TS 101 251 V6.1.0 (1998-10)

Technical Specification

Digital cellular telecommunications system (Phase 2+); Fault management of the Base Station System (BSS) (GSM 12.11 version 6.1.0 Release 1997)

Reference
RTS/SMG-061211Q6R1 (bl0030c3.PDF)

Keywords
BSS, Fault Management

ETSI

Postal address
F-06921 Sophia Antipolis Cedex - FRANCE

Office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16
Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Internet
secretariat@etsi.fr
http://www.etsi.org

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available **free of charge** from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.org/ipr).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by the Special Mobile Group (SMG) of the European Telecommunications Standards Institute (ETSI)

This specification describes the fault management of any NE of a GSM PLMN through the Q3 interface, with focus on the Base Station System (BSS) within the digital cellular telecommunications system.

The contents of this TS is subject to continuing work within SMG and may change following formal SMG approval. Should SMG modify the contents of this TS, it will be re-released by SMG with an identifying change of release date and an increase in version number as follows:

Version 6.x.y

where:

6 indicates GSM Phase 2+ Release 1997;

x the second digit is incremented for all other types of changes, i.e. technical enhancements, corrections, updates, etc.;

y the third digit is incremented when editorial only changes have been incorporated in the specification.

Introduction

The present document specifies the requirements and model necessary for the standardised fault management (FM) aspects of operation, administration and maintenance (OAM) of a multivendor GSM PLMN.

The management of a GSM PLMN follows the systems management model outlined in ITU-T X.701 [5] which structures systems management into various aspects. The present document addresses the functional aspects of fault management.

Fault management provides the following facilities during the operation and maintenance phases of the PLMN under normal and failure conditions:

- installation and acceptance testing
- putting into service
- keeping the network operational.

The structure of this document has been defined taking into consideration two aspects:

1) The structured, top-down process of the standardisation. A general description of the 'Fault Management' of a Network Element (NE) is presented in clause 4. In clause 5, there is a specification of the four fault management 'Service Components' and, in clauses 6 and 7, the specification of the 'Functions' which the above service components are based on.

2) The reusability of this specification for other NEs of the PLMN. The requirements in clauses 4, 5 and 6 have been specified in a generic way (referring to a generic NE instead of a specific Base Station System) and all BSS specific management functions are specified separately in clause 7.

GSM fault management is based upon the context set by GSM 12.00. Principles, concepts and definitions are based on the M- and X-series of the ITU-T standards (with the exception of the Alarm Surveillance management functions based on ITU-T Q.821 and the Fault Correction management functions based on ITU-T G.774.03). Where the M-series of standards is not applicable, then the X-series is used as far as possible.

1 Scope

The present document describes the fault management of any NE of a GSM PLMN through the Q3 interface, with focus on the Base Station System (BSS).

The OAM of the GSM Public Land Mobile Network (PLMN) is organised and described in terms of TMN Management Services. GSM 12.00 describes the architecture and gives a general overview of the OAM services, while the rest of the GSM 12 series gives the detailed specification for each service and other aspects of the OAM.

Among all the TMN Management Services listed in GSM 12.00, the following are addressed by this specification:

- Management of the BSS (also covered by GSM 12.20).
- Restoration and Recovery (also covered by GSM 12.06).

The present document deals with "Fault Management" aspects of the above services.

For the TMN Management Services covered, the following "TMN Management Service Components" have been defined in clause 5:

- Alarm surveillance.
- Fault localisation.
- Fault correction.
- Testing.

The above TMN Management Service Components are based on several "Management Functions", some of which are defined in other ITU-T documents (e.g. State Management Functions, Alarm Reporting Functions, etc.), and others which are specifically defined here, in clauses 6 and 7.

For some management functions, the management information model is already provided by referenced ITU-T recommendations as well as GSM 12.00 and GSM 12.20.

The GDMO and ASN.1 definition of the information model for the present document is fully defined through explicit references to ITU-T M.3100, ITU-T Q.821, ITU-T G.774.03, ITU-T X.700-series and GSM 12.00 information models, together with extensions defined in the Annex of the present document.

Although sometimes considered as part of fault management, various administrative policies and procedures such as trouble ticketing and tracking, parts inventory, etc. are not included in the present document. Such aspects may be considered to be the responsibility of the operator and thus outside the scope of the EN.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies.

- A non-specific reference to an ETS shall also be taken to refer to later versions published as an EN with the same number.

2.1 Normative references

The present document incorporates, by dated or undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to, or revisions of, any of these publications apply to the present document only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

[1] ITU-T Recommendation M.20 (1992) : "Maintenance Philosophy for Telecommunication Networks".

[2] ITU-T Recommendation M.3100 (1995): "Generic Network information Model".

[3] ITU-T Recommendation M.3200 (1992): "TMN Management Services: Overview".

[4] ITU-T Recommendation M.3400 (1992): "TMN Management Functions".

[5] ITU-T Recommendation X.701 (1992) | ISO/IEC 10040: " Information technology - Open Systems Interconnection - Systems Management Overview ".

[6] ITU-T Recommendation X.721 (1992) | ISO/IEC 10165-2: " Information technology - Open Systems Interconnection - Structure of management information: Definition of Management Information ".

[7] ITU-T Recommendation X.722 (1992) | ISO/IEC 10165-4: "Information technology - Open Systems Interconnection - Structure of management information: Guidelines for the Definition of Managed Objects".

[8] Not used.

[9] ITU-T Recommendation X.733 (1992) | ISO/IEC 10164-4: "Information technology - Open Systems Interconnection - Systems Management: Alarm Reporting Function".

[10] ITU-T Recommendation X.734 (1992) | ISO/IEC 10164-5: "Information technology - Open Systems Interconnection - Systems Management: Event Report Management Function".

[11] ITU-T Recommendation X.735 (1992) | ISO/IEC 10164-6: "Information technology - Open Systems Interconnection - Systems Management: Log Control Function".

[12] ITU-T Recommendation X.737 (1995) | (ISO/IEC 10164-14): "Information technology - Open Systems Interconnection - Systems Management: Confidence and Diagnostic Test Categories".

[13] ITU-T Recommendation X.745 (1994) | ISO/IEC 10164-12: "Information technology - Open Systems Interconnection - Test Management Function".

[14] ITU-T Recommendation X.751 (1995) | (ISO/IEC 10164-17): "Information technology - Open Systems Interconnection - Change Over Function".

[15] ITU-T Recommendation Q.821: (1993) "Q3 Interface for Alarm Surveillance".

[16] ETR 100 (GSM 01.04): Digital cellular telecommunications system (Phase 2+); "Abbreviations and acronyms".

[17] GSM 08.08: Digital cellular telecommunications system (Phase 2+);"Mobile Switching Centre (MSC) to Base Station System (BSS) Interface; Layer 3 Specification".

[18] GSM 12.00: Digital cellular telecommunications system (Phase 2); "Objectives and Structure of GSM PLMN Management".

[19] GSM 12.01: Digital cellular telecommunications system (Phase 2); "Common Aspects of GSM/DCS 1800 PLMN Management".

[20] GSM 12.04: Digital cellular telecommunications system (Phase 2);"Performance Management and Measurements for a GSM PLMN".

[21] GSM 12.06: Digital cellular telecommunications system (Phase 2); "GSM Network Configuration Management".

[22] GSM 12.20: Digital cellular telecommunications system (Phase 2); "Base Station System (BSS) Management Information".

[23] ITU-T Recommendation G.774.03 (1994): "Synchronous Digital Hierarchy (SDH) Management of Multiplex-Section Protection for the Network Element View".

2.2 Informative references

1) ITU-T Recommendation M.3010 (1992): "Principles for a Telecommunications Management Network".

2) ITU-T Recommendation X.710 (1991): "Common management information service definition for CCITT applications". ISO/IEC 9595: "Information technology - Open Systems Interconnection - Common management information service definition" is technically aligned with ITU-T X.710.

3) ITU-T Recommendation X.711 (1991): "Common management information protocol specification for CCITT applications" ISO/IEC 9596: "Information technology - Open Systems Interconnection - Common management information protocol specification " is technically aligned with ITU-T X.711.

4) ITU-T Recommendation X.725 (1995): | ISO/IEC 10165-7: "Information technology - Open Systems Interconnection - Structure of management information: General Relationship Model".

5) ITU-T Recommendation X.731 (1992): | ISO/IEC 10164-2: "Information technology - Open Systems Interconnection - Systems Management: State Management Function".

6) GSM 08.52: Digital cellular telecommunications system (Phase 2+); "Base Station Controller to Base Transceiver Station Interface Principles".

7) GSM 08.58: Digital cellular telecommunications system (Phase 2+); "BSC-BTS Layer 3 Specification".

8) GSM 08.60: Digital cellular telecommunications system (Phase 2); "Inband control of remote transcoders and rate adapters ".

9) GSM 12.21: Digital cellular telecommunications system (Phase 2); "Network Management Procedures and Messages On the A-bis Interface".

10) GSM 12.22: Digital cellular telecommunications system (Phase 2); "Interworking of GSM Network Management (NM) procedures and messages at the Base Station Controller (BSC)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following definitions apply:

Alarm: A notification, of the form defined by the alarm reporting function (ITU-T X.733 [9]), of a specific event.

Active resource: An active resource in the context of redundancy is equivalent to a primary resource.

Alarm report: A specific type of event report used to convey alarm information.

Outstanding alarm condition: The state in which the conditions that originated an alarm are still present in the system.

Anomaly: An anomaly is a discrepancy between the actual and desired characteristics of an item (ITU-T M.20 [1]). In the context of this specification, the item may also be external to the NE (e.g. : environmental alarm detector).

Asymmetric redundancy: A redundancy where the primary and secondary resources have different capabilities, and therefore cannot exchange their roles (where the secondary may take the primary role, but the primary may not take the secondary role). Once the faulty primary resource is repaired and restored to service a change back needs to be performed.

Back up: A back up resource is a secondary resource providing redundancy to a primary resource.

Backed up: A backed up resource is a primary resource which has a secondary resource providing redundancy.

Change-back: A change back is the reverse change over in an asymmetric redundancy to restore the resources into their original roles.

Change-over: Change over is the action within a system capable of supporting redundancy which results in a secondary resource taking over the primary role. In a symmetric redundancy, the primary resource may take the secondary role.

Cleared Alarm: An alarm notification with the perceived severity set to cleared.

Cold standby: A secondary resource that requires initialisation activity before it can provide backup capability is defined as being in a cold standby state (ITU-T X.751 [14]).

Counter: Counters are a management abstraction of an underlying process, which may be associated with a defined internal event in management information. The current value of a counter is incremented when this event occurs (see note). It can take any values in its range. When a counter reaches its maximum value, it wraps around and counts upwards from 0; overflow information is not in general retained. An additional notification may be defined for counters with wrap behaviour.

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

Counter Threshold: A counter threshold is the application of the thresholding mechanism to a counter.

