



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Q.827.1
(10/2004)

SERIES Q: SWITCHING AND SIGNALLING
Q3 interface

**Requirements and analysis for the common
management functions of NMS-EMS interfaces**

ITU-T Recommendation Q.827.1

ITU-T Q-SERIES RECOMMENDATIONS
SWITCHING AND SIGNALLING

SIGNALLING IN THE INTERNATIONAL MANUAL SERVICE	Q.1–Q.3
INTERNATIONAL AUTOMATIC AND SEMI-AUTOMATIC WORKING	Q.4–Q.59
FUNCTIONS AND INFORMATION FLOWS FOR SERVICES IN THE ISDN	Q.60–Q.99
CLAUSES APPLICABLE TO ITU-T STANDARD SYSTEMS	Q.100–Q.119
SPECIFICATIONS OF SIGNALLING SYSTEMS No. 4, 5, 6, R1 AND R2	Q.120–Q.499
DIGITAL EXCHANGES	Q.500–Q.599
INTERWORKING OF SIGNALLING SYSTEMS	Q.600–Q.699
SPECIFICATIONS OF SIGNALLING SYSTEM No. 7	Q.700–Q.799
Q3 INTERFACE	Q.800–Q.849
DIGITAL SUBSCRIBER SIGNALLING SYSTEM No. 1	Q.850–Q.999
PUBLIC LAND MOBILE NETWORK	Q.1000–Q.1099
INTERWORKING WITH SATELLITE MOBILE SYSTEMS	Q.1100–Q.1199
INTELLIGENT NETWORK	Q.1200–Q.1699
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2000	Q.1700–Q.1799
SPECIFICATIONS OF SIGNALLING RELATED TO BEARER INDEPENDENT CALL CONTROL (BICC)	Q.1900–Q.1999
BROADBAND ISDN	Q.2000–Q.2999

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

Document ID: 123456789

ITU-T Recommendation Q.827.1

Requirements and analysis for the common management functions of NMS-EMS interfaces

Summary

This Recommendation provides the requirements and analysis for the common management functions for NMS-EMS interface which can be used for network technology-specific management interfaces. The requirements and analysis for the management interface between EMS and NMS are provided using the TMN interface specification methodology described in ITU-T Rec. M.3020.

Source

ITU-T Recommendation Q.827.1 was approved on 7 October 2004 by ITU-T Study Group 4 (2001-2004) under the ITU-T Recommendation A.8 procedure.

Keywords

Common management, Unified Modelling Language (UML).

FOREWORD

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

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

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

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

NOTE

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

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

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

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

© ITU 2005

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

CONTENTS

	Page
1 Scope	1
2 References.....	1
3 Terms and definitions	2
4 Abbreviations.....	2
5 General overview.....	3
6 Requirements	3
6.1 Business level requirements	3
6.2 Specification level requirements	8
7 Analysis	26
7.1 Conventions	26
7.2 Common management function set	26
7.3 Performance management function set	57
7.4 Fault management function set.....	72
7.5 Security management function set.....	80
Appendix I – Table of managed entities	80

Downloaded from: 192.168.1.100

ITU-T Recommendation Q.827.1

Requirements and analysis for the common management functions of NMS-EMS interfaces

1 Scope

This Recommendation provides the requirements and analysis of common management functions applicable at NMS-EMS interface. The term "common" is used in two different contexts. In the first context, it includes analysis of the requirements that are applicable for fault, performance, configuration and security management functions. The second usage is to analyse the requirements that are independent of the specific technology. An example of the first context is logging events for alarms, performance threshold crossing, and inventory changes to the data of the entities. An example of the second context is the method for collecting and reporting performance measurements independent of the network technologies, such as EPON or SDH-DLC.

This Recommendation follows the interface specification methodology described in ITU-T Rec. M.3020 [2]. In this Recommendation, the Element Management System (EMS) is an Operation System (OS) [1] used to manage the individual network elements supporting specific network technologies as well as the networks between them. One or more systems may be required depending on the different supplier products and geographic distribution of the elements in the network. The Network Management System (NMS) represents an integrated management OS across different technology and supplier systems.

The management functions covered in this Recommendation include common management, performance management, fault management and security management. Configuration management and performance parameters are network technology-dependent, and they will be addressed in other technology-specific Recommendations. Security management is for future study. These management functions are described and divided by UML use cases in the requirements. The analysis provides the managed entities that support a protocol-neutral information model for common management functions, and also explains the static and dynamic relationships between these managed entities using UML class diagrams and sequence diagrams.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [1] ITU-T Recommendation M.3010 (2000), *Principles for a telecommunication management network*.
- [2] ITU-T Recommendation M.3020 (2000), *TMN interface specification methodology*.
- [3] ITU-T Recommendation M.3100 (1995), *Generic network information model*.
- [4] ITU-T Recommendation M.3200 (1997), *TMN management services and telecommunications managed areas: overview*.
- [5] ITU-T Recommendation M.3400 (2000), *TMN management functions*.

- [6] ITU-T Recommendation X.720 (1992) | ISO/IEC 10165-1:1993, *Information technology – Open Systems Interconnection – Structure of management information: Management information model.*
- [7] ITU-T Recommendation X.721 (1992) | ISO/IEC 10165-2:1992, *Information technology – Open Systems Interconnection – Structure of management information: Definition of management information.*
- [8] ITU-T Recommendation X.731 (1992) | ISO/IEC 10164-2:1993, *Information technology – Open Systems Interconnection – Systems Management: State management function.*
- [9] ITU-T Recommendation X.733 (1992) | ISO/IEC 10164-4:1992, *Information technology – Open Systems Interconnection – Systems Management: Alarm reporting function.*
- [10] ITU-T Recommendation X.734 (1992) | ISO/IEC 10164-5:1993, *Information technology – Open Systems Interconnection – Systems Management: Event report management function.*
- [11] ITU-T Recommendation X.735 (1992) | ISO/IEC 10164-6:1993, *Information technology – Open Systems Interconnection – Systems Management: Log control function.*
- [12] ITU-T Recommendation X.739 (1993) | ISO/IEC 10164-11:1994, *Information technology – Open Systems Interconnection – Systems Management: Metric objects and attributes.*
- [13] ITU-T Recommendation X.792 (1999), *Configuration audit support function for ITU-T applications.*

3 Terms and definitions

No new definitions are needed in this Recommendation.

4 Abbreviations

This Recommendation uses the following abbreviations:

ASAP	Alarm Severity Assignment Profile
CM	Configuration Management
CMIP	Common Management Information Protocol
CORBA	Common Object Request Broker Architecture
EFD	Event Forwarding Discriminator
EMS	Element Management System
EPON	Ethernet Passive Optical Network
FM	Fault Management
FS	Function Set
FTP	File Transfer Protocol
ID	Identifier
ITU-T	International Telecommunication Union – Telecommunication Standardization Sector
MO	Managed Object
NMS	Network Management System
OS	Operations System
PM	Performance Management

QoS	Quality of Service
SDH-DLC	Synchronous Digital Hierarchy – Digital Loop Carrier
TMN	Telecommunications Management Network
UML	Unified Modelling Language

5 General overview

The EMS shown in Figure 5-1 is used to manage the individual network elements supporting some specific network technologies. One or more systems may be required depending on the different supplier products and geographic distribution of the elements in the network. NMS, the network layer management system shown in Figure 5-1, represents an integrated management OS across the different technology and supplier systems. The logical representation shown by NMS may be realized by one or more physical interfaces. Figure 5-1 shows the Q interface addressed in this Recommendation.

