatingGaugeThreshold;

z) scannerId;

aa) specificProblemsIndicator.

Table 2 identifies the relationship between the attributes defined in 8.1 and the attribute type specifications in Annex A.

11.1.3 Notifications

11.1.3.1 Referenced notifications

This Recommendation | International Standard references the following events defined in CCITT Rec. X.730 | ISO/IEC 10164-1:

a) attribute value change notification;

b) object creation notification;

c) object deletion notification.

This Recommendation | International Standard references the following event defined in CCITT Rec. X.731 | ISO/IEC 10164-2:

– state change notification

This Recommendation | International Standard references the following event defined in CCITT Rec. X.733 | ISO/IEC 10164-4:

– quality of service alarm notification

11.1.3.2 Defined notifications

This Recommendation | International Standard defines no notifications.

11.1.4 Specific problems

Table 3 identifies the relationship between the specific problems defined in 8.2.1 and the ASN.1 value references defined in Annex A.

Table 2 – Attributes

Attribute name in clause 8	Attribute name in Annex A
Algorithm identifier	algorithmIdentifier
Configurable PCT	configurablePCT
Derived gauge	derivedGauge
Derived gauge timestamp	derivedGaugeTimestamp
Estimate of largest	estimateOfLargest
Estimate of largest in replication	estimateOfLargestInReplication
Estimate of mean	estimateOfMean
Estimate of mean severity indicating gauge-threshold	estimateOfMeanSeverityIndicatingGaugeThreshold
Estimate of median	estimateOfMedian
Estimate of PCT percentile	estimateOfPCTPercentile
Estimate of smallest	estimateOfSmallest
Estimate of smallest in replication	estimateOfSmallestInReplication
Estimate of variance	estimateOfVariance
Estimate of 100-PCT percentile	estimateOf100-PCTPercentile
Granularity period	granularityPeriod
Modulus value	modulusValue
Moving time period	movingTimePeriod
Number of replications	numberOfReplications
Observed attribute identifier	observedAttributeId
Observed object instance	observedObjectInstance
Period synchronization time	periodSynchronizationTime
Previous scan counter value	previousScanCounterValue
Previous scan gauge value	previousScanGaugeValue
Scanner Id	scannerId
Second moving time period	secondMovingTimePeriod
Severity indicating gauge-threshold	severityIndicatingGaugeThreshold
Specific problems indicator	specificProblemsIndicator

Table 3 – Specific Problems

Specific problems name	ASN.1 value reference
Resource request rate	resourceRequestRate
Resource utilization	resourceUtilization
Rejection rate	rejectionRate

11.2 Negotiation of functional units

Within the Systems management application context, the mechanism for negotiating the functional units is described by CCITT Rec. X.701 | ISO/IEC 10040.

NOTE – The requirement to negotiate functional units is specified by the application context.

12 Relationships with other functions

This Recommendation | International Standard uses services defined in CCITT Rec. X.731 | ISO/IEC 10164-2 for the notification of state changes, the services defined in CCITT Rec. X.730 | ISO/IEC 10164-1 for the creation and deletion of managed objects, the retrieval of attributes, the modification of attribute values, and notification of attribute value changes, and the services defined in CCITT Rec. X.733 | ISO/IEC 10164-4 for the reporting of alarms.

13 Conformance

There are two conformance classes: general conformance class and dependent conformance class. A system claiming to implement the elements of procedure for system management services referenced by this Recommendation | International Standard shall comply with the requirements for either the general or the dependent conformance class as defined in the following clauses. The supplier of the implementation shall state the class to which the conformance is claimed.

NOTE – The use of the two terms “general conformance class” and “dependent conformance class”, is under review. However, this standard continues to use these terms in order to be consistent with CCITT Rec. X.701 | ISO/IEC 10040 and other standards under the general title Information technology – Open Systems Interconnection – Systems Management. When the review has been completed, it is intended to clarify and/or correct this conformance clause together with the related clauses in those other Recommendations | International Standards.

13.1 General conformance class requirements

A system claiming general conformance shall support this function for all managed object classes that import the management information defined in this Recommendation | International Standard.

NOTE – This is applicable to all subclasses of the management support object classes defined in this Recommendation | International Standard.

13.1.1 Static conformance

The system shall:

a) when acting in the agent role, support one or more instances of at least one of the metric managed object classes or any of their subclasses;

b) support the transfer syntax derived from the encoding rules specified in CCITT Rec. X.209 | ISO/IEC 8825 and named {joint-iso-ccitt asn1(1) basic-encoding(1)}, for the purpose of generating and interpreting the MAPDUs, defined by the abstract data types referenced in 11.1.3.