Defect: A defect is a limited interruption in the ability of an item to perform a required function. It may or may not lead to maintenance action depending on the results of additional analysis (M.20 [1]). In the context of this specification, the item may also be external to the NE (e.g. : environmental alarm detector).

Duplex redundancy: A duplex redundancy is a redundancy in which a given function can be performed by two resources: a primary resource and secondary resource (also known as active and standby resources respectively).

Failure: A failure is the termination of the ability of an item to perform a required function (M.20 [1]). In the context of this specification, the item may also be external to the NE (e.g. : environmental alarm detector).

NOTE: After a failure, the item has a fault (M.20 [1]).

Fault: A fault is the inability of an item to perform a required function, excluding that inability due to preventive maintenance, lack of external resources or planned actions.

NOTE: A fault is often the result of a failure of the item itself, but may exist without prior failure (M.20 [1]).

Gauge: The gauge is the management abstraction of the value of a dynamic variable, such as the number of connections currently operated by a protocol machine or the rate of change of a traffic counter. There is no restriction on what the dynamic variable may represent, within the constraints set out below. The value of the gauge is subject to change in either direction. The value of the increment or decrement is unconstrained, except that a change that would take the gauge beyond its minimum or maximum value, will leave the gauge value at its minimum or maximum value respectively, until it is subsequently again within the gauge range values.

Gauge Threshold: A gauge threshold is the application of the thresholding mechanism to a gauge.

Hot standby: A secondary resource that is able to provide backup capability for a primary resource without the need for initialisation activity is defined as being in a hot standby state (X.751 [14]).

Least Replaceable Unit (LRU): The smallest piece of equipment that can be replaced by field service personnel.

N+K redundancy: A redundancy where a given function can be performed by N primary resources and K secondary resources. When a failure occurs on one of the N primary resources, one of the K secondary resources may take over the role of that faulty primary resource.

Primary object: A primary object is a managed object that represents a primary resource.

Primary resource: A primary resource (in a system capable of supporting redundancy) is a resource that is performing a given function. On failure of a primary resource a secondary resource may take over the role of the faulty primary resource. A primary resource may also be referred to as an "active" or "backed up" resource.

Primary role: A primary object being backed up in a redundancy relationship is defined as being in the primary role.

Redundancy: The capability of a system to perform fault tolerant functionality by means of spare resources (or groups of resources).

Secondary object: A secondary object is a managed object that represents a secondary resource.

Secondary resource: A secondary resource (in a system capable of supporting redundancy) is a resource that may back up a primary resource. A secondary resource may take over the role of a primary resource on failure of that resource. A secondary resource may also be referred to as a "standby" or "back up" resource.

Secondary role: A secondary object backing up in a redundancy relationship is defined as being in the secondary role.

Standby resource: A standby resource in the context of redundancy is equivalent to a secondary resource.

Symmetric redundancy: A redundancy where the primary and secondary resources have the same capabilities, and therefore each of the resources can exchange their roles (primary and secondary). Once the faulty resource is repaired and restored to service there is no need to perform a change back.

Thresholding: Thresholding is the general mechanism (based on counters or gauges) to generate a defined notification from numeric changes in the value(s) of a counter(s) or gauge(s). The defined notification is generated as a result of a value change crossing the threshold level of a counter or gauge.

Threshold Level: A threshold level is a value which a threshold mechanism compares with the value of a counter or gauge, to determine whether a defined notification is to be generated.

3.2 Abbreviations

For the purpose of the present document, the following abbreviations apply, in addition to those listed in GSM 01.04 [16]:

AO	Associated Object
ACSE	Association Control Service Element
ASN.1	Abstract Syntax Notation (number) 1
BSC	Base Station Controller
BSS	Base Station System
BTS	Base Transceiver Station
CASC	Current Alarm Summary Control
CMIS	Common Management Information Service
CMISE	Common Management Information Service Element
CMIP	Common Management Information Protocol
EFD	Event Forwarding Discriminator
FTAM	File Transfer Access and Management
GDMO	Guidelines for the Definition of Managed Objects
LRU	Least Replaceable Unit
MIB	Management Information Base
MOC	Managed Object Class
MORT	Managed Object Referring to Test
MOS	Management Operations Schedule
NE	Network Element
OAM	Operation, Administration and Maintenance
OMC	Operations and Maintenance Centre

OS	Operations System
OSI	Open Systems Interconnection
PLMN	Public Land Mobile Network
QOS	Quality of Service
ROSE	Remote Operation Service Element
SFTC	Simple File Transfer Control
TARR	Test Action Request Receiver
TMN	Telecommunications Management Network
TO	Test Object

4 General requirements on fault management

GSM fault management follows TMN principles as specified in GSM 12.00 [18]. These principles provide for management of a GSM PLMN based on an information model. This model is developed through the definition of a required set of management services which are then decomposed into service components. These service components are supported by a number of management functions according to M.3400 [4], and an information model is defined to support this set of functions.

In GSM 12.00, the terms reactive maintenance and proactive maintenance are used to identify two different forms of maintenance for a GSM network element. Reactive maintenance is the use of the above management functions to detect a fault and restore all or part of the network element following a failure. Proactive maintenance is the use of the above management functions and manual routine maintenance activities to prevent, as far as possible, the occurrence of a failure. The present document does not explicitly assign the above management functions to either proactive or reactive maintenance, and generically defines the management functions such that they may be used either as part of proactive or reactive maintenance. The remainder of the present document thus makes no further reference to either proactive or reactive maintenance.

Fault Management is a functional area which can support a number of management services (M.3400) [4]. The following list gives examples of the general objectives of the NE's fault management:

- Inform the operator and/or OS of the current NE condition;
- Provide timely and accurate data regarding any abnormal change in the condition of the NE;
- Maintain synchronisation between the actual conditions in the NEs and the knowledge of the conditions as understood by the OS;
- Provide procedures which allow system recovery either automatically or on operator demand after a fault detection.

To support these objectives the NE shall offer the following set of capabilities:

- Surveillance capability to monitor the system such that faults, defects and anomalies may be detected and reported;
- Fault Localisation capability to identify the one or more replaceable units at fault;
- Fault Correction capability to isolate the faulty units and restore the system to operation;
- Testing capability to verify the proper operation of physical and functional resources in the NE.

To support fault management, the state management capability may also be necessary, for example, to isolate a faulty unit by changing its administrative state, to provide a specific environment for testing etc. The usage of state management for fault management purposes will be described in each clause where it is appropriate.

In addition to the above, the capabilities of other functional management areas are often used in support of fault management. For example, parts of performance management services may be used for the fault detection capability (e.g.: counters and gauges for threshold management) and configuration management functions may be used to restore the system to the best operational configuration.

Based on M.3400 [4], GSM fault management requirements can be achieved by means of the following management service components:

- Alarm surveillance service components;
- Fault localisation service component;
- Fault correction service component;
- Testing service component;

- Trouble administration service component.

Of this list, the trouble administration component is not addressed by the present document as it is too closely related with the operator's operational procedures and thus is not suitable for standardisation.

The operator and the OS are informed of an NE failure by means of functions provided by the alarm surveillance component: the alarm reporting functions. The information provided by alarm reporting should be sufficient to localise the fault. However, if necessary, the operator may also use the testing capabilities to obtain further details for fault diagnosis. Depending on the type of the detected fault and its impact on the telecommunication services, the fault correction service component provides automatic or manual actions to configure the NE so as to minimise the loss of the telecommunication services. When the faulty unit(s) is repaired,the fault correction service component again provides automatic or manual actions to restore the previously faulty unit(s) to its normal operation. To complete the fault management process, the operator is able to perform a final test to certify the behaviour of the repaired system.

4.1 Overview of the service components

4.1.1 Alarm Surveillance service component

The Alarm Surveillance Service Component performs system monitoring and fault detection in near real time. When a failure occurs in an NE, an alarm record is stored in a log (depending on the filter criteria) and an alarm report is forwarded (depending on the filter and forwarding criteria) as soon as possible to the OS across the Q3 interface. The nature and severity of the faults are determined by the NE. It is important that alarm reports are not lost in case of temporary interruption of communication between the NE and the OS.

The Alarm Surveillance service component is mandatory, and the refinement of the mandatory and optional parts of this service component is further defined in subsequent clauses.

4.1.2 Fault Localisation service component

The objective of Fault Localisation is to identify the faulty unit by means of the information provided by the NE when it notifies the OS of the failure. If necessary, further identification by means of localisation routines (e.g.: tests controlled by the OS) can also be run to get more details.

The Fault Localisation service component is mandatory, and the refinement of the mandatory and optional parts of this service component is further defined in subsequent clauses.

4.1.3 Fault Correction service component

After the identification of the fault and the replaceable faulty units, support by the Fault Correction service component is necessary in order to perform system recovery and/or restoration, either automatically by the NE and/or the OS, or manually by the operator. The first fault correction action is the isolation of the faulty unit, to reduce the effect of the fault on other parts, internal or external to the NE.

The Fault Correction service component is optional. If implemented, at least the mandatory parts of this service component shall be provided as further defined in subsequent clauses.

4.1.4 Testing service component

The Testing service component provides support for the other three fault management capabilities. Testing can be carried out in two ways: uncontrolled and controlled by the management system, and can be performed through periodic scheduling or on demand. Several categories of tests are necessary to cover all the requirements.

The Testing service component is optional. If implemented, at least the mandatory parts of this service component shall be provided as further defined in subsequent clauses.

5 Fault management service components

According to the TMN scheme (see M.3200 [3] and M.3400 [4]), the general requirements on fault management described in the previous clauses are achieved by means of management service components. This clause contains a more detailed description of the requirements and defines the related management service components.

5.1 Alarm surveillance

This service component requires that the OS and the operator have a consistent and up-to-date view of the current operating condition and quality of service of the managed network elements. For efficient and accurate fault management of a PLMN, it is also essential to achieve early detection of faults so that they can be corrected before significant effects have been noticed by the end-user.

In support of this, the Alarm Surveillance functions are used to monitor and interrogate NEs about faults, defects and anomalies. This results in the following requirements:

a) All detected faults, defects and anomalies in the NE shall be reported to the OS (for each case which matches the reporting conditions set by the OS). The NE/OS shall therefore support the sending/reception of unsolicited event reports notifying such events.

   Whenever possible, the NE should generate a single notification for a single fault. When a single fault results in the failure of other functionalities, the NE should filter these "dependency faults". Such filtering is internal to the NE, and is not standardised in the present document.

   There are a number of possible mechanisms by which the NE can detect faults, defects and anomalies. Depending on the implementation within the NE, several sources of information may be available. The most obvious examples are:

   - hardware/firmware detectors: co-operating or supervising units which continuously check the correctness of analogue or digital signals (sense points, transmission error detectors);

   - software detectors: to detect run time software errors as well as equipment errors;

   - performance detectors: to monitor the performance of the NE and generate a quality of service alarm notification if it is out of normal range.