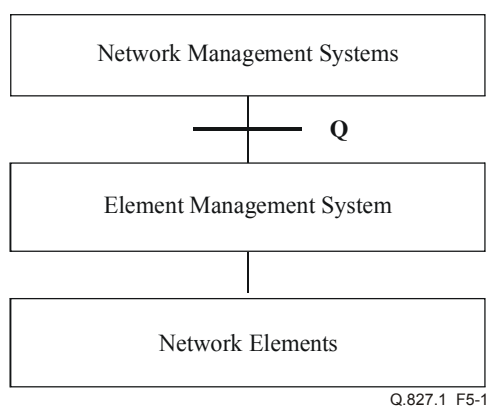


Figure 5-1/Q.827.1 – Reference interface

6 Requirements

6.1 Business level requirements

This Recommendation only focuses on the management interface between NMS and EMS and the interface management functions associated with them. Through the interface, NMS can query and modify configuration information, and EMS should report changes in configuration, performance data, and fault information to NMS.

6.1.1 Actor roles

The actor in this Recommendation is NMS. NMS is the network management system that manages the whole network system by interacting with suppliers' EMSs.

6.1.2 Telecommunications resources

Both the suppliers' EMSs and the managed network equipments are viewed as relevant telecommunications resources in this Recommendation.

6.1.3 High-level use-case diagrams

This clause contains the high-level UML use-case diagrams that summarize the functionalities and interfaces of EMS.

The first overview use-case diagram (Figure 6-1) shows the main management function sets (FS) involved in the NMS-EMS management interface.

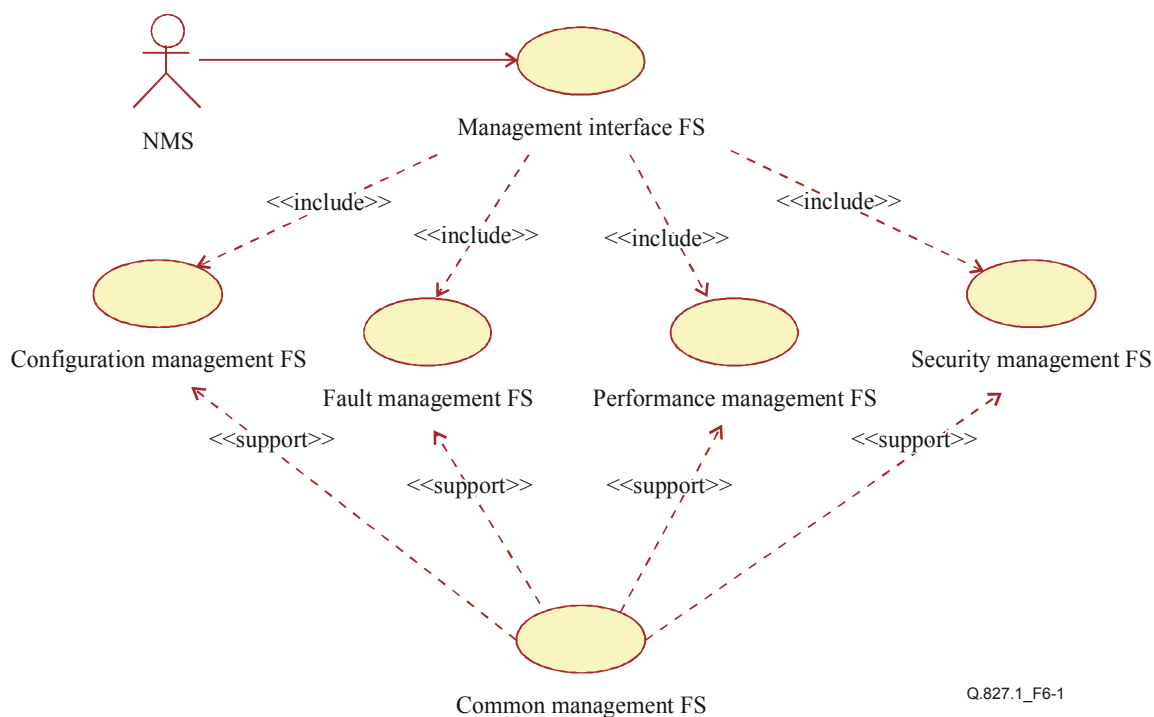


Figure 6-1/Q.827.1 – NMS-EMS management interface function set

The following four use-case diagrams (Figures 6-2 to 6-5) show the common management function sets, which are: Common management function set overview (Figure 6-2), Notification management function set (Figure 6-3), Log management function set (Figure 6-4), and Bulk data transfer control function set (Figure 6-5).