A system need not support sending all of the named types within the MetricModule.TimePeriod CHOICE.

A system supporting the algorithm indicating mean monitor managed object class need not conform to the algorithm indicated in the algorithm identifier attribute.

13.1.2 Dynamic conformance

The system shall, in the role(s) for which conformance is claimed:

- support the elements of procedure defined in:
  - CCITT Rec. X.730 | ISO/IEC 10164-1 for the PT-GET, PT-CREATE, PT-DELETE, PT-SET services.
  - CCITT Rec. X.730 | ISO/IEC 10164-1 for the Object creation reporting, Object deletion reporting, and Attribute value change reporting services.
  - CCITT Rec. X.731 | ISO/IEC 10164-2 for the State change reporting service.
  - CCITT Rec. X.733 | ISO/IEC 10164-4 for the Alarm reporting service.

13.2 Dependent conformance class requirements

13.2.1 Static conformance

The system shall:

a) when acting in the agent role, support one or more instances of at least one of the metric managed object classes or any of their subclasses;

b) support the transfer syntax derived from the encoding rules specified in CCITT Rec. X.209 | ISO/IEC 8825 and named {joint-iso-ccitt asn1(1) basic-encoding(1)}, for the purpose of generating and interpreting the MAPDUs, defined by the abstract data types referenced in 11.1.3 as required by a referencing Recommendation | International Standard.

A system need not support sending all of the named types within the MetricModule.TimePeriod CHOICE.

A system supporting the algorithm indicating mean monitor managed object class need not conform to the algorithm indicated in the algorithm identifier attribute.

13.2.2 Dynamic conformance

The system shall support the elements of procedure referenced by this Recommendation | International Standard, as required by a standardized use of this system management function.

13.3 Conformance to support managed object definitions

The metric objects supported by the open system shall comply with the behaviour specified in clause 8 and the syntax specified in Annex A. For minimum conformance to support managed objects defined in this Recommendation | International Standard, at least one of the metric object classes or any of their subclasses shall be supported in at least read-only mode, and at least one of the quality of service alarm notifications shall be supported.

NOTE – The MOCS and PICS Proforma are for further study.

Annex A

Metric objects

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

NOTE – The definition of metric object classes does not prevent managed object definers from including management information defined in clause 8 within the definition of other managed object classes. When defining new managed object classes with metric attributes, the definers should consider the utilization of the metric object classes defined in this annex.