   Some of these detectors are manageable and some are not, depending on the nature of their implementation. Furthermore, some of the manageable aspects of these detection functions (such as managing of thresholds for quality of service alarms) are the subject of standardisation in the present document (these are described below), whilst others are outside of the scope.

b) The forwarding of alarm reports through the Q3 interface shall be manageable both in terms of filtering (which reports shall be forwarded and which shall be discarded) as well as in terms of report destination(s). The alarm reporting is based on the model described in X.734 [10], i.e. the result of the fault detection process shall be an alarm notification sent to the Event pre-processing which may generate a potential event report that is sent to all the existing EFDs. The EFD is a managed object which receives the potential alarm reports and determines which event reports are to be forwarded and which are to be discarded. For the forwarded reports it also determines the destination, the time frame and the forwarding mode (i.e. confirmed or non-confirmed).

c) The NE shall be able to log alarm information as alarm records and support later retrieval of the logged alarm records. The logging functionality is based on the model described in X.735 [11] and can be used for any type of event information, including the alarm information. According to this model, when a fault is detected, an alarm notification is also sent to the Log pre-processing which may generate a potential log report that is sent to all the existing logs. The Log is a managed object which receives the potential log reports and determines which of them are stored as log records and which are discarded.

d) The NE shall be able to provide information to the OS about all the current outstanding alarm conditions in the NE. The outstanding alarm information may be reported in a summary on demand or periodically to the OS. This functionality is based on the Q.821 [15] and can be used to obtain a view of the NE's current alarm condition on demand. This functionality can also be used to align alarm information between the OS and NE, for instance after

an interruption of communication between the OS-NE (e.g. link failure, OS restart, NE restart) without waiting for the forwarding of all the events which occurred during the failure.

To support these alarm surveillance requirements over the Q3 interface, a number of management functions are defined in more detail in clause 6. These functions are grouped as follows:

- Threshold Management functions (X.721 [6]);
  The set of threshold management functions is optional.

- Alarm Reporting functions (M.3400 [4]/Q.821[15]/X.733 [9]/X.734[16]/GSM 12.00 [18]);
  The set of alarm reporting functions is mandatory.

- Log control functions (M.3400 [4]/Q.821[15] X.735 [11]);
  The set of log control functions is optional.

- Alarm Summary functions (M.3400 [4]/Q.821[15]).
  The set of alarm summary functions is optional.

Alarm surveillance also has implications for the operating practices for co-ordination whenever multiple OSs are involved in management operations. Such practices are the responsibility of the operator and are not standardised in the present document.

5.1.1 The model

The model adopted for Alarm Surveillance is depicted in figure 1. In this figure, the circles represent managed objects, stacks of circles represent sets of managed objects, and the names of the MOC(s) are reported in the middle of the circles.

The leftmost stack of managed objects in figure 1 represents the managed objects of the NE that can generate alarm notifications (for example, in the case of a BSS, instances of the GSM 12.20 MOCs and their subclasses).

The solid arrows represent the information flow and the dotted ones the control flow. Only the flows related to the Alarm Surveillance are represented and are described subsequently.

The "Event/Log pre-processing" is an NE internal function (implementation dependent) and is not subject to specification in the present document.

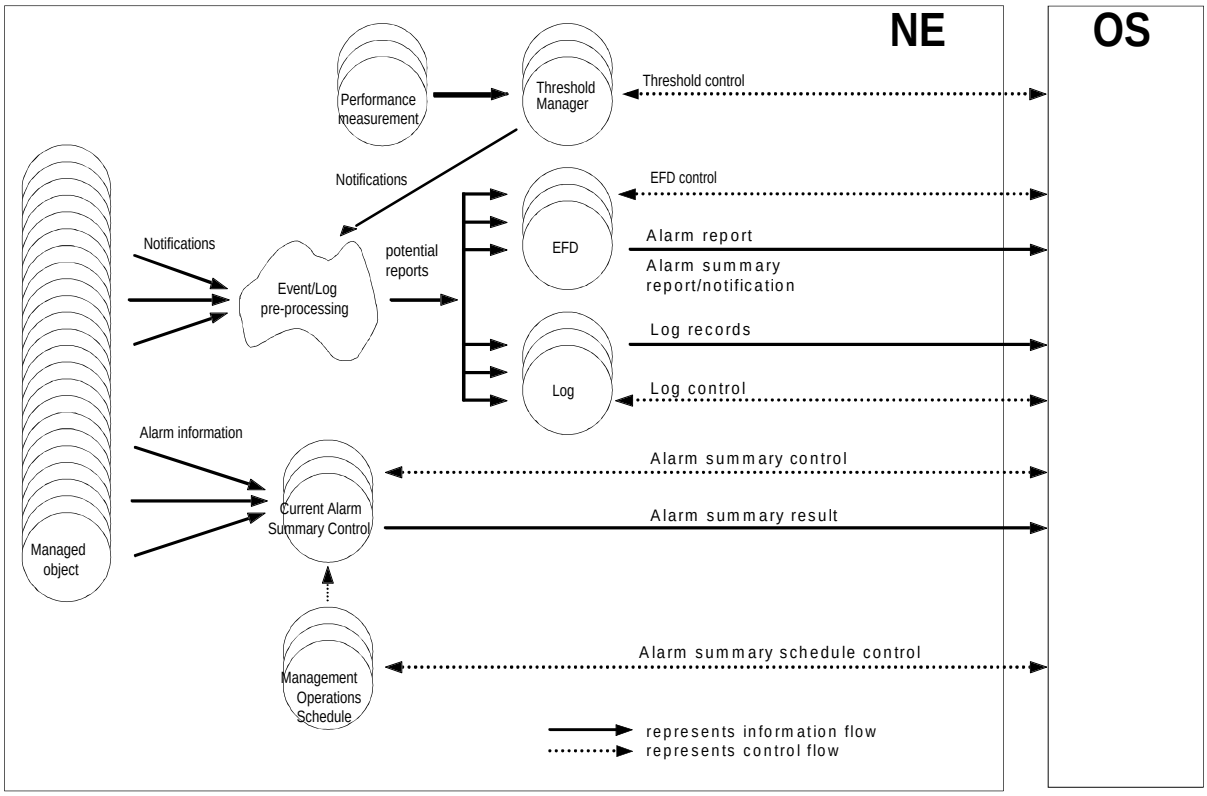

Figure 1: Alarm surveillance management model

The Alarm Surveillance model is a combination of other models (or parts of models) specified by ITU and ETSI for the various functionalities that compose this service. These consist of

- Fault detection functions which are achieved by

  - detectors that are not seen in figure 1 because, if the detectors are not manageable, they can be considered embedded in the managed object representing them. If they are manageable, they need to be represented with an instance of a specific MOC, which is outside the scope of this document;
  - the Threshold Management managed objects, which may produce notifications.

- Alarm reporting functions which are achieved by

  - the above fault detectors, including the generation of the alarm notification;
  - the Event pre-processing, which may produce a potential alarm report;
  - the EFD(s) for the final step of the reporting (discrimination, report forwarding, and EFD management). For more details, see subclause 5.1.3, ITU-T X.733 [9] and ITU-T X.734 [10].

- Log control functions which are achieved by

  - the Log pre-processing, which may produce a potential log report;
  - the Log(s) for the final step of the logging (discrimination, log record formatting and storing, log record retrieving and log management). For more details, see subclause 5.1.4, ITU-T X.733 [9], ITU-T X.735 [11] and GSM 12.00 [18].

- Alarm summary functions which are achieved by

  - the Management Operations Schedule managed objects, which are used to program the reporting time;
  - the Current Alarm Summary Control managed objects, which are used to collect the information on the outstanding alarm conditions from the managed objects, and, to produce the Potential Alarm Summary report toward the EFD or as a response to an operator request. For more details, see subclause 5.1.5 and ITU-T Q.821 [15].

5.1.2 Threshold Management

In the present document, the only form of fault detection which is standardised is the performance and equipment function monitoring by means of threshold management.

5.1.2.1 General

Thresholding may be used within the NE to generate defined alarm notifications as a result of a value change crossing the threshold level of a counter or gauge. The principles outlined are based on the recommendations of ITU-T X.721 [6].Note that both a counter and a gauge in the context of thresholding in the present document, refer to the counters and gauges as defined for Performance Measurements in GSM 12.04 [20] or elsewhere.

NOTE: The counter or gauge could also be a result of a more complex post-processing operation in a future GSM phase 2+ version of this document, i.e. an arithmetic expression (or formula) combining a number of counters or gauges. These latter type of counters or gauges could be referred to as *hybrid* counters and gauges.

The use of such threshold mechanisms allows the NE, for example, to

- detect faults, defects and anomalies (which may also be detected by other fault detection mechanisms);
- provide an early notification of a probable problem;
- provide information from the NE which for other reasons is of importance to the NE manager.

The notifications generated by thresholding shall contain information pertinent to the context in which the alarm was triggered. The information which may be contained in the notifications are a reference to the counter or gauge, threshold level, severity value, and probable cause associated with the counter or gauge.

For gauge thresholds, it shall also be possible to define whether the corresponding alarm clear notification shall be generated.

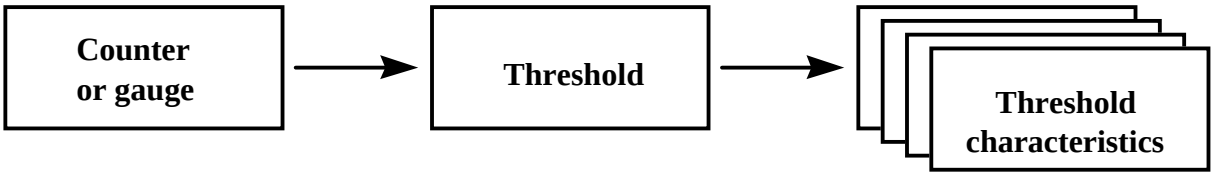

Figure 2: Inter-relationship of counters and gauges with thresholds

Figure 2 depicts the relationship between a counter or gauge and its threshold. There exists a one to one relationship between a counter/gauge and its threshold. In turn, a counter/gauge threshold consists of several characteristics which define the behaviour of the threshold and are subsequently described.

It shall be possible to have changeable threshold levels related to a counter or gauge threshold, for which an alarm notification is generated when the respective value crosses each threshold level, depending on whether notification generation has been enabled or disabled. For every gauge threshold it shall also be possible to define hysteresis levels.

It shall be possible to set and read the characteristics (e.g. perceived severity, notification generation) of counter and gauge thresholds. It shall not be possible to modify the thresholding mechanism without first deactivating thresholding for that particular thresholding mechanism.

There are two basic types of threshold mechanisms considered: counter thresholds and gauge thresholds which are subsequently described.

5.1.2.2 Counter Thresholds

A counter shall be directly related to a single counter threshold where applied. A counter threshold shall have the characteristic that a notification is triggered when the value of a counter crosses the threshold level of the counter threshold, if notifications are enabled. The threshold level is related to a defined notification.