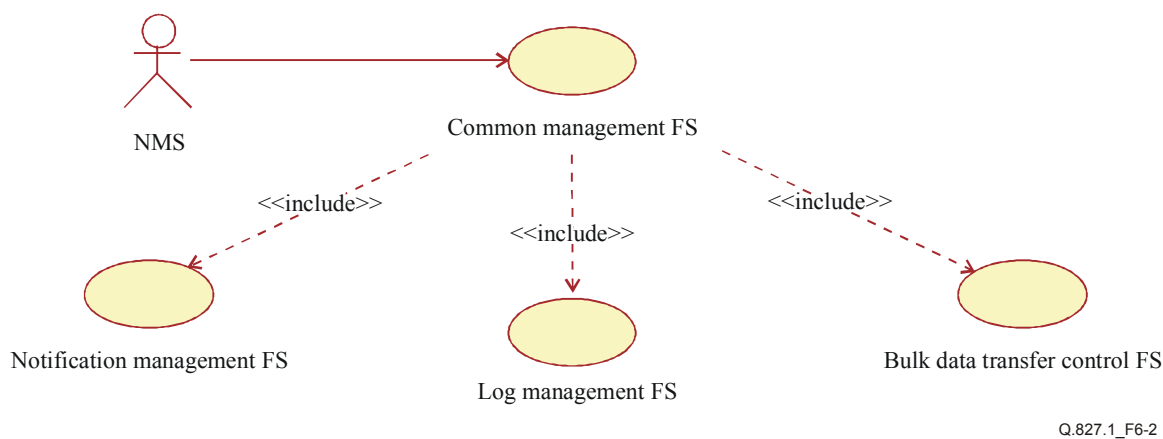


Figure 6-2/Q.827.1 – Common management function set overview

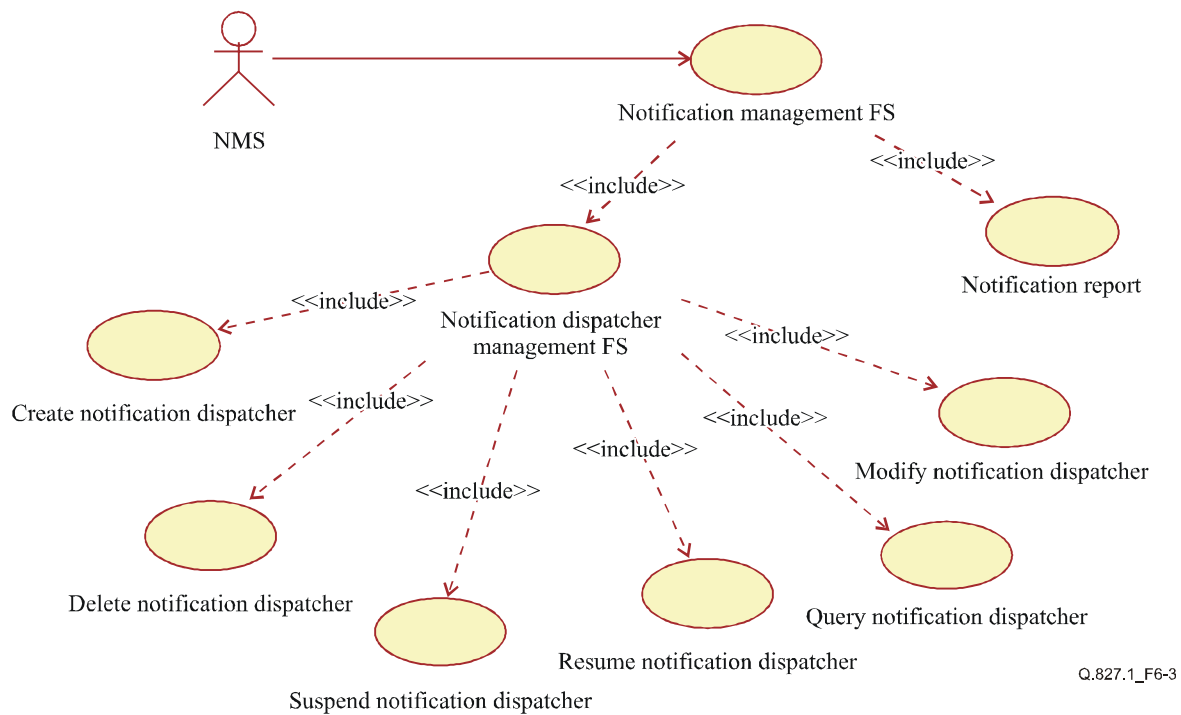


Figure 6-3/Q.827.1 – Notification management function set

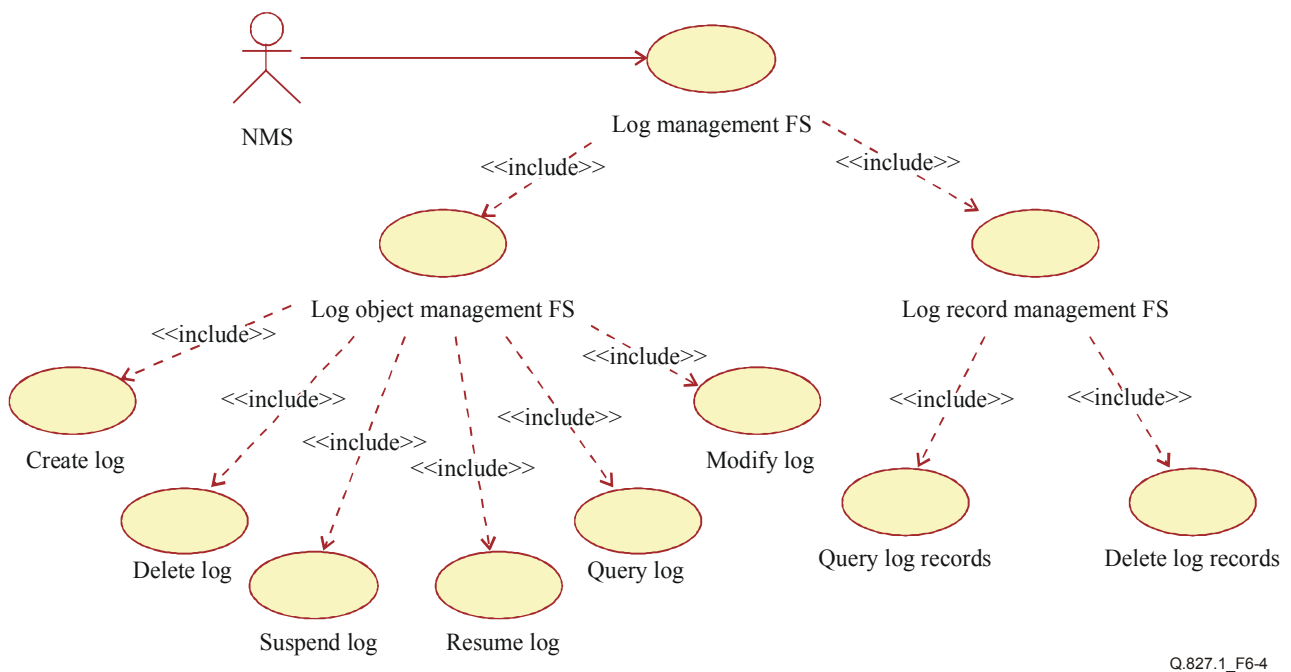


Figure 6-4/Q.827.1 – Log management function set

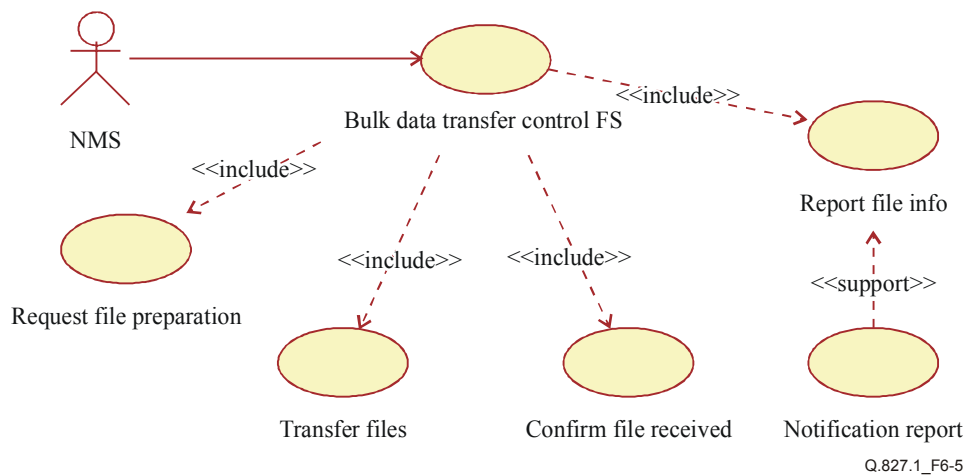


Figure 6-5/Q.827.1 – Bulk data transfer control function set

Configuration management function set is network technology dependent, so it will not be addressed in this Recommendation.

The following three diagrams show the functions involved in the performance management function set. Figure 6-6 illustrates performance management function set, Figure 6-7 is the performance measurement management function set use-case diagram, and Figure 6-8 is the threshold management function set use-case diagram.