A.1 Managed object class definitions

```
algorithmIndicatingMeanMonitor  MANAGED OBJECT CLASS
DERIVED FROM meanMonitor;
CHARACTERIZED BY
algorithmIndicatingMeanMonitorPackage PACKAGE
        BEHAVIOUR
        algorithmIndicatingMeanMonitorBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.4.3.";;
        ATTRIBUTES
        algorithmIdentifier GET;;;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 8};

meanAndMinMaxMonitor  MANAGED OBJECT CLASS
DERIVED FROM movingAverageMeanMonitor;
CHARACTERIZED BY
meanAndMinMaxMonitorPackage PACKAGE
        BEHAVIOUR
        meanAndMinMaxMonitorBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.8.3.";;
        ATTRIBUTES
        estimateOfLargest GET-REPLACE,          --  See 8.1.8.2.
        estimateOfSmallest GET-REPLACE;;;       --  See 8.1.8.2.
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 1};

meanAndPercentileMonitor MANAGED OBJECT CLASS
DERIVED FROM movingAverageMeanMonitor;
CHARACTERIZED BY
meanAndPercentileMonitorPackage PACKAGE
        BEHAVIOUR
        meanAndPercentileMonitorBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.7.3.";,
        ewmaGaugePercentileBehaviour BEHAVIOUR
        DEFINED AS "See B.2.4";;
        ATTRIBUTES
        secondMovingTimePeriod GET-REPLACE,             --  used in GaugePercentileBehaviour to calculate T2,
                                                        --  See B.2.4 and 8.1.7.2.

        estimateOfLargestInReplication GET-REPLACE,     --  See B.2.4 and 8.1.7.2.
        estimateOfSmallestInReplication GET-REPLACE,    --  See B.2.4 and 8.1.7.2.
        estimateOfMedian GET-REPLACE,                   --  See B.2.4 and 8.1.7.2.
        estimateOf100-PCTPercentile GET-REPLACE,        --  See B.2.4  and 8.1.7.2.
        estimateOfPCTPercentile GET-REPLACE,            --  See B.2.4 and 8.1.7.2.
        numberOfReplications GET-REPLACE                --  See B.2.4 and 8.1.7.2.
        ;;;
CONDITIONAL PACKAGES
        configurablePercentilePackage  PRESENT IF "configurable percentiles is required. See 8.1.7.2.";
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 2};
```

```
meanAndVarianceMonitor MANAGED OBJECT CLASS
DERIVED FROM movingAverageMeanMonitor;
CHARACTERIZED BY
meanAndVarianceMonitorPackage PACKAGE
        BEHAVIOUR
        meanAndVarianceMonitorBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.6.3.";,
        ewmaGaugeVarianceBehaviour BEHAVIOUR
        DEFINED AS "See B.2.3." ;;
        ATTRIBUTES
        secondMovingTimePeriod GET-REPLACE,      -- used in EWMA Gauge Variance Behaviour to calculate "g".
        estimateOfVariance GET-REPLACE ;;;       -- See B.2.3.
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 3};

meanMonitor MANAGED OBJECT CLASS
DERIVED FROM monitorMetric;
CHARACTERIZED BY
meanMonitorPackage PACKAGE
        BEHAVIOUR
        meanMonitorBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.3.3." ;;
        ATTRIBUTES
        estimateOfMean GET-REPLACE,
        movingTimePeriod GET-REPLACE;;;
CONDITIONAL PACKAGES
        estimateOfMeanThresholdPackage
        PRESENT IF "threshold on estimateOfMean is required",
        specificProblemsIndicationPackage
        PRESENT IF   "specific problems indication behaviour is required and either the estimate of mean threshold
                     package is present or the derived gauge-threshold package is present";
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 4};

monitorMetric MANAGED OBJECT CLASS
DERIVED FROM scanner;
CHARACTERIZED BY
monitorMetricPackage PACKAGE
        BEHAVIOUR
        monitorMetricBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.2.3.";;
        ATTRIBUTES
        observedObjectInstance GET,
        observedAttributeId GET,
        derivedGauge GET;;;
CONDITIONAL PACKAGES

        counterDifferencePackage  PRESENT IF           "counter to gauge conversion is required and
                                                       gaugeDifferencePackage is not present",
        counterOverflowPackage  PRESENT IF             "the counterDifference package is present and modulo
                                                       arithmetic is required to calculate the new value of the derived
                                                       gauge on counter overflow",
        gaugeDifferencePackage  PRESENT IF             "gauge to derivative conversion is required and
                                                       counterDifferencePackage is not present",
        derivedGaugeThresholdPackage  PRESENT IF       "threshold on derivedGauge is required",
        specificProblemsIndicationPackage  PRESENT IF  "specific problems indication behaviour is required and the
                                                       derivedGaugeThresholdPackage is present",
        derivedGaugeTimestampPackage PRESENT IF        "timestamping associated with updating the derivedGauge is
                                                       required";
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 5};
```

```
movingAverageMeanMonitor MANAGED OBJECT CLASS
DERIVED FROM meanMonitor;
CHARACTERIZED BY
movingAverageMeanMonitorPackage PACKAGE
        BEHAVIOUR
        movingAverageMeanMonitorBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.5.3." ;;;;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 6};

scanner MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":top;
CHARACTERIZED BY
        scannerPackage PACKAGE
        BEHAVIOUR
        scannerBehaviour BEHAVIOUR
        DEFINED AS  "See 8.1.1.3.";;
        ATTRIBUTES
        scannerId   GET,
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":     administrativeState  GET-REPLACE,
        granularityPeriod   GET-REPLACE,
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":     operationalState     GET;;;
CONDITIONAL PACKAGES
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":     availabilityStatusPackage
                        PRESENT IF          "the managed object can be scheduled",
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":     duration
                        PRESENT IF          "the scanning function is to be enabled between specified start
                                            and stop times",
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":     dailyScheduling
                        PRESENT IF          "daily scheduling is required and the weekly or external
                                            scheduling package is not present",
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":     weeklyScheduling
                        PRESENT IF          "weekly scheduling is required and the daily or external
                                            scheduling package is not present",
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":     externalScheduler
                        PRESENT IF          "reference to external scheduler is required and the daily or
                                            weekly scheduling package is not present",
        periodSynchronizationPackage
                        PRESENT IF          "configurable agent internal synchronization of repeating time
                                            periods is required",
        "Recommendation M.3100:1992":createDeleteNotificationsPackage
                        PRESENT IF          "notification of object creation and object deletion events is
                                            required",
        "Recommendation M.3100:1992":attributeValueChangeNotificationPackage
                        PRESENT IF          "notification of attribute value change events is required",
        "Recommendation M.3100:1992":stateChangeNotificationPackage
                        PRESENT IF          "notification of state change event is required"
        ;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) managedObjectClass(3) 7};
```