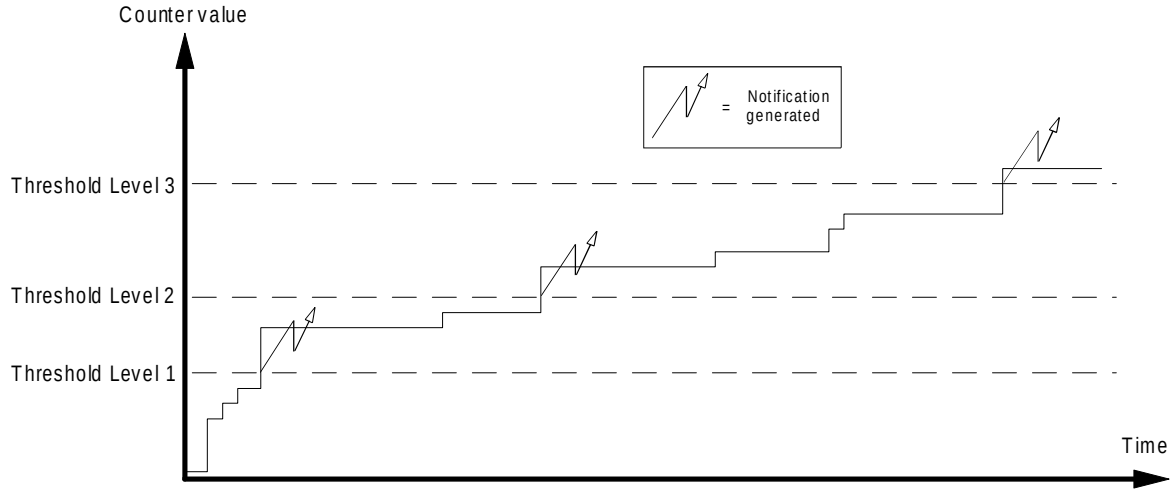

Figure 3: Example counter threshold behaviour

An offset level may also be associated and de-associated with a counter threshold level. This association may be cancelled at any time. Thus, whenever the counter threshold is triggered by a counter crossing the threshold level, the threshold level itself is incremented by the offset level. This results in the threshold triggering a further notification when the newly incremented threshold is crossed. The effect of the offset level mechanism is that notifications can be engineered to be generated at a large number of threshold levels at regular intervals, without having to directly specify each and every individual comparison level.

If it is necessary to modify a counter threshold, then it shall first be deactivated. If on reactivation of the counter threshold it is detected that a modification had been made to the counter threshold, then this will automatically result in an alarm clear notification being generated for any outstanding alarm condition for the counter threshold.

On activation or reactivation of the counter threshold, no notifications are generated for any threshold levels related to the history of the counter. Next, in the case where an offset is defined the comparison level must be recalculated such that the new comparison level is equal to the value of the initial comparison level plus a multiple of the offset level value (where the new comparison level is greater than the counter value by less than one offset level value). This calculation is algebraically defined as follows:-

$$CL_{INITIAL} + OL * (n-1) < \text{Counter Value} <= CL_{NEW} = CL_{INITIAL} + OL * n$$

where $CL_{INITIAL}$ = Initial comparison level, OL = Offset Level, and CL_{NEW} = New comparison level.

The behaviour of the threshold level is that it wraps round when it exceeds the maximum value of the counter. The result is that the value of the threshold level is less than or equal to the value of the offset value after it has been incremented by the offset level.

Further, whenever the counter to which the counter threshold is associated is reset, then the threshold level is also reset to the initial threshold value.

5.1.2.3 Gauge thresholds

A gauge is directly related to a single gauge-threshold where applied. The gauge threshold has the characteristic that a notification shall be triggered when the value of a gauge crosses the threshold value, if notifications are enabled.

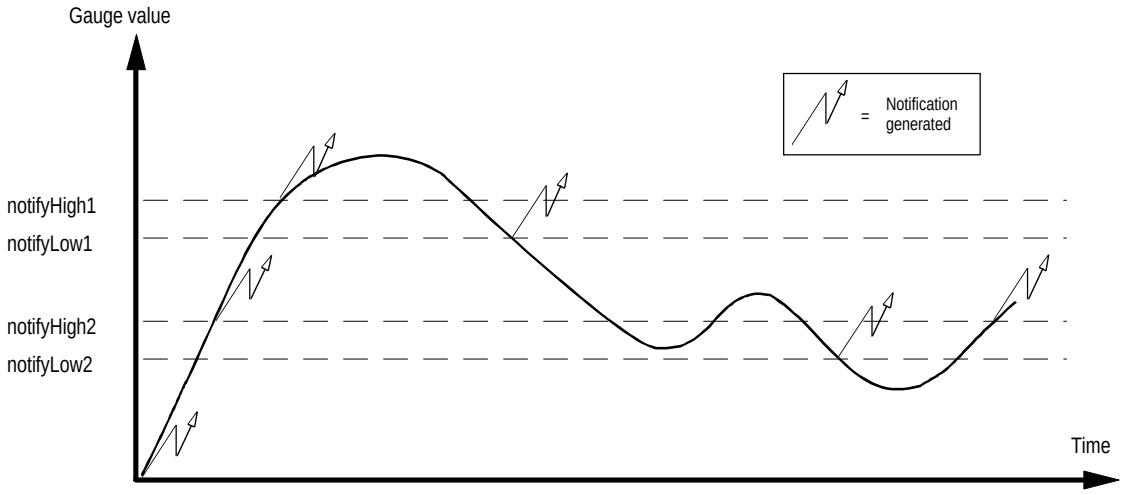

Figure 4: Example gauge threshold behaviour

If it is necessary to modify a gauge threshold, then it shall first be deactivated. Further, if on reactivation of the gauge threshold it is detected that a modification had been made to the gauge threshold, then this will automatically result in an alarm clear notification being generated for any outstanding alarm condition for the gauge threshold. On activation and reactivation, the gauge threshold will generate alarm notifications for each threshold level that is met. Therefore, an "alarm on" notification is generated for each NotifyHigh comparison level associated with an "alarm on" notification, and which is below the gauge value. Similarly, an "alarm on" notification is generated for each NotifyLow comparison level associated with an "alarm on" notification, and which is above the gauge value..

In order to avoid repeatedly triggering defined event notifications around a particular threshold level, a hysteresis mechanism may also be provided by defining threshold levels in pairs (high levels and low levels), and within the range of these two threshold levels (i.e. the hysteresis interval) no notifications are triggered.

A gauge threshold may have a set of zero or more entries defining the threshold levels associated with notifications. Each member in this set consists of two submembers (a notifyHigh value and a notifyLow value), each together with an on/off switch for the generation of the notification at that level. A notifyHigh's threshold value shall be greater than or equal to the notifyLow's threshold value.

A notification that a notifyHigh level has been reached is generated whenever the gauge value reaches or crosses above the notifyHigh level in a positive direction. A subsequent similar crossing of the notifyHigh level does not generate a further notification until after the gauge value is equal to or less than the corresponding notifyLow value.

Similarly, a notification that a notifyLow level has been reached is generated whenever the gauge value reaches or crosses below the notifyLow level in a negative direction. A subsequent similar crossing of the notifyLow level does not generate a further notification until after the gauge value is equal to or greater than the corresponding notifyHigh value.

For each pair of notifyHigh and notifyLow threshold levels, one of them shall generate an alarm notification, and the other shall generate an alarm clear notification. This means that the alarm clear notification may be generated either at the notifyHigh value or at the notifyLow value. The alarm notification shall always be generated before the alarm clear notification.

5.1.2.4 Operations

The following generic basic operations may be performed on a counter threshold or gauge threshold:

- Add Thresholding
  The thresholding mechanism and its characteristics are defined for the specified counter/gauge.

- Get Thresholding
  The characteristics of the threshold mechanism are reported for one or more counters or gauges.

- Set Thresholding

The characteristics of the specified counter or gauge threshold mechanism are modified. It is only possible to modify the threshold mechanism when thresholding has been deactivated for that particular counter threshold or gauge threshold.

- Remove Thresholding
  The thresholding mechanism is no longer defined for the specified counter/gauge (i.e. thresholding is not performed).

- Activate/deactivate Thresholding
  The thresholding mechanism for the specified counter threshold or gauge threshold is switched on or off. (Note that in the event of deactivating the notifications for a thresholding mechanism, the defined characteristics remain in place.)

5.1.3 Alarm Reporting

Event Reporting identifies the standard mechanism to be used in the NE for the generation of event notifications, pre-processing of the event notifications, discrimination of potential event reports, and the formatting and forwarding of the event reports. Within the context of the present document, event reporting is used for reporting of alarms.

In addition to the requirements identified in clauses 5 and 6, the generation of alarm notifications for the GSM NE shall be performed according to ITU-T X.733 [9], X.721 [6] and GSM 12.20 [22]. For the semantics and complete definition of the attributes and parameters, please refer to these specifications.

According to the adopted specifications, the information contained in the alarm report (apart from the standard CMIS parameters) shall be:

Probable cause	M
Specific problems	U
Perceived severity	M
Backed-up status	U
Back-up-object	C
Trend indication	U
Threshold information	C
Notification identifier	U
Correlated notifications	U
State change definition	U
Monitored attributes	U
Proposed repair actions	U
Additional text	U
Additional information	U

Where M stands for "Mandatory", U stands for "User optional" and C stands for "Conditional".

The pre-processing of the alarm notifications is not subject to standardization in the present document.

The discrimination of potential alarm reports and forwarding of alarm reports shall be performed by means of the managed object class "Event Forwarding Discriminator" (EFD) defined in ITU-T X.734 [10] and ITU-T X.721 [6]. For the semantics and complete definition of the attributes and parameters, please refer to these specifications.

According to ITU-T X.734 [10] the functionalities of the event reporting management (which here are used for for alarm reporting) are:

- Initiation of event forwarding
- Termination of event forwarding
- Suspension of event forwarding
- Resumption of event forwarding
- Modification of event forwarding conditions
- Retrieval of event forwarding conditions.

Alarm notifications are extremely important for the management of the NE. If notifications are lost or delayed it may affect the operator's ability to effectively manage the system. Breakdowns in communication between the NE and a

remote OS shall be accounted for. Therefore the NE shall provide a storage capability locally, where notifications can be accessed and retrieved by the managing system.

The output queue from the NE to the OS is currently being studied by ITU-T SG4, and it is expected that the analysis of the output queue will be addressed in a future GSM Phase 2+ version of this document.

5.1.4 Log control

Logging identifies the standard mechanism used in the NE for the log pre-processing of event notifications, the discrimination of potential log records, the formatting and the storing of the log records in one or more logs and the retrieving of the log records by the OS from the NE. Within the context of the present document, this functionality is used for logging alarm notifications.