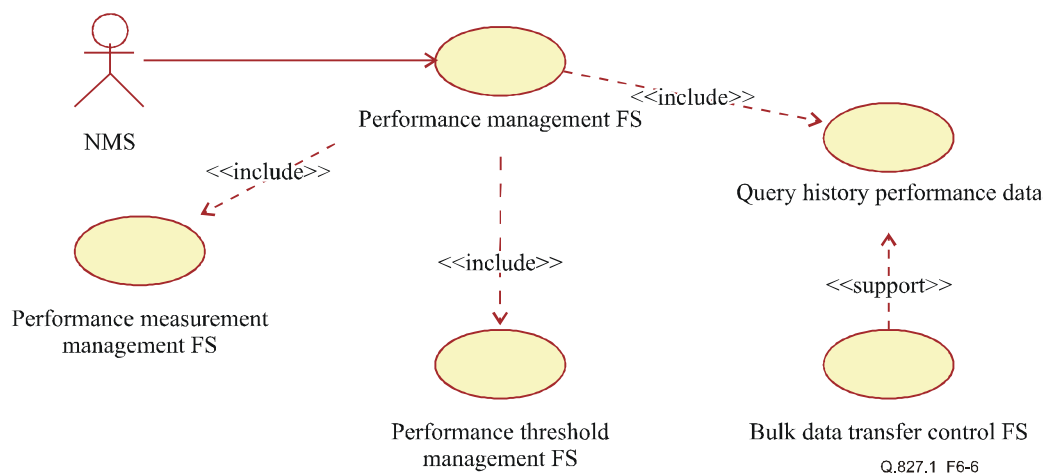


Figure 6-6/Q.827.1 – Performance management function set

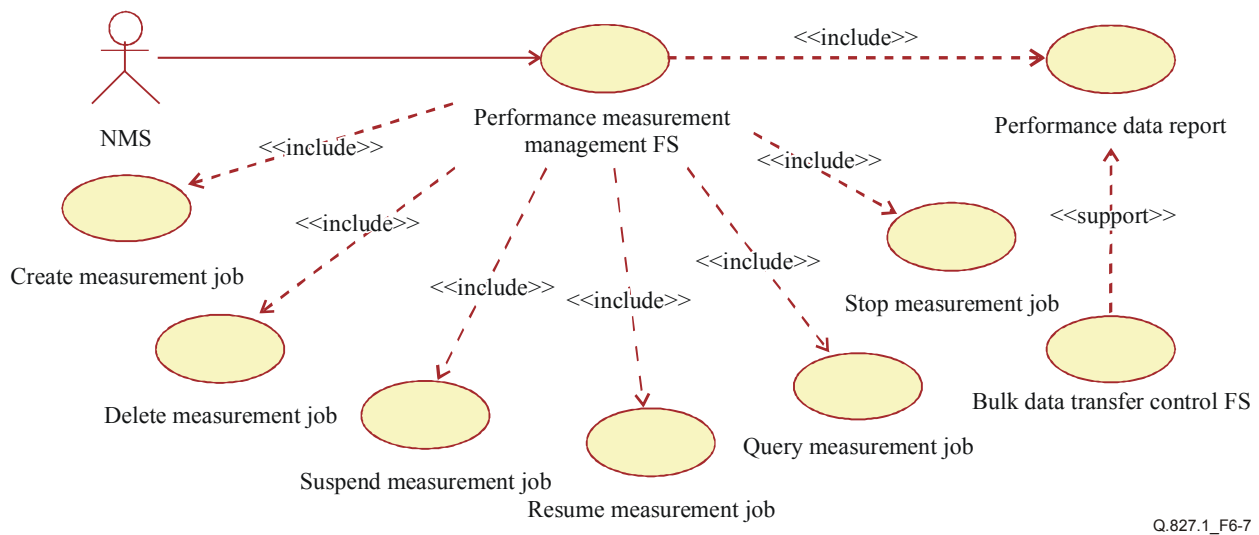


Figure 6-7/Q.827.1 – Performance measurement management function set

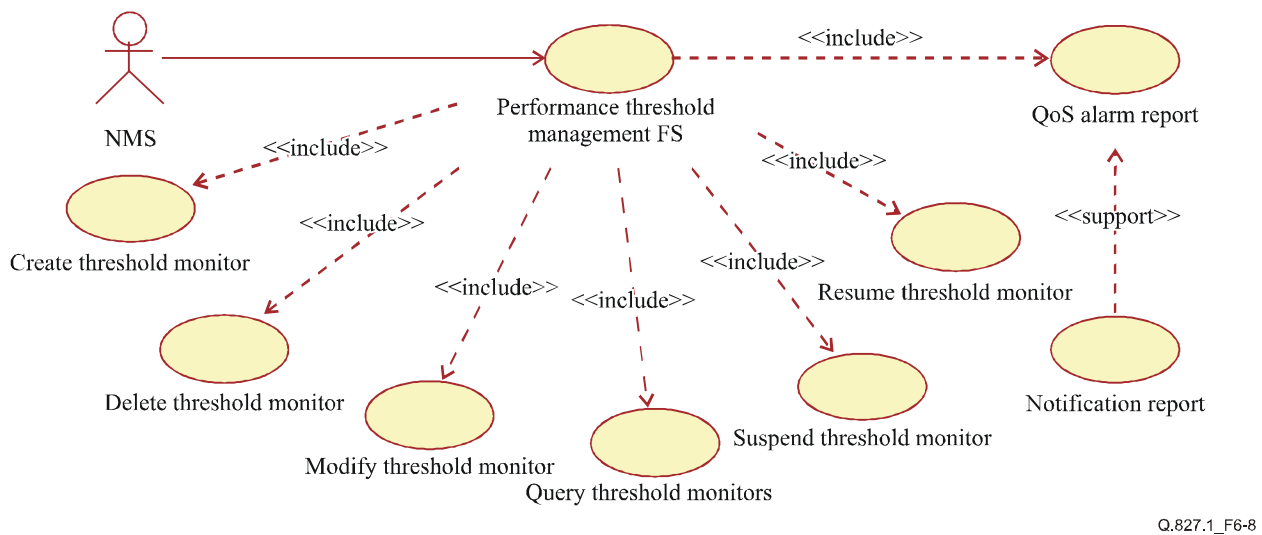
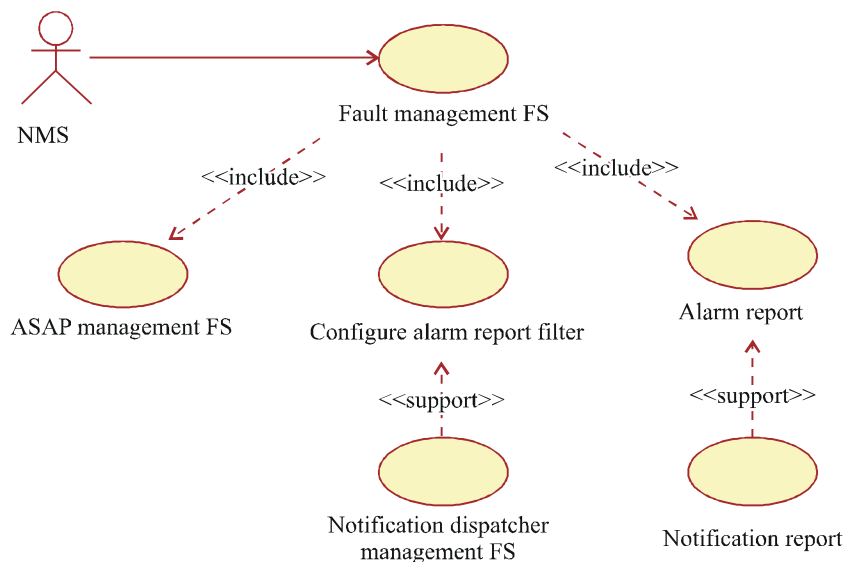


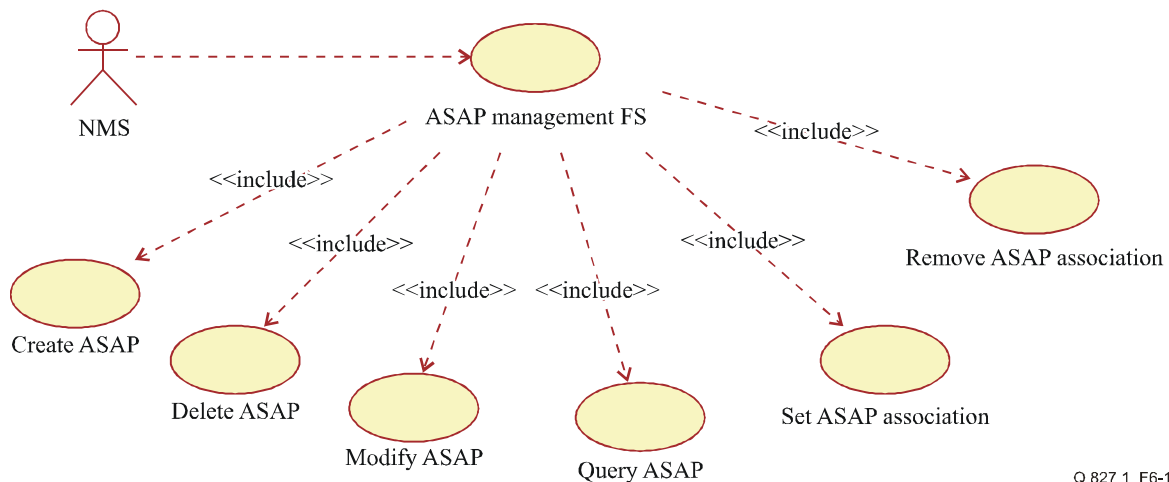
Figure 6-8/Q.827.1 – Performance threshold management function set

The following two use-case diagrams (Figures 6-9 and 6-10) show the functions involved in fault management function set.