A.2 Package definitions

```
configurablePercentilePackage PACKAGE
        BEHAVIOUR
        configurablePercentilePackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.8." ;;
        ATTRIBUTES
        configurablePCT GET-REPLACE ;                -- Allowed values are 1 to 49, inclusive
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 1};
```

```
counterDifferencePackage PACKAGE
        BEHAVIOUR
        counterDifferencePackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.1.3." ;;
        ATTRIBUTES
        previousScanCounterValue   GET,
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":proceduralStatus  GET;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 2};

counterOverflowPackage PACKAGE
        BEHAVIOUR
        counterOverflowPackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.2.3.";;
        ATTRIBUTES
        modulusValue   GET-REPLACE;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 3};

derivedGaugeThresholdPackage PACKAGE
        BEHAVIOUR
        derivedGaugeThresholdPackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.4.4.";;
        ATTRIBUTES
        severityIndicatingGaugeThreshold GET-REPLACE ADD-REMOVE;
        NOTIFICATIONS
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":qualityofServiceAlarm;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 4};

derivedGaugeTimestampPackage PACKAGE
        BEHAVIOUR
        derivedGaugeTimestampPackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.6.";;
        ATTRIBUTES
        derivedGaugeTimestamp GET;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 5};

estimateOfMeanThresholdPackage PACKAGE
        BEHAVIOUR
        estimateOfMeanThresholdPackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.5.3.";;
        ATTRIBUTES
        estimateOfMeanSeverityIndicatingGaugeThreshold GET-REPLACE ADD-REMOVE;
        NOTIFICATIONS
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":qualityofServiceAlarm;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 6};

gaugeDifferencePackage PACKAGE
        BEHAVIOUR
        gaugeDifferencePackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.3.3.";;
        ATTRIBUTES
        previousScanGaugeValue   GET,
        "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":proceduralStatus  GET;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 8};

periodSynchronizationPackage   PACKAGE
        BEHAVIOUR
        periodSynchronizationPackageBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.9.3." ;;
        ATTRIBUTES
        periodSynchronizationTime GET-REPLACE;
REGISTERED AS  { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 10};
```

```
specificProblemsIndicationPackage  PACKAGE
        BEHAVIOUR
        specificProblemsIndicationPackageBehaviour BEHAVIOUR
        DEFINED AS    "Included if specific problems indication is wanted in the metric object instance,
                      See 8.1.9.7.3.";;
        ATTRIBUTES
        specificProblemsIndicator GET-REPLACE;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) package(4) 9};
```

A.3 Attribute definitions

```
algorithmIdentifier   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.AlgorithmIdentifier;
        BEHAVIOUR
        algorithmIdBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.4.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 26};

configurablePCT   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.Integer;
        BEHAVIOUR
        configurablePCTBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.8.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 0};

derivedGauge   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        derivedGaugeBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.2.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 2};

derivedGaugeTimeStamp   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.GlobalTime;
        BEHAVIOUR
        derivedGaugeTimeStampBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.6.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 3};

estimateOfLargest   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfLargestBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.8.3.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 4};

estimateOfLargestInReplication   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfLargestInReplicationBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.7.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 21};

estimateOfMean   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfMeanBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.3.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 5};
```

```
estimateOfMeanSeverityIndicatingGaugeThreshold   ATTRIBUTE
        DERIVED FROM severityIndicatingGaugeThreshold;
        BEHAVIOUR
        estimateOfMeanSeverityIndicatingGaugeThresholdBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.5.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 6};

estimateOfMedian   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfMedianBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.7.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 7};

estimateOfPCTPercentile   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfPCTPercentileBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.7.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 8};

estimateOfSmallest   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfSmallestBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.8.3.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 9};

estimateOfSmallestInReplication   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfSmallestInReplicationBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.7.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 22};

estimateOfVariance   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOfVarianceBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.6.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 10};

estimateOf100-PCTPercentile   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        estimateOf100-PCTPercentileBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.7.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 11};

granularityPeriod   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   MetricModule.TimePeriod;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR
        granularityPeriodBehaviour BEHAVIOUR
        DEFINED AS   "This attribute specifies the time between two successive scans. See 8.1.1.3.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 23};

modulusValue   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":counter;
        BEHAVIOUR
        modulusValueBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.2.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 12};
```

```
movingTimePeriod   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.TimePeriod;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR
        movingTimePeriodBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.3.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 13};

numberOfReplications   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.Integer;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR
        numberOfReplicationsBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.7.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 14};

observedAttributeId   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX CMIP-1.AttributeId;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        observedAttributeIdBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.2.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 15};

observedObjectInstance   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX CMIP-1.ObjectInstance;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        observedObjectInstanceBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.2.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 16};

periodSynchronizationTime   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  MetricModule.GlobalTime;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        periodSynchronizationTimeBehaviour BEHAVIOUR
        DEFINED AS      "This attribute specifies the synchronization time for repeating periods. The start for each
                        period is at a time which an integral number of periods before or after the
                        periodSynchronizationTime. See 8.1.9.9.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 24};

previousScanCounterValue   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":counter;
        BEHAVIOUR
        previousScanCounterValueBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.1.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 1};

previousScanGaugeValue   ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":gauge;
        BEHAVIOUR
        previousScanGaugeValueBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.3.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 20};

scannerId   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.SimpleNameType;
        MATCHES FOR EQUALITY ;
        BEHAVIOUR
        scannerIdBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.1.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 25};
```

```
secondMovingTimePeriod   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.TimePeriod;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR
        secondMovingTimePeriodBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.6.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 17};

severityIndicatingGaugeThreshold   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.SeverityIndicatingGaugeThreshold;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        severityIndicatingGaugeThresholdBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.4.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 18};

specificProblemsIndicator   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX MetricModule.ObjectIdentifier;
        MATCHES FOR EQUALITY;
        BEHAVIOUR
        specificProblemsIndicatorBehaviour BEHAVIOUR
        DEFINED AS "See 8.1.9.7.2.";;
REGISTERED AS { joint-iso-ccitt ms(9) function(2) part11(11) attribute(7) 19};
```