The logging of alarm notifications in the NE shall be performed by means of the two managed object classes "Log" and "Alarm Record" which are defined in ITU-T X.735 [11], ITU-T X.733 [9] and ITU-T X.721 [6]. For the semantics and complete definition of the attributes and parameters, please refer to these specifications.

According to the adopted specifications the functionalities for the log control are:

- Retrieval of alarm records from the Log;
- Deletion of alarm records in the Log;
- Initiation of alarm logging;
- Termination of alarm logging;
- Suspension of alarm logging;
- Resumption of alarm logging;
- Scheduling of alarm logging;
- Modification of logging conditions;
- Retrieval of logging conditions.

In addition to the alarm record retrieval mechanism mentioned above, GSM 12.00 [18] annex B provides a definition for transfer of selected log records from a log instance to the OS as a file. This functionality shall be controlled through the managed object class "simpleFileTransferControl". The functionality required is:

- Bulk transfer of alarm records from the Log (transfer of data from the NE to OS requested by OS).

5.1.5 Alarm summary

The alarm summary functions allow an NE to report a summary of the outstanding alarm conditions of all or selected managed objects to the OS. They provide the following facilities:

- Definition of the alarm summary criteria;
- Requesting of the alarm summary criteria;
- Reporting of the alarm summary (on demand or scheduled);
- Scheduling of a current alarm summary (optional).

By setting the criteria for the generation of current alarm summary reports, the operator can determine whether outstanding alarm condition information on objects is included in a current alarm summary report. The following criteria shall be supported:

- managed objects with outstanding alarm conditions;
- perceived severity;
- alarm status;
- probable cause (optional).

The alarm summary report is sent from the NE to the OS and contains information about those outstanding alarm conditions that match the alarm summary criteria. The information for scheduled reports shall include:

- identification of the object instances with outstanding alarm conditions;
- alarm status;
- perceived severity;
- probable cause.

The presence of alarm status, perceived severity and probable cause in unscheduled reports, is dependent on whether they have been requested.

The model adopted for the alarm summary is based on the one defined in ITU-T Q.821 [15]. To provide the alarm summary functions, the OS and the NE shall be able to manage the current alarm summary control object class and, optionally, the management operations schedule object class. The first one provides the criteria for the generation of current alarm summary reports while the latter allows the scheduling of an alarm summary to occur at a specified time or periodically.

Information regarding a managed object shall be included in a current alarm summary report if :

- The managed object is included in the object list (which describes a list of object instances).
- The managed object has an alarm status that is present in the alarm status list (describes criteria for inclusion in the current alarm summary report and consists of a set of possible alarm status values (see ITU-T M.3100 [2]).
- The managed object has an alarm with a perceived severity and probable cause matching members of the perceived severity list and probable cause list.

If the object list is empty, then the criteria in the current alarm summary control shall be applied to all the objects of the NE. If any of the other criteria are empty then they are not used in selecting objects that will appear in the current alarm summary report.

The alarm summary provides information about all the objects selected and matching the selection criteria.

5.2 Fault localisation

This service component requires that the NE shall have the capability to provide all the necessary information to the OS in order to localise the faults that may occur in the NE itself.

In the process of localising the faults, the first piece of information is provided by the alarm surveillance service component, since after the fault detection an alarm notification is generated and, if the corresponding potential report is not discriminated, an alarm report which should contain sufficient information to localise the fault is forwarded to the OS.

In order to support localization wherever possible (as mentioned in 5.1 item 1), the NE should generate a single notification for a single fault.

In case of ambiguity in the localisation, the operator can request from the NE (in case this optional feature is supported by the NE):

- the execution of diagnostic tests
- retrieval of some log records to have a clear view of the events that occurred before the failure
- other information like the current configuration, the value of some measurements, the value of some attributes, etc.

Testing may also be needed to verify the fault if the localisation process is initiated due to, for example, customer complaints instead of an alarm report.

The detailed fault localisation process is closely related to the NE's internal architecture as well as the operator's maintenance and operating procedures and thus not subject to standardisation in the present document.

The resolution of the localisation should be down to one LRU (Least Replaceable Unit) for the majority of the faults. When the NE cannot localise the fault to one LRU, it should, as far as possible, indicate a restricted number of LRUs, ordered according to the probability of being faulty.

5.3 Fault correction

This service component requires capabilities that are used in various phases of the fault management:

1) After a fault detection, the NE shall be able to evaluate the effect of the fault on the telecommunication services and autonomously take recovery actions in order to minimise service degradation.

2) Once the faulty unit(s) has been replaced or repaired, it shall be possible from the OS to put the previously faulty unit(s) back in to service so that it is restored to its normal operation. This transition should be done in such a way that the currently provided telecommunication services are not, or minimally, disturbed.

3) At any time the NE shall be able to perform recovery actions if requested by the operator. The operator may have several reasons to require such actions; e.g. he has deduced a faulty condition by analysing and correlating alarm reports, or he wants to verify that the NE is capable of performing the recovery actions (proactive maintenance).

The recovery actions that the NE performs (autonomously or on demand) in case of fault depend on the nature and severity of the faults, on the hardware and software capabilities of the NE and on the current configuration of the NE.

The faults are distinguished in two categories: software faults and hardware faults.

In the case of software faults, depending on the severity of the fault, the recovery actions may be: System initialisations (at different levels), activation of a backup software load, activation of a fallback software load, download of a software unit etc.

In the case of hardware faults, the recovery actions depend on the existence and type of redundant (back-up) resources.

If the faulty resource has no redundancy, the recovery actions shall be:

a) Isolate and remove from service the faulty resource so that it cannot disturb other working resources.

b) Remove from service the physical and functional resources (if any) which are dependent on the faulty one. This prevents the propagation of the fault effects to other fault-free resources.

c) Adjust the Operational State and Status attributes of the faulty managed object and the affected managed objects, in a consistent way, reflecting the new situation.

d) Generate and forward (if possible) the reports to inform the OS about all the changes performed.

If the faulty resource has redundancy, the NE shall perform actions a), c) and d) above and, in addition, the recovery sequence which is specific to that type of redundancy.

In the NE, the redundancy of some resources may be provided to achieve fault tolerance and to improve the system availability. There exist several types of redundancy (e.g. hot standby, cold standby, duplex, symmetric/asymmetric, N plus one or N plus K redundancy, etc.) and for each one, in case of failure, there is a specific sequence of actions to be performed. The present document specifies the management (the monitoring and control) of the redundancies, but does not define the specific recovery sequences of the redundancy types.

The NE shall provide the OS with the capability to monitor and control any redundancy of the NE. The control of a redundancy (which means the capability to trigger a change-over or a change-back) from the OS can be performed by means of the state management services or by means of specific actions.

When the state management services are used, the transitions are triggered by locking/unlocking one of the objects participating in the redundancy.

In the case of a failure of a resource represented by a managed object providing service, the recovery sequence shall start immediately. Before or during the change-over, a temporary and limited loss of service shall be acceptable. In the case of a management command, the NE should perform the change-over without degradation of the telecommunication services.

5.3.1 The model

The model adopted supports the redundancy management part of the fault correction service component. The model is imported as a subset of the ITU-T G.774.03 recommendation [13] - SDH Management of Multiplex-Section Protection

for the Network Element View. The parts relevant for redundancy management in GSM 12.11 are introduced in subclause 5.2.1 of G.774.03, and the necessary managed object classes, protectionGroup and protectionUnit, are subsequently defined in subclause 6.3, 6.4 and related definitions. They are also depicted and exemplified in the annexes A and B of G.774.03.

5.4 Testing

This service component provides capabilities that can be used in different phases of the fault management; therefore it can also be seen as a 'support' to other service components. For example:

- when a fault has been detected and if the information provided through the alarm report is not sufficient to localise the faulty resource, tests can be executed to localise the fault (Fault Localisation service component);
- during normal operation of the NE, tests can be executed for the purpose of detecting faults (Alarm Surveillance service component);
- once a faulty resource has been repaired or replaced, before it is restored to service, tests can be executed on that resource to be sure that it is fault free (Fault correction service component).

However, regardless of the context where the testing is used, its target is always the same: verify if a system's physical or functional resource performs properly and, in case it happens to be faulty, provide all the information to help the operator to localise and correct the faults.

Testing is an activity that involves the operator, the managing system (the OS) and the managed system (the NE). Generally the operator requests the execution of tests from the OS and the managed NE autonomously executes the tests without any further support from the operator.

In some cases, the operator may request that only a test bed is set up (e.g. establish special internal connections, provide access test points, etc.). The operator can then perform the real tests which may require some manual support to handle external test equipment. Since the "local maintenance" and the "inter NE testing" are out of the scope of the present document, this aspect of the testing will not be treated any further.

The requirements for the testing service component are based on ITU-T X.745, where the testing description and definitions are specified.

5.4.1 The model

The model adopted for the test management is specified in ITU-T X.745 [13] and depicted in figure 5. According to this model, the execution of a test involves two entities: a manager that initiates the test (test conductor), and an agent that executes the test (test performer). In this case, the test conductor resides on the OS while the test performer resides on the managed NE. They communicate with each other via the Q3 interface.

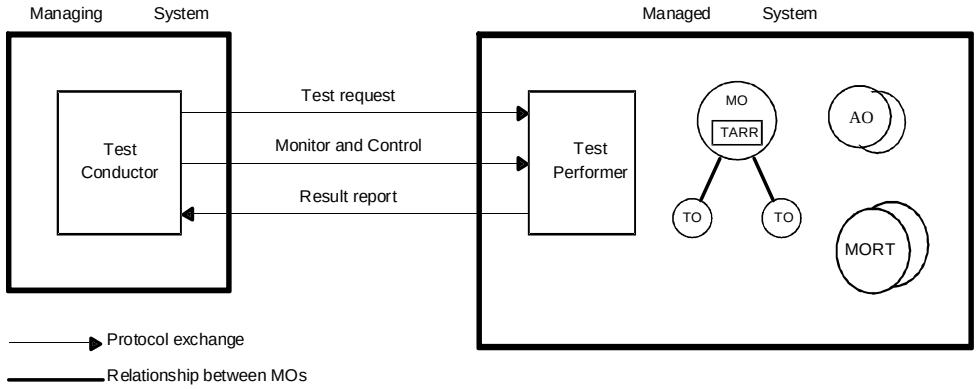

Figure 5: Testing Management Model

The model requires that, in the managed NE, there is at least one object which has the functionality to receive and to respond to the test requests coming from the manager; this functionality is named TARR functionality (where TARR stands for Test Action Request Receiver).

The model also requires that, in the managed NE, for each test execution there is at least one MORT, which is the managed object whose functionality is requested to be tested; and in some cases, there could be one or more associated objects (AO) which participate in the test execution although they are not the first target of the test. These two conditions imply that the resources which are subject to be individually tested should each be represented by a managed object.