Q.827.1_F6-9

Figure 6-9/Q.827.1 – Fault management function set



Q.827.1_F6-10

Figure 6-10/Q.827.1 – ASAP management function set

6.2 Specification level requirements

This clause contains textual details for each of the use cases shown in the high-level use-case diagrams of the previous clause. The details are provided to clarify the roles of external actors and telecommunications resources, to establish the basis for interactive diagrams in the analysis part, and to refine the previous high-level use-case diagrams to a specification level. Use-case details include the following components:

- **Summary:** short summary of use-case functionality referencing TMN functionality as needed.
- **Assumptions:** listing of requirements surrounding the use case that would affect the design of the application code of a supplier's EMS.
- **Actors:** actors which are described in 6.1.1.
- **Preconditions:** identifies the trigger for the use-case commencement.
- **Description:** detailed textual rendition of the functionality of the use case, including stops where exceptions can occur.

- **Exceptions:** identifies unsuccessful completion circumstances for the use case.
- **Post-conditions:** identifies the conditions that will hold if the use case ends successfully.

The use-case details are listed by the use-case title shown in the use-case diagrams of the previous clause.

6.2.1 Common management function set

6.2.1.1 Overview

The common management function set includes the notification management function set, log management function set and bulk data transfer control function set. Figure 6-2 shows the details.

The notification management function set can be divided into notification report function and notification dispatcher management function set. The latter also includes the following functions: create notification dispatcher, delete notification dispatcher, suspend notification dispatcher, resume notification dispatcher, modify notification dispatcher, and query notification dispatcher. Figure 6-3 shows the details.

The log management function set can also be divided into log object management function set, and log record management function set (see also ITU-T Rec. X.735 [11] for more information). Log object management FS includes create log, delete log, suspend log, resume log, modify log, and query log functions. Log record management FS includes query log records and delete log records functions. Figure 6-4 shows the details.

In the cases that the managed system is required to transfer bulk data, the bulk data transfer control function set will be used to improve the transport efficiency and reliability. This function set is involved in configuration management, fault management, performance management, and it may also be used in common management for querying log records.

The bulk data transfer control function set includes the following functions: request file preparation, transfer file, confirm file received and report file preparation information. Figure 6-5 shows the details.

The "Common management function set" here falls into the first context explained in clause 1 "Scope", including functions that are applicable for fault, performance, configuration and security management functions and can be achieved in the same mechanisms.

6.2.1.2 Notification management function set

6.2.1.2.1 Notification report

Summary: EMS can report notifications related to configuration, fault, performance or common management to NMS. See also ITU-T Rec. X.734 [10] for more descriptions.

Assumptions: The communication between EMS and NMS is available.

Actors: NMS.

Preconditions: An object has been created or deleted. An attribute value or state of an object has changed. Alarm occurs, or the bulk data file preparation is ready, or error occurs during bulk data file preparation.

Description: The information associated with configuration, fault and performance can be reported to NMS using the notification report function. The object creation notification, object deletion notification, attribute value change notification and state change notification are used in configuration management. Equipment alarm notification, environment alarm notification, communication alarm notification and processing error alarm notification are used in fault management. QoS alarm notification is used in performance management. Bulk data transfer ready and bulk data transfer preparation error notifications are used in common management.

Exceptions: Communication error.

Post-conditions: EMS will trigger notifications to the appropriate notification dispatcher(s), and those passing the corresponding filtering criteria will be forwarded to NMS.

6.2.1.2.2 Notification dispatcher management function set

6.2.1.2.2.1 Create notification dispatcher

Summary: NMS can send a request to EMS to create a notification dispatcher.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of creating a notification dispatcher.

Actors: NMS.

Preconditions: NMS needs to receive notifications from EMS.

Description: NMS can create a notification dispatcher through the management interface. In the case that EMS can initialize a notification dispatcher instance(s) by itself, it is not required to provide this function in the interface. (For example, the CORBA-based interface does not require the notification dispatcher creation function. As for the CMIP-based interface, this function is provided to create an EFD instance.) In this use case, NMS requests EMS to create a notification dispatcher. The parameters in the request include the destination(s), the initial administrative state and filtering criteria of the dispatcher.

If the dispatcher has been created successfully, EMS will then return the identifier of the notification dispatcher and may send a corresponding object creation notification to NMS. The subsequent notifications will be forwarded according to the attribute values of the dispatcher. If the creation fails, EMS will return error information. The possible errors are listed under "Exceptions".

Exceptions: Invalid parameter; EMS processing error; Communication error.

Post-conditions: The notification dispatcher is created by EMS at the request of NMS. EMS may send the corresponding object creation notification to NMS, and the newly created dispatcher acts on new notifications.

6.2.1.2.2.2 Delete notification dispatcher

Summary: NMS can delete a notification dispatcher through the management interface.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of deleting a notification dispatcher.

Actors: NMS.

Preconditions: NMS no longer needs to receive notifications from the specified notification dispatcher in EMS. The specified notification dispatcher is suspended.

Description: NMS can delete a notification dispatcher through the management interface. In the case that EMS can initialize a notification dispatcher instance(s) by itself, it is not required to provide this function in the interface (as in CORBA-based interfaces). In this use case, NMS requests EMS to delete a notification dispatcher. The parameter in the request is the ID of the notification dispatcher. If the dispatcher has been deleted successfully, EMS may then send a corresponding object deleted notification to NMS. If the deletion fails, EMS will return error information.

Exceptions: Unknown dispatcher; Dispatcher not suspended; EMS processing error; Communication error.

Post-conditions: The notification dispatcher is deleted by EMS on the request of NMS. EMS may send the corresponding object deletion notification to NMS. No more notifications will be forwarded to NMS by this deleted notification dispatcher.

6.2.1.2.2.3 Suspend notification dispatcher

Summary: NMS can suspend a notification dispatcher through the management interface. Thereafter, the dispatcher will not forward any notifications.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of suspending a notification dispatcher.

Actors: NMS.

Preconditions: The specified notification dispatcher exists in EMS and it is not suspended. NMS temporarily does not want to receive notifications from the dispatcher in EMS, or NMS needs to change the attribute values of the dispatcher.

Description: In this use case, NMS sends a request to EMS to suspend a notification dispatcher, that is, to set the administrative state of the dispatcher from "unlocked" to "locked". If the operation succeeds, the dispatcher will not forward any notifications until it is resumed. If the operation fails, EMS will return error information.

Exceptions: Dispatcher already suspended; EMS processing error; Communication error.

Post-conditions: The notification dispatcher is suspended by EMS on the request of NMS. EMS may send the corresponding state change notification to NMS, and the dispatcher does not forward notifications any more until it is resumed.

6.2.1.2.2.4 Resume notification dispatcher

Summary: NMS can resume a suspended notification dispatcher through the management interface, so that the dispatcher can forward notifications again.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of resuming a notification dispatcher.

Actors: NMS.