A.4 Name binding definitions

```
scanner-system NAME BINDING
        SUBORDINATE OBJECT CLASS scanner AND SUBCLASSES;
        NAMED BY
        SUPERIOR OBJECT CLASS "CCITT Rec. X.721 | ISO/IEC 10165-2:1992":system AND SUBCLASSES;
        WITH ATTRIBUTE scannerId;
        CREATE
                WITH-REFERENCE-OBJECT,
                WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
                ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS {joint-iso-ccitt ms(9) function(2) part11(11) nameBinding(6) 0};
```

A.5 ASN.1 definitions

```
MetricModule {joint-iso-ccitt ms(9) function(2) part11(11) asn1Module(2) 0}
DEFINITIONS IMPLICIT TAGS ::=
BEGIN

-- EXPORTS everything

IMPORTS

        ObservedValue, PerceivedSeverity, SimpleNameType, SpecificIdentifier  FROM
        Attribute-ASN1Module  {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1};

-- Value assignments for the specific problems parameter
moaSpecificProblems OBJECT IDENTIFIER ::=
        { joint-iso-ccitt ms(9) function(2) part11(11) standardSpecificExtension(0) 0}

resourceRequestRate SpecificIdentifier ::= { moaSpecificProblems 1 }
resourceUtilization SpecificIdentifier ::= { moaSpecificProblems 2 }
rejectionRate SpecificIdentifier ::= { moaSpecificProblems 3 }

moaAlgorithmIdentifiers OBJECT IDENTIFIER ::=
        { joint-iso-ccitt ms(9) function(2) part11(11) standardSpecificExtension(0) algorithm(1) }
```

```
ewmaAlgorithm AlgorithmIdentifier ::= { moaAlgorithmIdentifiers 0 }
        --This identifier is allocated to indicate that the EWMA algorithm is used for data smoothing.
        --The EWMA algorithm is defined in B.2.

uwmaAlgorithm AlgorithmIdentifier ::= { moaAlgorithmIdentifiers 1 }
        --This identifier is allocated to indicate that the UWMA algorithm is used for data smoothing.
        --The UWMA algorithm is defined in C.2.

AlgorithmIdentifier ::= OBJECT IDENTIFIER

GlobalTime ::= GeneralizedTime

Integer ::= INTEGER

ObjectIdentifier ::= OBJECT IDENTIFIER

SeverityIndicatingGaugeThreshold ::= SET OF SEQUENCE {
        notifyLow SeverityIndicatingThreshold,
        notifyHigh SeverityIndicatingThreshold
}

SeverityIndicatingThreshold ::= SEQUENCE {
        threshold        ObservedValue,
        notifyOnOff      BOOLEAN,
        severityIndication PerceivedSeverity OPTIONAL
}

TimePeriod ::= CHOICE {
                days          [0] INTEGER,
                hours         [1] INTEGER,
                minutes       [2] INTEGER,
                seconds       [3] INTEGER,
                milliSeconds          [4] INTEGER,
                microSeconds          [5] INTEGER,
                nanoSeconds           [6] INTEGER,
                picoSeconds           [7] INTEGER
}

END
```

Annex B

EWMA data enhancement algorithms

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

B.1 Introduction

The text that follows presents the Exponentially Weighted Moving Average (EWMA) technique and defines its usage in obtaining estimators for a gauge V out of a sequence, V_1 ,V_2, ..., of derived gauge values.