The model distinguishes between two types of tests: uncontrolled and controlled tests. It is recommended to use the uncontrolled tests (that cannot be monitored or controlled) to model those tests that run very fast and can provide the test results very quickly. For the tests that may take a long time (minutes), it is preferable to use the controlled tests so that, during the execution time, the operator can perform some management activity on the tests, like the monitoring of the test state, test suspend, test resume, etc.

For controlled tests, Test Objects (TOs) are created in the NE as a result of the test request.

5.4.2 Testing requirements

In order to have a flexible, efficient and powerful testing service, the managed NE shall provide the following capabilities to the managing system.

The NE should provide a set of tests which homogeneously cover all the physical and functional parts of the system. Every possible fault occurring on every part of the NE should be covered by at least one test.

The tests should localise the faults as precisely as possible. For the majority of the faults, the localisation should be down to one LRU.

The NE shall have the ability to provide, to the OS, the list of the supported tests (both controlled and uncontrolled tests) and all the related information.

In the NE, the received test requests shall be checked to ensure that the test execution does not produce any uncontrolled and undesired effect on the telecommunication services currently provided by the NE itself. These acceptance checks depend on the type of test (intrusive or non-intrusive), on the current state of the resources to be tested (MORTs), on the current state of other involved resources necessary to set up the test environment (AO), on the current state of the test objects (TO), on the availability of the test infrastructures etc.

The non-intrusive tests can be run independently from the state of the MORTs and therefore, they do not require any preceding state change of the MORTs.

The intrusive tests can be run only if the MORTs are in the administrative state "locked" and/or in the operational state "disabled". The operator may use the state management services to change the administrative state of the MORTs; the change from unlocked to locked can be graceful, using the transient "shutting down" state.

Depending on the result of the tests, the operational state of the MORTs can change from enabled to disabled if some tests do not pass, or from disabled to enabled if all the tests pass. In the first case, when the tests detect a fault, they shall generate an alarm notification using the alarm surveillance services. In the second case, when the MORTs are returned enabled they shall generate an alarm notification with severity "cleared" and forward it to the EFD.

It is also possible that some tests fail because of minor faults, so it may be convenient to leave the MORT in service (enabled) instead of removing it from service. In this case, the availability status "degraded" shall be used to remind the operator that the MORT, although enabled, is not in perfect condition and has some minor trouble that requires correction.

When a MORT changes its operational state as a consequence of a test execution, the NE shall automatically and consistently change the state of all the other related managed objects, whose operational state is dependent on the MORT's operational state.

If the NE provides the capability to execute controlled tests, then it shall be possible to:

- suspend and resume the tests
- monitor the processing of the tests through the state attribute of the TOs
- terminate the tests
- get the test results from the TO when they are provided as attributes.

When a controlled test is suspended, the TO is put in "suspended" state while the involved resources (the AOs) may be released, depending on the specific characteristics of the TO. When the test is resumed, the TO itself determines at what point in the test life-cycle the test will be resumed.

If the NE provides the capability to schedule the test execution within a time window, then it shall be possible to set up the boundaries of the time window with a start time and a stop time. The start time is the earliest time at which the test performer can start the test execution (the actual starting time depends on the current conditions of the NE during the time window). The stop time is the latest time at which the test execution should be completed. The actual stop time depends on when the test was actually started and usually it should be reached before the stop time. The NE may also provide the capability to schedule the time to perform the initialisation of the tests and the time window within which to perform the real test execution. The test performer shall provide complete information to the OS about the actual initialisation time and execution time, together with the test results. If the NE cannot perform the test within the time window, the OS shall be informed.

The results of the tests are made available by the NE as attribute values of the TO(s) and/or returned via notifications issued by the TO(s).. The latter method is possible only for controlled tests. The test results shall contain all the information necessary to localise the faults, and may propose repair actions for the faults, if any.. If for any reason a test is prematurely terminated, the partial results collected so far may be reported to the OS.

In the case of uncontrolled tests, the results are reported in the reply(ies) to the test request.

It is expected that the analysis of "cyclic tests" will be addressed in a future GSM Phase 2+ version of this document.

5.4.3 Test Categories

The test categories are defined in ITU-T X.737 [12] to group and to classify all the tests commonly applied to telecommunication systems, the tests of each group having the same characteristics from a functional and management point of view.

Some of the aspects of the testing that are common to all the test categories are specified by ITU-T X.745 [13]. In ITU-T X.737 [12] specific managed object classes (one for each test category) are derived from the TO MOC and characterised with additional packages, to model the characteristics specific to each test category. The test categories adopted for this specification are shortly described below.

5.4.3.1 Resource Self Test

This test category is used to characterise those tests that verify the ability of single resources (physical or functional) to correctly perform their allotted functions. For this test category there is only one MORT which represents the resource to be tested. This test category shall not have any AO.

5.4.3.2 Resource Boundary Test

This test category is used to characterise those tests that check the physical resources by observing them from their boundary. To test these physical resources there are one or more Points of Control and Observation (PCOs) from which they are stimulated by means of electrical signals, and observed in their reactions, measuring other electrical signals.

For this test category, there shall be one MORT that represents the physical resource under test, and one or more AOs which represent the PCOs. In this case, the AOs are usually objects specifically designed for testing purposes.

5.4.3.3 Connection Test

This test category is used to characterise those tests that verify the capability of a communication path (real or virtual) to support the desired service or level of functionality. In case it is possible to establish different types of connections on the same communication path, the test can be arranged as a sequence of different 'exercises', each one dedicated to a type of connection.

For this test category, there is one MORT that represents the communication path under test, and two AOs (or more in case of multitype connection) which represent the resources used to verify the connection. According to the scope restrictions of this specification, both the MORT and the AOs always belong to the same NE, therefore there is no special requirement for the AOs to agree on the details of the exercises to be performed.

The tests of this category shall be organised taking into account that the MORT is the communication path. Therefore when an exercise fails and the fault cannot be localised to the MORT (because the fault could also be on one of the AOs), then the exercise should be repeated with different AOs.

5.4.3.4 Data Integrity Test

The purpose of this test category is to verify the capability of a resource to correctly exchange data with other resources.

For this test category there is one MORT which represents the resource under test, and one or more AOs which co-operate to carry out the test.

Usually, during the test execution, the MORT transmits data to an AO which, upon reception, reflects the data to the MORT which will verify that the data is correctly received. In case of one way transmission, the receiving AO does not reflect the data back to the MORT but it has to verify the correctness of the received data.

Also, in this case when a single exercise cannot localise the fault to the MORT, the exercise should be repeated with another AO.

5.4.3.5 Loopback Test

The purpose of this test category is to verify that data can be sent and received through a communication path which is composed of one or more resources, each one represented by a managed object. A loopback may be implemented in a variety of ways, for example by physical loop connection or by echoing data received.

For this test category the MORTs are the MOs of the communication path which is under test. Not all the MOs of the communication path are necessarily under test; some of them could be involved as AOs, especially for the objects that represent the loopback points.

In some cases it is desirable to be able to specify that a loopback be set somewhere within a MORT. This may occur when a resource is modelled as a single MO but is actually complex enough to allow loopbacks to occur in several places within the MORT. For these cases the AOs may be present at locations that will test only a part of the MORT.

To allow the invocation of loopback tests, the configuration in which the MORTs and the AOs are to be placed, needs to be defined.

The manager may request a single loopback test by involving only one AO to operate the loopback point, or a multiple loopback test, involving a set of AOs. In this latter case, the order in which the AOs are operated is not defined.

5.4.3.6 Protocol Integrity Test

The purpose of this test category is to verify if the MORT can conduct proper protocol interactions with a specified AO.

It is planned that the analysis of the protocol integrity test will be addressed in a future GSM Phase 2+ version of this document.

5.4.3.7 Test-Infrastructure Test

The purpose of this test category is to verify the ability of the NE to initiate tests, return result reports and, (for the controlled tests) respond to monitoring and control actions.

For this test category the MORT is the MO which has the TARR functionality while the TO is a "null test" whose purpose is solely to verify the correct behaviour of the test performer.

6 Fault management functions

According to TMN principles, (see M.3200 [3] and M.3400 [4]) the TMN management service components described in the previous clauses are achieved by means of the TMN management functions described in this clause.

Since the PLMN NEs shall include the "Open Systems Interconnection" capabilities, in general, the telecommunications management functions are based on the "OSI System Management Functions" which are specified by ITU-T with the X.7xx series of recommendations. Each of these recommendations specifies a set of management functions for a specific aspect of the system management. The OSI system management functions are based on the Q.3 protocol services CMISE, ACSE and ROSE, which are specified in general by ITU-T, and in particular for the GSM System by the GSM 12.01 [19] specification.

6.1 Alarm surveillance functions

6.1.1 Threshold Management functions

The Threshold Management Functions identify the management and generation of defined alarm notifications from counter thresholds and gauge thresholds.

The management is covered by the threshold management functions subsequently identified, and based on the recommendations in ITU-T X.721 [6]. The permitted functions are Add Thresholding, Get Thresholding, Set Thresholding, Remove Thresholding and Activate/Deactivate Thresholding.

Add Thresholding: The OS requests the NE to add (i.e. define) the thresholding mechanism for the identified counter or gauge. The characteristic which is defined for the counter threshold or gauge threshold is:

- Thresholding Status — Can be set to True (thresholding mechanism added and in the activated state) or False (thresholding mechanism added and in the deactivated state).

The characteristics which are defined for each member of a counter threshold are:

- Comparison Level — The counter level at which the defined notification may be generated if thresholding is activated (see Notify On or Off below).
- Initial Comparison Level — The initial value of the Comparison Level at which the defined notification may be generated. The Comparison Level is set to this level when adding thresholding, and when the counter associated with the counter threshold is reset.
- Offset Level — Whenever the counter threshold is triggered by a counter crossing the Comparison Level, the Comparison Level itself is incremented by the offset level, if the offset level is not zero.
- Severity — Severity level to be included in the defined notification.
- Notify On or Off — The thresholding mechanism and the generation of defined notifications for the counter threshold or gauge threshold when crossing the Comparison Level may be switched on or off.

The characteristics which are defined for each member of a gauge threshold are Notify High and Notify Low, which both in turn consist of:

- Comparison Level — The gauge level at which the defined notification may be generated if thresholding is activated (see Notify On or Off below).
- Severity — Severity level to be included in the defined notification.
- Notify On or Off — The thresholding mechanism and the generation of defined notifications for the gauge threshold when crossing the Comparison level may be switched on or off.

Note: This function adds the thresholding mechanism for the counter or gauge, but the thresholding is only performed if it is activated (see Thresholding Status below).

Get Thresholding: The OS requests the NE to report the thresholding mechanism characteristics for the identified counter or gauge.

The characteristic which is reported for the counter threshold or gauge threshold is:

- Thresholding Status — Set to True if the thresholding mechanism is activated and to False if the thresholding mechanism is deactivated.