Preconditions: The specified notification dispatcher exists in EMS and it is suspended. NMS wants to receive notifications again from the specified dispatcher in EMS.

Description: In this use case, NMS sends a request to EMS to resume a suspended notification dispatcher, that is, to set the administrative state of the dispatcher object from "locked" to "unlocked". If the operation succeeds, the dispatcher will begin to forward notifications again. If the operation fails, EMS will return error information.

Exceptions: Dispatcher not suspended; EMS processing error; Communication error.

Post-conditions: The notification dispatcher is resumed by EMS on the request of NMS. EMS may send the corresponding state change notification to NMS, and the dispatcher acts on notifications again.

6.2.1.2.2.5 Modify notification dispatcher

Summary: NMS can modify the attribute values of a notification dispatcher through the management interface.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of modifying a notification dispatcher.

Actors: NMS.

Preconditions: NMS needs to change the filtering criteria of a notification dispatcher in EMS, or change the destination(s) of notifications to be forwarded by the dispatcher. The specified notification dispatcher exists in EMS and it is suspended.

Description: In this use case, NMS sends a request to EMS to modify a notification dispatcher. The filtering criteria and the destination(s) can be modified. If the modification succeeds, the dispatcher will forward notifications according to the new attribute values. If the modification fails, EMS will return error information.

Exceptions: Invalid parameter; Dispatcher not suspended; EMS processing error; Communication error.

Post-conditions: The notification dispatcher is modified by EMS on the request of NMS. EMS may send the corresponding attribute value change notification to NMS. The newly modified dispatcher then forwards notifications with the new filtering criteria to the new destination(s).

6.2.1.2.2.6 Query notification dispatcher

Summary: NMS can query the attribute values of a notification dispatcher in EMS.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of querying a notification dispatcher.

Actors: NMS.

Preconditions: NMS needs to query the attribute information of a notification dispatcher in EMS. The specified notification dispatcher exists in EMS.

Description: NMS can query the attribute values of a notification dispatcher such as administrative state, filtering criteria, destinations and so on. In this use case, NMS sends a request to query a notification dispatcher. The parameters in the request include the names of the attributes to be queried. If the function succeeds, EMS will return the specified attribute values. Otherwise, EMS will return error information.

Exceptions: Invalid parameter; EMS processing error; Communication error.

Post-conditions: EMS returns the attribute values requested by NMS.

6.2.1.3 Log management function set

6.2.1.3.1 Log object management function set

6.2.1.3.1.1 Create log

Summary: NMS can create a log through the management interface.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of creating a log.

Actors: NMS.

Preconditions: NMS needs EMS to log event information so that the information may be queried in the future for some specific situations such as data loss.

Description: NMS can create a log through the management interface. In the case that EMS can initialize a log instance(s) by itself, it is not required to provide this function in the interface. In this use case, NMS requests EMS to create a log. The parameters in the request include the initial administrative state, max log size, log full action, capacity alarm threshold, and filtering criteria of the log object.

If the log has been created successfully, EMS will then return the identifier of the log and send an object creation notification to NMS. Whether events will be recorded as log records depends on the filtering criteria defined in the log. If the creation fails, EMS will return error information.

Exceptions: Invalid parameter; EMS processing error; Communication error.

Post-conditions: The log is created by EMS on the request of NMS. EMS returns the identifier of the log and may send the corresponding object creation notification to NMS. The events will be recorded according to the filtering criteria in the newly created log.

6.2.1.3.1.2 Delete log

Summary: NMS can delete a log through the management interface.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of deleting a log.

Actors: NMS.

Preconditions: The specified log exists in EMS and it is suspended. NMS does not need the log in EMS to record any event information.

Description: NMS can delete a log through the management interface. In the case that EMS system can initialize a log instance(s), it is not required to provide this function in the interface. In this use case, NMS requests EMS to delete a log. The parameter in the request is the ID of the log object. If the log has been deleted successfully, EMS may then send the corresponding object deletion notification to NMS. If the deletion fails, EMS will return error information.

Exceptions: Unknown log, Log not suspended; EMS processing error; Communication error.

Post-conditions: The log and the associated log records are deleted by EMS on the request of NMS. EMS may send the corresponding object deletion notification to NMS.

6.2.1.3.1.3 Suspend log

Summary: NMS can suspend a log through the management interface. Thereafter, the log will not record any events until it is resumed.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of suspending a log.

Actors: NMS.

Preconditions: NMS temporarily needs to suspend a log in EMS. The specified log exists in EMS and it is not suspended. NMS temporarily does not want the log to record event information in EMS, or NMS needs to change the attribute values of the log object. The specified log exists in EMS.

Description: In this use case, NMS sends a request to EMS to suspend a log, that is, to set the administrative state of the Log object from "unlocked" to "locked". If the operation succeeds, the log object will not record any events until it is resumed. If the operation fails, EMS will return error information.

Exceptions: Log already suspended; EMS processing error; Communication error.

Post-conditions: The log is suspended by EMS on the request of NMS. EMS may send the corresponding state change notification to NMS, and the log does not record any events until it is resumed.

6.2.1.3.1.4 Resume log

Summary: NMS can resume a suspended log through the management interface, so that the log can record events again.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of resuming a log.

Actors: NMS.

Preconditions: The specified log exists in EMS and it is suspended. NMS wants the log to record events again in EMS.

Description: In this use case, NMS sends a request to EMS to resume a suspended log, that is, to set the administrative state of the log object from "locked" to "unlocked". If the operation succeeds, the log object will begin to record events again. If the operation fails, EMS will return error information.

Exceptions: Log not suspended; EMS processing error; Communication error.

Post-conditions: The log is resumed by EMS on the request of NMS. EMS may send the corresponding state change notification to NMS. The log continues to record events according to its filtering criteria.

6.2.1.3.1.5 **Modify log**

Summary: NMS can modify the attribute values of a log through the management interface.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of modifying a log.

Actors: NMS.

Preconditions: The specified log exists in EMS and it is suspended. NMS needs to change the attribute values of a log in EMS, such as the filtering criteria of events logging, the log full action, the capacity alarm threshold, and so on.

Description: In this use case, NMS sends a request to EMS to modify the attribute values of a log. The input parameters are the attribute names to be modified and the corresponding new values. The attributes that can be modified include: max log size, log full action, capacity alarm threshold, and filtering criteria. If the modification succeeds, the log will use new criteria to record events and act according to these new attribute values. If the modification fails, EMS will return error information.

Exceptions: Invalid parameter; Log not suspended; EMS processing error; Communication error.

Post-conditions: The log is modified by EMS on the request of NMS. EMS may send the corresponding attribute value change notification to NMS. The newly modified log then begins to record events with the new criteria and behaves according to the new attribute values.

6.2.1.3.1.6 **Query log**

Summary: NMS can query the attribute values of a log in EMS.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of querying a log.

Actors: NMS.

Preconditions: NMS needs to retrieve the attribute information of a log in EMS. The specified log exists in EMS.

Description: NMS can query the attribute values of a log, including administrative state, operational state, max log size, current log size, log full action, number of records, alarm status, capacity alarm threshold, filtering criteria, and so on. In this use case, NMS sends a request to query a log. The parameters in the request are the attribute names to be queried. If the operation succeeds, EMS will return the corresponding attribute values. Otherwise, EMS will return error information.

Exceptions: Invalid parameter; EMS processing error; Communication error.

Post-conditions: EMS returns the attribute values of the log requested by NMS.

6.2.1.3.2 Log record management function set

6.2.1.3.2.1 Query log records

Summary: NMS can query the log records contained in a log according to some criteria.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of querying log records.

Actors: NMS.