B.1.1 Exponentially weighted moving average

The EWMA technique, also called the "exponential digital filter", is a smoothing technique used for data enhancement. In general, given a sequence of observations, X_1, X_2, ..., the EWMA of this time series, Y is defined through a recursive relationship of the form

$$Y_i = f \cdot X_i + (1 - f) \cdot Y_{i-1} \quad 0 < f \leq 1 \quad i = 1, 2, 3, \ldots$$

with Y_0, the start-up value, initialized by setting an attribute representing the estimate of the mean.

The specification of EWMA as it pertains to this International standard is given below.

B.2 Algorithm behaviour

B.2.1 Common definitions

f is the EWMA mean smoothing constant, and its value is $2 \cdot GP/(GP + MTP)$.

GP is the granularity period and equals the number of time units that theoretically elapse between any two successive gauge measurements (the actual times may vary slightly). It is the value of the granularity period attribute.

g is the parameter used for gauge variance exponential smoothing and its value is $2 \cdot GP/(GP + SMTP)$.

h is the parameter used for smoothing and its value is $(2 \cdot M \cdot GP)/(SMTP + M \cdot GP)$

MTP is the moving time period and, conceptually, corresponds to the moving window concept. (The moving window concept states that at any time t, only the observations that occurred within the last MTP time are considered.)

NOTE 1 – MTP should be no smaller than GP.

SMTP is the second MTP.

NOTE 2 – When the system calculates variance estimates, SMTP should exceed GP. When it calculates percentile estimates, SMTP should exceed $M \cdot GP$.

M is the number of replications. It is the value of the number of replications attribute.

B.2.2 EWMA gauge mean behaviour

This algorithm calculates the estimate of the mean of the values of a gauge V using the EWMA algorithm. The EWMA is defined by the following iterative algorithm.

$$W_n = f \cdot V_n + (1 - f) \cdot W_{n-1} \quad n = 1, 2, 3, \ldots$$

where

V_n is the current value of the gauge,

W_n is the estimate of the mean of the gauge at the time observation V_n is taken,

W_{n-1} is the previous estimate of the mean of the gauge. W_0 may be initialized by setting the estimateOfMean attribute.

NOTES

1 When GP = MTP, then f = 1 and $W_n = V_n$ for all n, n = 1, 2, 3, . . . This case (f = 1) is an extension of the usual assumptions about EWMA transformations, and is the degenerate case with no EWMA smoothing.

2 Both GP and MTP can be reset by the managing system. At each such occurrence, parameter f must be reevaluated.

3 A Uniformly Weighted Moving Average (UWMA) is obtained by averaging at each time T the M observations that occur between times T – MTP and T. The value of f (equal to 2 · GP/(GP + MTP)) is chosen in such a way that the UWMA (with a window size of MTP) and EWMA estimators give the same average age of data.

4 The initial condition (or reset condition) of the EWMA of the estimated mean is provided by setting the estimateOfMean attribute.

5 The units of V and W are the same.

B.2.3 EWMA gauge variance behaviour

This algorithm calculates the estimate of the variance of the values of a gauge using the EWMA algorithm. This algorithm shall be used in combination with the EWMA gauge mean algorithm which is used to calculate W_n.

The EWMA variance estimate is defined by the following iterative algorithm:

$$S_n = g \cdot (V_n - W_n)^2 + (1 - g) \cdot S_{n-1}$$

where

S_n is an estimate of the variance of V_n.

S_{n-1} is the previous estimate of the variance. S_0 may be initialized by setting the attribute representing the estimate of variance.

NOTES

1 The values of GP and SMTP can be reset by the managing system.

2 This algorithm has a predictable bias which can be shown to be:

$$u = 2 \cdot (1-f)^2/(2-f), \text{ times the variance of } V_n - W_n.$$

These factors can be used to reduce a small amount of bias that is introduced into the calculations, (e.g. for f = 0.05, u ≈ 0.93), which may be used in the following equation:

$$S'_n = S_n/u$$

This calculation may be performed by managing systems to correct any bias in variance calculations. Alternatively, the bias may be ignored by managing systems if it is considered negligible.