The characteristics which are reported for each member of a counter threshold are:

- Comparison Level — See above for a description of Comparison Level
- Initial Comparison Level — See above for a description of Initial Comparison Level
- Offset Level — See above for a description of Offset Level
- Severity — See above for a description of Severity
- Notify On or Off — See above for a description of Notify On or Off

The characteristics which are reported for a gauge threshold are Notify High and Notify Low, which both in turn consist of:-

- Comparison Level — See above for a description of Comparison Level
- Severity — See above for a description of Severity
- Notify On or Off — See above for a description of Notify On or Off

Set Thresholding: The OS requests the NE to modify the thresholding mechanism characteristics for the identified counter or gauge.

The characteristics which may be modified for each member of a counter threshold are:

- Comparison Level — See above for a description of Comparison Level
- Initial Comparison Level — See above for a description of Initial Comparison Level
- Offset Level — See above for a description of Offset Level
- Severity — See above for a description of Severity
- Notify On or Off — See above for a description of Notify On or Off

The characteristics which may be modified for a gauge threshold are Notify High and Notify Low, which both in turn consist of:

- Comparison Level — See above for a description of Comparison Level
- Severity — See above for a description of Severity
- Notify On or Off — See above for a description of Notify On or Off

It is only possible to modify the threshold characteristics when thresholding has been deactivated for the counter threshold or gauge threshold (see Thresholding Status below).

Remove Thresholding: The OS requests the NE to delete (i.e. no longer define) the thresholding mechanism for the identified counter or gauge.

The characteristics which are deleted for each member of a counter threshold are:

- Comparison Level — See above for a description of Comparison Level
- Initial Comparison Level — See above for a description of Initial Comparison Level
- Offset Level — See above for a description of Offset Level
- Severity — See above for a description of Severity
- Notify On or Off — See above for a description of Notify On or Off

The characteristics which are deleted for a gauge threshold are Notify High and Notify Low, which both in turn consist of:

- Comparison Level — See above for a description of Comparison Level
- Severity — See above for a description of Severity
- Notify On or Off — See above for a description of Notify On or Off

Activate/Deactivate Thresholding: The OS requests the NE to either activate or deactivate the thresholding mechanism for the identified counter or gauge.

The characteristic which is modified for the counter threshold or gauge threshold is:-

- Thresholding Status — Set to True on activation of the thresholding mechanism and to False on deactivation of the thresholding mechanism

6.1.2 Alarm Reporting functions

The alarm reporting functions identify standard mechanisms for the generation of alarm notifications. The functions of alarm reporting fall into four general areas:

- The generation of an alarm notification by a managed object,
- The forwarding of that alarm notification to the OS,
- The storage of the alarm record in the log (optional), and
- The retrieval/deletion of the alarm record from the log (optional).

Report alarm: The NE notifies OS of alarm information upon the occurrence of an event. This is defined in M.3400 [4] and Q.821[15].

Route alarm report: OS specifies to the NE the destination address(es) for a specified set of alarm reports. This is defined in M.3400 [4] and Q.821[15].

Request alarm report route: OS requests the NE to send the current assignment of the destination address(es) for a specified set of alarm reports; NE responds with the current assignment of destination address(es). This is defined in M.3400 [4] and Q.821[15].

Condition alarm reporting: OS instructs the NE to assign Event Forwarding Discriminator attributes as specified by the OS or the OS instructs the NE to create/delete instances of Event Forwarding Discriminators. This is based on M.3400 [4] and Q.821[15].

Request alarm report control condition: OS requests the NE to send the current assignment of specified Event Forwarding Discriminator attributes; NE responds with the current assignment of specified attributes. This is defined in M.3400 [4] and Q.821[15].

Allow/Inhibit alarm reporting: OS instructs the NE to allow/inhibit alarm reports to the OS. This is defined in M.3400 [4] and Q.821[15].

Request alarm report history: OS requests the NE to send specified historical alarm information from a log; NE responds with the specified information. Alternatively using the GSM 12.00 annex B based on the managed object class simpleFileTransferControl, the OS may request the NE to generate a datafile from the specified historical alarm information which is subsequently transferred to the OS. This is based on definitions in Q.821 [15] and GSM 12.00 [18].

This method is referred to as the "Bulk Transfer of Alarm records" procedure and contains the following steps (for a detailed description, see GSM 12.00 [18] annex B):

- Start file creation basic service: The OS initiates the data file creation by issuing the requestTransferUp action. The resultType field in this action may contain either of the options defined in GSM 12.00 [18] annex B:

    - objectSelection: in this case, this parameter identifies a log (or a superior object) to be used as the base object for a scoped and filtered retrieval of alarm records (in the same way as they would be selected with CMISE primitives);

    - typeOfFile: the fileType parameter shall be set to the alarmRecords value. Additionally, the optional parameter FileSubType may be used to select alarm records according to predefined criteria such as the severity or the time of occurrence of the alarm (the coding of FileSubType parameter is described in Annex B of this specification).

- The NE creates one or more datafiles containing the requested Alarm Record data. The format of the transferred data shall be the file type "ObjectDataFile";

- Notify file creation basic service: The NE notifies the OS about the datafile(s) with the transferUpReady notification. If typeOfFile option was used, then fileType field (and optionally fileSubType) shall be set to the same value(s) in this notification as in requestTransferUp action. If the objectSelection option was used, the alarmRecords value (13) may be used for the fileType field;

- The OS initiates the transfer of the datafile(s) using FTAM services (see GSM 12.01 [19]);

- Complete file transfer basic service: The OS informs the NE about the completion of the file transfer by issuing the transferUpReceived action.

Delete alarm report history (added in Q.821 [15] and GSM 12.11 compared to M.3400 [4]): OS requests the NE to delete specified historical alarm information from the Log. This is defined in Q.821 [15].

6.1.3 Log control functions

The log control functions are used by the OS to control the operation of the alarm log(s) in the NE.

The retrieval and deletion of information stored in the log(s) is covered by the functions "Request alarm report history" and "Delete alarm report history" (part of the Alarm Reporting functions).

As mentioned in subclause 5.1, the logging of alarm notifications in the NE shall be performed by means of two managed object classes: "Log" and "Alarm Record", defined in ITU-T X.735 [11], X.733 [9] and X.721 [6]. The functionalities of the Log which belong to the Log control functions are:

- Initiation of logging.
- Termination of logging.
- Suspension of logging.
- Resumption of logging.
- Scheduling of logging.
- Modification of logging conditions.
- Retrieval of logging conditions.

The management functions defined for these functionalities are: (M.3400 [4], Q.821 [15]):

Allow/Inhibit logging: OS instructs the NE to allow/inhibit logging of Alarm records.

Condition logging: OS instructs the NE to assign Log attributes as specified by the OS, or to create/delete instances of the Log object class.

Request log condition: OS requests the NE to send the current assignment of specified Log attributes; NE responds with the current assignment of specified attributes.

6.1.4 Alarm summary functions

The alarm summary functions are based on the ones defined in Q.821 [15].

Request current alarm summary: The OS requests the NE to send a current alarm summary; the NE responds with the summary.

Condition current alarm summary: The OS instructs the NE to assign Current Alarm Summary Control attributes as specified by the OS, or to create/delete instances of the Current Alarm Summary Control object class. Note that this function is not present in Q.821 (1993) [15].

Request current alarm summary condition: The OS requests the NE to send the current assignment of specified Current Alarm Summary Control attributes; NE responds with the current assignment of specified attributes. Note that this function is not present in Q.821 (1993) [15].

Report current alarm summary: The NE provides the OS (based on a pre-defined schedule) with a current alarm summary report. This function is optional.

Route current alarm summary: The OS specifies to the NE the destination address(es) for a specified set of scheduled alarm summary reports. This function is optional.

Request current alarm summary route: The OS requests the NE to send the current assignment of the destination address(es) for a specified set of scheduled alarm summaries; the NE responds with the current assignment of destination address(es). This function is optional.

Schedule current alarm summary: The OS specifies a schedule for the NE to establish for the reporting of current alarm summaries. The schedule information specifies what should be reported as well as when it should be reported. This function is optional.

Request current alarm summary schedule: The OS requests the NE to send the current schedule information for current alarm summary reporting. The NE responds with the schedule information. This function is optional.

Allow/inhibit current alarm summary: The OS instructs the NE to allow/inhibit the reporting of the scheduled current alarm summaries. This function is optional.

6.1.5 Alarm surveillance related basic services

This subclause describes the mapping between the management functions, managed objects and the management services (called 'basic services' in the present document, in contrast to Q.821 [15]) needed to support the management functions defined in subclauses 6.1.2 - 6.1.4. The mapping of the basic services to the supporting CMIS services is also presented.

The mapping of the following basic services to the confirmed or non-confirmed mode of the supporting CMIS services, except where specified, is a local implementation issue and is not specified in the present document.

The term 'basic service' is used in the present document because, in contrast to Q.821 [15], this service component is based on both the management services as specified in Q.821 [15], as well as the management services for SFTC defined in the present document (in subclause 6.1.2) and in GSM 12.00 [18].

For the service definitions (except for the SFTC basic services, Condition Current Alarm Summary and Request Current Alarm Summary Condition), see Q.821[15].

Table 1: Alarm surveillance functions, managed objects and basic services

Function(s)	Managed Object(s)	Basic Service(s)	CMIS service(s)
Report alarm	EFD [1]	Alarm reporting (X.733)	M-EVENT-REPORT
Route alarm report	EFD	Set EFD (Q.821)	M-SET
Request alarm report route	EFD	Get EFD (Q.821)	M-GET
Condition alarm reporting	EFD	Set EFD (Q.821) Initiate alarm reporting (Q.821) Terminate alarm reporting (Q.821)	M-SET M-CREATE M-DELETE
Request alarm report control condition	EFD	Get EFD (Q.821)	M-GET
Allow/inhibit alarm reporting	EFD	Resume/suspend alarm reporting (Q.821)	M-SET
Request alarm report history	(Log [2]), Alarm Record SFTC, (Log, Alarm Record) [3]	Alarm report retrieving (Q.821) Start file creation (GSM 12.00, 12.11) Notify file creation (GSM 12.00, 12.11) Complete file transfer (GSM 12.00,12.11)	M-GET M-ACTION M-EVENT-REPORT M-ACTION
Delete alarm report history	(Log [2]), Alarm Record	Alarm report deleting (Q.821)	M-DELETE
Allow/inhibit logging	Log	Resume/suspend logging (Q.821)	M-SET
Condition logging	Log	Set Log (Q.821) Initiate log (Q.821) Terminate log (Q.821)	M-SET M-CREATE M-DELETE
Request log condition	Log	Get Log (Q.821)	M-GET
Request Current Alarm Summary	CASC	Retrieve current alarm summary (Q.821)	M-ACTION
Condition Current Alarm Summary [4]	CASC	Set current alarm summary control (Q.821) Initiate current alarm summary control (Q.821) Terminate current alarm summary control (Q.821)	M-SET M-CREATE M-DELETE
Request Current Alarm Summary Condition [4]	CASC	Get current alarm summary control (Q.821)	M-GET
Report Current Alarm Summary	CASC, (MOS) [5]	Current alarm summary reporting (Q.821)	M-EVENT-REPORT
Route Current Alarm Summary	MOS[6]	Terminate MOS (Q.821) Initiate MOS (Q.821)	M-DELETE M-CREATE
Request Current Alarm Summary Route	MOS [6]	Get MOS (Q.821)	M-GET
Schedule Current Alarm Summary	MOS	Set MOS (Q.821) Initiate MOS (Q.821) Terminate MOS (Q.821)	M-SET M-CREATE M-DELETE
Request Current Alarm Summary Schedule	MOS	Get MOS (Q.821)	M-GET
Allow/inhibit Current Alarm Summary	MOS	Resume/suspend MOS (Q.821)	M-SET
Add Thresholding	thresholdManager	Create thresholdManager (GSM 12.11)	M-CREATE
Get Thresholding	thresholdManager	Get thresholdManager (GSM 12.11)	M-GET
Set Thresholding	thresholdManager	Set thresholdManager (GSM 12.11)	M-SET
Remove Thresholding	thresholdManager	Delete thresholdManager (GSM 12.11)	M-DELETE
Activate/Deactivate Thresholding	thresholdManager	Activate thresholdManager (GSM 12.11) Deactivate thresholdManager (GSM 12.11)	M-SET M-SET