Preconditions: In the case that the notification information between NMS and EMS is not synchronized due to broken communication, abnormal data lost or other reasons, NMS needs to query log records in EMS to synchronize the notification information between two systems. The specified log exists in EMS.

Description: In this use case, NMS sends a "query log record" request to EMS. The parameters in the request are the filtering criteria and the time boundary. If the query succeeds, EMS will return the log records that satisfy the criteria and the time boundary. Otherwise, if the query fails, EMS will return error information.

Exceptions: Invalid parameter; No such log records; EMS processing error; Communication error.

Post-conditions: The corresponding log records are returned by EMS.

6.2.1.3.2.2 Delete log records

Summary: NMS can delete log records contained in a log according to certain criteria.

Assumptions: The communication between NMS and EMS is available. EMS supports the function of deleting log records.

Actors: NMS.

Preconditions: In the case where the information in some log records is of no use to NMS (too old or having been transferred to NMS), NMS can delete some or all log records in EMS. The specified log exists in EMS.

Description: In this use case, NMS sends a deletion request to EMS. The parameters in the request are the filtering criteria and the time boundary. If the deletion succeeds, the corresponding records that satisfy the criteria and the time boundary will be deleted. Otherwise, if the deletion fails, EMS will return error information.

Exceptions: Invalid parameter; No such log records; EMS processing error; Communication error.

Post-conditions: The log records that satisfy the filtering criteria and the time boundary are deleted.

6.2.1.4 Bulk data transfer control function set

6.2.1.4.1 Request file preparation

Summary: NMS can send a request to EMS to start a bulk data file preparation task.

Assumptions: The communication between NMS and EMS is available. EMS supports the FTP service.

Actors: NMS.

Preconditions: NMS needs to synchronize configuration audit data, performance history data, or log records information through the bulk data file transfer in EMS.

Description: NMS can request EMS to start a file preparation task using this function. On receiving this request, EMS will start to prepare the file(s) to be transferred. When EMS is ready for file transfer, it will send a "Bulk Data Transfer Ready" notification to NMS. If any exceptions occur during the preparation, EMS will send a "Bulk Data Transfer Preparation Error" notification to



NMS. In this use case, NMS sends a request to EMS to start a file preparation task. The parameters in the request include the file type, object selection (optional), job identifier (optional) and time boundary (optional). The file type may be one of configuration audit file, performance data file or log record file. For CM audit, and log record files, the object selection shall be specified to locate the managed object(s) for retrieving information from. For PM files, the identifier of the measurement job shall be specified. For PM and Log files, the time boundary can be applied. EMS then processes the request. If this operation succeeds, a file transfer ID will be returned to NMS for future uses. If the file has been prepared according to the request parameters, EMS will send a "Bulk Data Transfer Ready" notification to NMS. If the preparation fails, EMS will send a "Bulk Data Transfer Preparation Error" notification to NMS along with the possible causes.

Exceptions: Invalid parameter; EMS processing error; Communication error.

Post-conditions: EMS starts a file preparation task on the request of NMS. When the preparation is finished, EMS will send a "Bulk Data Transfer Ready" notification to NMS.

6.2.1.4.2 Transfer files

Summary: NMS retrieves the files from EMS.

Assumptions: The communication between NMS and EMS is available. EMS supports FTP service.

Actors: NMS.

Preconditions: NMS receives a "Bulk Data Transfer Ready" notification from EMS.

Description: Files can be transferred from EMS to NMS using FTP service. When NMS receives a "Bulk Data Transfer Ready" notification, it will begin to get the data files from EMS using FTP.

This function may actually be achieved by FTP lower level operations (such as FTP "get" operation), and no application level operation is needed for this function.

Exceptions: None (no application level exceptions).

Post-conditions: NMS gets the data files from EMS according to the file information specified in the received "Bulk Data Transfer Ready" notification. When file transfer finishes, NMS should check integrity and correctness of the received files.

6.2.1.4.3 Confirm file received

Summary: After data files have been transferred from EMS to NMS, NMS will inform EMS for confirmation.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: Data files for a transfer ID have been transferred from EMS to NMS.

Description: After the data files have been transferred from EMS to NMS, NMS will send a confirmation to EMS. The confirmation includes the file transfer ID and the status for each file. The status can be one of the following cases:

- 1) the specified file is successfully received and the syntax checking is OK;
- 2) the specified file is not found;
- 3) the specified file is received, but the file format is invalid; or
- 4) the specified file is received, but it is invalid for some other reasons.

When receiving this confirmation, EMS may then deal with the files without problems, for example, delete or move them to some other places. When the confirmation indicates there are some files not found or invalid, EMS should re-trigger the file preparation process for these files.

Exceptions: Unknown transfer ID; EMS processing error; Communication error.

Post-conditions: After receiving the confirmation, EMS can deal with the successfully confirmed files.

6.2.1.4.4 Report file information

Summary: The bulk data transfer control function set uses the notification report function described in the common management function set. The used notifications are the "Bulk Data Transfer Ready" notification and the "Bulk Data Transfer Preparation Error". When the data have been collected and the files have been prepared, EMS will send the "Bulk Data Transfer Ready" notification to NMS. When some exception occurs during the preparation, EMS will send the "Bulk Data Transfer Preparation Error" notification to NMS. Please refer to 6.2.1.2.1, "Notification Report" function in the Common management function set for details.

6.2.2 Configuration management function set

Configuration management function set is network technology dependent, and thus, it is outside the scope of this Recommendation.

6.2.3 Performance management function set

6.2.3.1 Overview

The performance management function set contains: Performance measurement management function set, Performance threshold management function set, and Query history performance data function, as depicted in Figure 6-6.

Performance measurement is the activity whereby the EMS periodically collects the performance data from physical equipments as well as logical entities and reports them to the NMS. Performance measurement management function set is provided to the NMS to manage the parameters related to performance measurement, through which performance data can be reported by the EMS according to the requests of the NMS.

Performance measurement management FS contains the following functions: Create measurement job, Delete measurement job, Stop measurement job, Suspend measurement job, Resume measurement job, Query measurement jobs, and Performance data report. Figure 6-7 shows the details.

(NOTE – The modify measurement job function can actually be achieved by deleting the existing measurement job and create a new measurement job.)

Performance measurement management function set in this Recommendation relates to the functions involving PM data collection and reporting specified in 5.3.4 "Performance administration function set" and 5.2.9 "Performance monitoring data accumulation function set" of ITU-T Rec. M.3400 [5].

NMS can set up performance threshold monitors, through which a corresponding QoS alarm may be emitted whenever a performance threshold is crossed. Performance threshold management FS contains the following functions: Create threshold monitor, Delete threshold monitor, Modify threshold monitor, Query threshold monitor, Suspend threshold monitor, Resume threshold monitor and QoS alarm report. Figure 6-8 shows the details.

Performance threshold management function set in this Recommendation relates to the functions involving threshold management specified in 5.3.4 "Performance administration function set" and 5.4.9 "NE(s) performance characterization function set" of ITU-T Rec. M.3400.

Measurement data are network technology dependent and are thus outside the scope of this Recommendation.

6.2.3.2 Performance measurement management function set

6.2.3.2.1 Create measurement job

Summary: NMS can request EMS to create a performance measurement job through the management interface.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: NMS needs to collect the performance measurement data on some managed entities in order to monitor the network performance, or the performance measurement data can be used for optimizing the network resource in the future.

Description: NMS sends a request to EMS to start a performance measurement job. The parameters may contain: the ID or criteria for the managed entities in which performance data will be collected, the start time of the collection task (optional), the stop time of the collection task (optional), the collection interval of the job, the report interval of the job, the schedule of the job (optional), and the performance parameters to be collected (optional). If the measurement job is started successfully, the job ID will be returned to NMS, and EMS will start the performance collection on the specified network resources according to the parameters of the request. Performance data files are stored in files, and on each reporting interval, the file information will be reported to NMS. Otherwise, it will return error information to NMS.