3 The units of S are the same as the units of V squared.

B.2.4 EWMA gauge percentile behaviour

The metric algorithms defined in this subclause include estimates of the smallest in a sample of size M, the largest in a sample of size M, the median, the PCTth percentile, and the 100-PCTth percentile of the relative frequency distribution of the observed attribute of the observed managed object.

By definition, the Kth percentile is the pth fractile for p = K/100. For reasons of economy of notation, the fractile notation is used.

The procedure for calculation of percentile estimates are carried out in two phases:

– a set of M replications of the gauge is obtained, and coarse estimates are calculated for the percentiles (as specified in B.2.4.1);

– each of these coarse estimates are applied in turn to an EWMA algorithm to derive smoothed estimates of percentiles, which are used to update the metric output attributes (as specified in B.2.4.2).

NOTE – Since percentiles are calculated for 100-PCT, when PCT is close to zero, those percentiles are correspondingly close to 100 (e.g. 98th percentile). For these large percentiles, if the distribution is such that there is a long tail, the coarse estimates may be significantly biased. The EWMA smoothing does not reduce this bias. Alternatives to reduce this bias are for further study.

B.2.4.1 Calculation behaviour for deriving coarse estimates for percentiles using M replications

Coarse estimates of sample percentiles are calculated from a sorted set of M sample values of the gauge, V. The repeated sampling of a random variable is called replicating. Each of these individual samples is called a replication.

The coarse calculations of percentiles are performed as follows:

1) The M replications are sorted to form a list X_1, X_2, ..., X_M sorted from the lowest value to the highest value.

2) An estimate of the p-fractile, $0 < p < 1$, of the underlying Probability distribution function is obtained as follows:

   – If $p \cdot (M + 1) \leq 1$, the estimate is X_1.

   – If $p \cdot (M + 1) \geq M$, the estimate is X_M.

   – If neither of the above conditions apply, then the p-fractile estimator Z[p] is given by the formula

$$Z[p] = X_j + W \cdot (X_{j+1} - X_j)$$

   where j is the integer part of $p \cdot (M + 1)$, and $w = p \cdot (M + 1) - j$.

3) By convention, Z[0] and Z[1] represent the minimum and maximum values obtained in a set of M sample values (replications).

NOTES

1 The algorithm above is based on the fact that if the original replications are independent and identically distributed with cumulative distribution function F(x), the expected value of $F(X_i)$ is $i/(M + 1)$. Thus, when $p = i/(M + 1)$, X_i is an estimator for the p-fractile. If, on the other hand, p falls between $i/(M + 1)$ and $(i + 1)/(M + 1)$, the p-fractile is determined through a linear combination of X_i and X_{i+1} (i.e. F is approximated between these two points through a linear function).

2 This algorithm is used to calculate fractiles for the following values of p: PCT/100, 0.5, and (100-PCT)/100. PCT is an attribute whose value is constrained to be between 0 and 50 (0 < PCT < 50). The default value of PCT is 25.

3 The algorithm outputs may be inaccurate when the fractiles calculated correspond to a flat (i.e. low density) region of the probability distribution.

B.2.4.2 EWMA percentile algorithm specification

Better percentile estimates can be obtained by exponential smoothing of the coarse percentile estimates.

Thus, if $Z_n[p]$ is a current coarse p-fractile estimate (i.e. an estimate obtained from the current, nth, set of replications), then the corresponding smoothed estimate $Q_n[p]$ equals

$$Q_n[p] = h \cdot Z_n[p] + (1 - h) \cdot Q_{n-1}[p]$$

where

p is either PCT/100, 0.5, and (100-PCT)/100, or the special values of 0 and 1, which, by convention imply the estimate of the largest and smallest values in the sample of size M; and

$Q_{n-1}[p]$ is the previous smoothed p-fractile estimate.

$Q_n[0]$ and $Q_n[1]$ are estimates of the expected largest and smallest values, respectively in a sample of size M based on the convention that $Z_n[0]$ and $Z_n[1]$ are defined as in B.2.4.1.

NOTES

1 It is assumed that distinct replication sets have no replications in common.

2 The initial condition (or reset condition) of the EWMA of the estimated percentile is provided by setting each $Q_0[p]$. This is done by setting the corresponding attributes.