Exceptions: Invalid parameter; EMS processing error; Communication error.

Post-conditions: A measurement job is started on the request, and it starts to collect and report the corresponding performance data according to their intervals respectively. EMS may send an object creation notification to NMS.

6.2.3.2.2 Delete measurement job

Summary: NMS can request EMS to delete a performance measurement job through the management interface. When a measurement job is deleted, the associated measurement data files are not required to be maintained in EMS.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: NMS does not need the measurement job to collect performance data from the specified managed entities. The history performance data files related to the job in EMS will no longer be used. The specified measurement job exists in EMS and it is stopped or suspended.

Description: NMS sends a request to EMS to delete a performance measurement job. The request parameter is the identifier of the measurement job. If the operation succeeds, the specified measurement job will stop working and the related collecting resources, including the data files, will be released, and EMS will return success information. Otherwise, it will return error information to NMS.

Exceptions: Unknown measurement job; Measurement job not suspended or stopped; EMS processing error; Communication error.

Post-conditions: The specified measurement job is deleted on the request. EMS may send an object deletion notification to NMS.

6.2.3.2.3 Suspend measurement job

Summary: NMS can request EMS to suspend a performance measurement job through the management interface.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: The specified measurement job exists in EMS and it is not suspended. NMS temporarily does not want the measurement job to collect or report performance data.

Description: NMS can request EMS to suspend a performance measurement job through the management interface. If the operation succeeds, the measurement job will no longer collect and report the corresponding performance data until resumed.

Exceptions: Measurement job already suspended; EMS processing error; Communication error.

Post-conditions: The measurement job is suspended on request, and no performance measurement data associated with this job is collected or reported. EMS may send a state change notification to NMS.

6.2.3.2.4 Resume measurement job

Summary: NMS can request EMS to resume a suspended performance measurement job through the management interface.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: The specified measurement job exists in EMS and it is suspended. NMS wants the measurement job to collect and report performance data again.

Description: NMS can resume a suspended performance measurement job through the management interface. If the resuming operation succeeds, EMS will return success information and continue to collect and report performance data specified for this job. If the operation fails, it will return error information to NMS.

Exceptions: Measurement job not suspended; EMS processing error; Communication error.

Post-conditions: A measurement job is resumed on request, and it continues to collect and report performance measurement data. EMS may send a state change notification to NMS.

6.2.3.2.5 Query measurement job

Summary: NMS can request EMS to query the parameter values of a performance measurement job through the management interface.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: NMS needs to query the attribute information of a performance measurement job in EMS. The specified measurement job exists in EMS.

Description: NMS sends a request to EMS to query the parameters of a performance measurement job, which include: the job ID(s), the identifier(s) for the managed objects in which performance data are collected, the start time and stop time of the measurement job, the collection interval of the job, the report interval of the job, the schedule for the job, the administrative state, and the performance parameters to be collected. If the operation succeeds, EMS will return the attribute information of the performance measurement job. If the operation fails, it will return error information to NMS.

Exceptions: EMS processing error; Communication error.

Post-conditions: The corresponding attribute information is returned by EMS as requested.

6.2.3.2.6 Stop measurement job

Summary: NMS can request EMS to permanently stop a performance measurement job through the management interface. When a measurement job is stopped, it does not collect performance data any more, but it still holds the performance data files.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: NMS does not need the measurement job to collect performance data from the specified managed entities. The history performance data files related to the job in EMS may still be used. The specified measurement job exists in EMS.

Description: NMS sends a request to EMS to stop a performance measurement job. If the operation succeeds, the specified measurement job will stop working and EMS will return success information. Otherwise, it will return error information to NMS.

Exceptions: EMS processing error; Communication error.

Post-conditions: The measurement job is stopped on request but the measurement data files are still maintained by this measurement job, which can be retrieved by NMS. EMS may send a state change notification to NMS.

6.2.3.2.7 Performance data report

Summary: Performance data are stored in files. At each report interval, the corresponding performance data file(s) will be prepared by EMS and a "Bulk Data Transfer Ready" notification will be sent to NMS, and then the prepared files will be transferred from EMS to NMS using FTP service. Performance data report function uses "Bulk data transfer function set". See 6.2.1.4.4 for details.

6.2.3.3 Performance threshold management function set

6.2.3.3.1 Create threshold monitor

Summary: NMS can request EMS to create a performance threshold monitor through the management interface.

Assumption: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: NMS needs to monitor some performance measurement parameters collected from managed entities in order to know whether or not there are any performance degradation or service-related performance problems in time. The measurement job(s) containing the measurement parameters to be monitored have been started.

Description: NMS sends a request to EMS to create a performance threshold monitor. The input parameters in the request contain the ID or criteria for the managed entities to be monitored, the monitoring granularity period, and a set of sequence of the name of the measurement parameter, corresponding threshold value, the notifyOnOff switch of alarm notifications, and the related alarm severity (optional). The output parameter is the threshold monitor ID. If the creation succeeds, EMS will return success information. If the operation fails, it will return error information to NMS.

Exceptions: Invalid parameter; EMS processing error; Communication error.

Post-conditions: A performance threshold monitor is created by EMS, and an object creation notification may be reported to NMS. EMS starts to monitor the performance parameters according to the specified threshold values. When a threshold value is crossed, a QoS alarm will be raised.

6.2.3.3.2 Delete threshold monitor

Summary: NMS can request EMS to delete a performance threshold monitor through the management interface.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: NMS does not need to monitor the specified performance measurement parameters in EMS. The specified threshold monitor exists in EMS and it is suspended.

Description: NMS sends a request to EMS to delete a performance threshold monitor. The parameter is the identifier of the performance threshold monitor. According to the request, EMS will delete the specified performance threshold monitor. If the deletion succeeds, EMS will return success information and no longer monitor the corresponding performance parameters. If the operation fails, it will return error information to NMS.

Exceptions: Unknown Threshold Monitor; Threshold Monitor Not Suspended; EMS Processing Error; Communication Error.

Post-conditions: The specified performance threshold monitor is deleted by EMS, and an object deletion notification may be reported to NMS. EMS will no longer monitor the corresponding performance parameters. If the operation fails, it will return error information to NMS.

6.2.3.3.3 Suspend threshold monitor

Summary: NMS can suspend a performance threshold monitor through the management interface.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: The specified threshold monitor exists in EMS and it is not suspended. NMS temporarily does not want the threshold monitor to raise any QoS alarms for the performance parameters, or NMS needs to change some attribute values of the monitor.

Description: In this use case, NMS sends a request to suspend a performance threshold monitor. If the suspension succeeds, EMS will return success information and the performance threshold monitor will not act on the corresponding performance parameters and no QoS alarms on the performance parameters will be raised. If the operation fails, it will return error information to NMS.

Exception: Threshold monitor already suspended, EMS processing error, Communication error.

Post-conditions: The performance threshold monitor is suspended according to the request, and it does not monitor the corresponding performance parameters until resumed. A state change notification may be sent to NMS.

6.2.3.3.4 Resume threshold monitor

Summary: NMS can resume a suspended performance threshold monitor through the management interface.

Assumptions: The communication between NMS and EMS is available.

Actors: NMS.

Preconditions: The specified measurement job exists in EMS and it is suspended. NMS wants the threshold monitor to act on the performance parameters again.

Description: In this use case, NMS sends a request to resume a suspended performance threshold monitor. If the resumption succeeds, EMS will return success information and the performance