Annex C

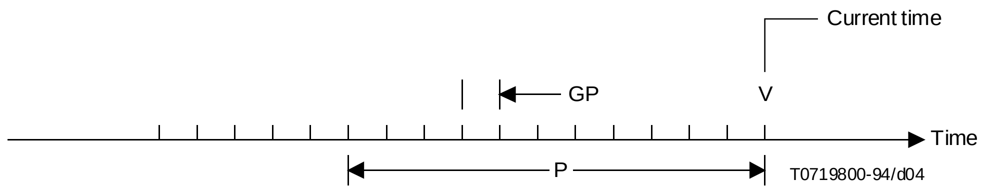
Uniformly weighted moving average smoothing algorithm

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

C.1 Uniformly weighted moving average

If the counts of events between observations occurring within a moving time period are stored for the entire moving time period, then an algorithm can be calculated exactly at the end of the interval between observations. To store these counts there must exist at least as many storage locations as there are observation intervals within the moving time period. At the end of each observation interval, the contents of the storage location containing the oldest value can be replaced with the newest count. If these counts are maintained then calculations can be performed which result in a "uniformly weighted moving average" (UWMA). In such cases the moving time period is known as the "sliding time window". The results are termed "uniformly weighted" because occurrences during the entire moving time period are weighted equally (i.e. uniformly) in calculating the algorithm at the end of the observation interval. The UWMA is further specified in the UWMA gauge mean behaviour (see C.2).

This is shown in the following diagram with GP indicating the length of an observation interval and P indicating the sliding time window:



C.2 UWMA gauge mean behaviour

The estimate of the mean (estimateOfMean), ~V(t), is defined to be a weighted moving average of a gauge, V[t]. The specified weighting is uniform. The uniformly weighted moving average (UWMA) is defined by the following algorithm.

The estimate of the gauge mean (estimateOfMean) is updated by the following equation:

$$\sim V[t] = \frac{\sum_{i=0}^{N-1} V[t - i \cdot GP]}{N}$$

where

t is the current value of time,

V[t] is the value of the derivedGauge attribute at time t,

V[t – i · GP] is the value of the derivedGauge attribute at time t – i · GP,

~V[t] is the estimate of the mean of V[t] at time t,

GP is the time between the successive observations of V[t], constrained to be a positive value larger than zero. It is sufficient to have a best effort to make GP a constant value. The value is given by the value of the granularity period attribute.

MTP is the moving time period over which the estimate of mean is calculated. It is the value of the moving time period attribute.

N is calculated as MTP/GP, truncated to an integer value.

NOTE 1 – When GP = MTP, the mean value of the gauge is the current gauge value, V(t).

The initial value of V[t – i · GP] for all values of i from zero to N-1 shall be set equal to the value provided in estimateOfMean when the managed object is created.

The units of ~V[t] are the same as the units of the gauge to which the estimate of the mean is applied.

NOTE 2 – To improve the initialization response time of the algorithm, an incrementing factor can be used. The factor is used to reduce the time it takes for the output value to reflect the estimate of the actual current mean. When used the initial value (~V[t – GP] for EWMA or V[t – i · GP] for UWMA) is not needed when the managed object is created or reset. When either algorithm is started the time constant in the algorithm is increased from GP to the value given in the attribute time constant T1 in multiples of GP. For example, if the factor value is two, the time constant in the algorithm will increase in steps of 2 · GP until it reaches the value given by the attribute time constant.

Annex D

Workload reporting requirements

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

This non-normative annex discusses how three of the metrics discussed in clause 7, Models, can be used for estimating needed capacity.

There are three metric algorithm output types which defined in this function which are particularly useful for workload monitoring. These three metric attribute types (with units) are:

W – estimated mean resource request rate (capacityUnits/timeUnits);

U – estimated mean resource utilization (capacityUnits);

R – estimated mean rejection rate (capacityUnits/timeUnits).

If the estimates for these three metric attributes are derived over a consistent period of time, they can used together to derive further estimates of quantities which can be used to determine the proper amount of capacity required to meet desired service levels.

As an example, the mean holding time, T, per resource server can be derived if all three estimates W, U, and R are available.

The carried workload acceptance rate, C, can be calculated as C = W – R.

The quantity C can be used together with U to determine the estimated mean holding time, using Little's Law:

T – Estimated mean holding time (timeUnits) = U/C.

Traffic theory defines a quantity called offered workload, A, as:

A – Offered workload (capacityUnits) = W · T.

The quantity A is used to design the required number of servers to meet desired quality of service, assuming distributions for instantaneous resource request rate and holding time.

In particular, given a Poisson distribution for arrivals and an Exponential distribution for holding time, the Erlang B formula provides a closed form solution for the rejection probability resulting from a specific value of A and a specific number of provided resource capacityUnits.