NOTE 1: Here EFD is the MIS-User of the Alarm Reporting basic service (M-EVENT-REPORT service), not the target of the operation as for the other services in this table.
NOTE 2: Log may be used as the base object class and instance for scoped and filtered operations. The scope of these operations shall not include the base object.
NOTE 3: Log and Alarm record objects are addressed indirectly through the SFTC when objectSelection option is used.
NOTE 4: This function is added compared to Q.821 (1993). However, ITU-T SG4 has decided to introduce this function in the next version of Q.821.
NOTE 5: CASC is here the source of the emitted report. MOS is indirectly related to reporting by triggering the report generation according to the schedule.
NOTE 6: According to the Q.821 model the destination of the scheduled Current Alarm Summary reports is specified by the MOS read-only attribute destinationAddress. Thus, these reports are not routed via the EFD, and in order to change the destination address, an instance of MOS has to be re-created.

6.2 Fault localisation functions

Fault localisation is accomplished through the analysis of the information contained in alarm notifications and/or test results. Therefore functions which are part of alarm surveillance and test service components may be used for fault localisation.

6.2.1 Alarm report function

This function supports the requirements for the agent to report and for the manager to receive information that may be used to localise a fault to a LRU. See subclause 6.1.2 for complete information.

For alarm notifications, the first piece of localisation information is provided by the identification of the object instance reporting the alarm. Following this, the "Probable Cause" and, optionally, "Specific Problem" values will provide ITU-T standard or GSM or manufacturer specific information that will help to localise the fault to a specific replaceable/repairable unit.

6.2.2 Test management functions

Tests from all the categories presented (see subclause 5.4.3) may be used for fault localisation purposes depending on the unit suspected to be faulty. The following functions support testing for fault localisation:

- Controlled Test Request function (see subclause 6.4.1.1);
- Uncontrolled Test Request function (see subclause 6.4.1.2);
- Resume/suspend Test function (see subclause 6.4.1.3);
- Terminate Test function (see subclause 6.4.1.4);
- Test Result function (see 6.4.1.5).

6.3 Fault correction functions

The fault correction functions identify the following mechanisms for the management of fault correction:

Add Redundancy Relationship: The NE is requested to create the redundancy relationship. The NE responds with an acknowledgement of the request.

Remove Redundancy Relationship: The NE is requested to remove the redundancy relationship. The NE responds with an acknowledgement of the request.

Change Over: The NE is requested to initiate the action that results in the secondary resource taking over the primary role as defined by the redundancy relationship. The NE responds with an acknowledgement of the request.

Change Back: The NE is requested to initiate the action that results in the restoration of the resources into their original roles as defined by the redundancy relationship. The NE responds with an acknowledgement of the request.

Request Redundancy Relationship Condition: The OS requests the NE to send the current condition of the redundancy relationship. The NE responds with the current condition of the redundancy relationship.

Condition Redundancy Relationship: The NE is requested to assign the characteristics of the redundancy relationship as specified by the OS, or to initiate or terminate the redundancy relationship.

Report Redundancy Relationship Condition: The NE informs the OS of the current characteristics of the redundancy relationship; the report may be a result of an autonomous NE modification of the redundancy relationship".

6.3.1 OS controlled fault correction

The OS may control redundancy to execute a Change Over or a Change Back. This "on demand" control by the OS may be performed in the following manner:-

- using state management services to lock or unlock one or more objects in the redundancy.

  ITU-T Recommendation X.731 specifies the lock and unlock state management functions referred to in this section.

  The use of the lock and unlock state management functions to trigger a Change Over or a Change Back between two managed objects in a defined redundancy relationship is performed in the following manner. Assume that managed object "A" is defined as a primary object in the redundancy relationship, and managed object "B" is defined as the secondary object in the redundancy relationship. In order to use state management functions to effect a change over, the OS first locks the primary resource (managed object "A"), preventing it from performing its functions. The NE then proceeds to use the secondary resource (managed object "B") to take over the functions of the primary resource (managed object "A"). The two resources remain in these roles until their original roles are re-instated as described below.

  State management functions may be subsequently used to revert the primary and secondary resources to their original roles. In order to use state management functions to effect a change back, the OS first unlocks the primary resource (managed object "A"), enabling it to perform functions. Secondly, the OS then locks the secondary resource (managed object "B"), preventing it from performing its functions. The NE then proceeds to use the primary resource (managed object "A") to take over the functions of the secondary resource (managed object "B"), resulting in the primary resource again performing the functions it was originally doing. The two resources are then back in their original roles.

- explicit use of the specific actions for the protectionGroup MOC defined in G.774.03 (see also table 2).

  The Change Over and Change Back fault correction functions identified in sub-clause 6.3 may be requested by the OS.

6.3.2 Autonomous fault correction

The Change Over, Change Back and Condition Redundancy Relationship fault correction functions identified in sub-clause 6.3 may be performed autonomously by the NE without OS intervention.

The means by which the NE internally gains access to these fault correction functions is outside the scope of the present document, but the effects of using these correction functions by the agent shall be notified to the OS (refer to Table 2).

6.3.3 Fault correction related basic services

This subclause describes the mapping between the management functions, managed objects and the 'basic services' needed to support the management functions defined in subclauses 6.3, 6.3.1 and 6.3.2. The mapping of the basic services to the supporting CMIS services is also presented.

The mapping of the following basic services to the confirmed or non-confirmed mode of the supporting CMIS services, except where specified, is a local implementation issue and is not specified in the present document.

Table 2: Fault correction functions, managed objects and basic services

Function(s)	Managed Object(s)	Basic Service(s)	CMIS service(s)
Add Redundancy Relationship	protectionGroup protectionUnit	Create protectionGroup (G.774.03) Create protectionUnit (G.774.03)	M-CREATE M-CREATE
Remove Redundancy Relationship	protectionGroup protectionUnit	Delete protectionGroup (G.774.03) Delete protectionUnit (G.774.03)	M-DELETE M-DELETE
Change Over	protectionGroup MO* protectionGroup protectionUnit	Invoke Protection (G.774.03) Lock/Unlock MO* (X.731) Protection Switch Reporting (G.774.03) Attribute Value Change protectionUnit (G.774.03)	M-ACTION M-SET M-EVENT-REPORT M-EVENT-REPORT
Change Back	protectionGroup MO* protectionGroup protectionUnit	Release Protection (G.774.03) Lock/Unlock MO* (X.731) Protection Switch Reporting (G.774.03) Attribute Value Change protectionUnit (G.774.03)	M-ACTION M-SET M-EVENT-REPORT M-EVENT-REPORT
Request Redundancy Relationship Condition	protectionGroup protectionUnit	Get protectionGroup (G.774.03) Get protectionUnit (G.774.03)	M-GET M-GET
Condition Redundancy Relationship	protectionGroup protectionUnit protectionUnit	Set protectionGroup (G.774.03) Create protectionUnit (G.774.03) Delete protectionUnit (G.774.03)	M-SET M-CREATE M-DELETE
Report Redundancy Relationship Condition	protectionGroup protectionGroup protectionGroup protectionUnit	State Change protectionGroup (G.774.03) Attribute Value Change protectionGroup (G.774.03) Protection Switch Reporting (G.774.03) Attribute Value Change protectionUnit (G.774.03)	M-EVENT-REPORT M-EVENT-REPORT M-EVENT-REPORT M-EVENT-REPORT

NOTE: * MO means a managed object pointed to by a protectionUnit managed object instance

6.4 Test management functions

The management functions for the testing of the NE are closely based on the ITU-T recommendations X.745 [13] and ITU-T X.737 [12].

ITU-T X.745 includes:

- the model on which these management functions are based (the same model described in the previous clause);
- the definitions used in this context;
- a list of management functions (also named service definitions);
- definitions of two functional units;
- the protocol services that are necessary for these functions;
- a definition of a first nucleus of Information Model (GDMO and ASN.1 syntax) associated with this model.

The testing management functions consist of actions and notifications provided to manage the tests (how to start, stop, suspend, etc. the supported tests). In order to fully define these management functions, it is necessary to specify both the characteristics of the tests supported by the NE, and how they can be managed from the OS.

6.4.1 Functions

This subclause defines the management functions required for test management. Each of the management functions can be to applied to the tests in one or more of the test categories defined in 5.4.3 Test Categories.

6.4.1.1 Controlled Test Request function

This function allows the test conductor (on the managing system) to send a request to the test performer (in the managed NE), to execute a controlled test.

The test response does not include the test results which may be returned later, using another function, or may be put in the TO's test result attribute, so that they can be retrieved by the managing system.

6.4.1.2 Uncontrolled Test Request function

Like the previous function, this one allows the test conductor to send a request to the test performer to execute an uncontrolled test.

The test response shall include the test results.

6.4.1.3 Resume/suspend Test function

This function allows the test conductor to send a request to the test performer to suspend or resume a single test or all the tests of a session. It is applicable only to controlled tests.

The test response shall include the state of the affected TOs, as it is immediately before the suspension or immediately after the resumption.

6.4.1.4 Terminate Test function

This function allows the test conductor to send a request to the test performer to terminate a single test or all the tests of a session. It is applicable only to controlled tests.

6.4.1.5 Test Result function

This function allows the managed NE to return the test results of controlled tests to the managing OS. The test results are produced as unsolicited events; however they shall include the test invocation identifier which relates them to the original test request.

6.4.1.6 Scheduling Conflict Report function

This function allows the test performer to report a test schedule conflict to the test conductor. This function is provided only when the NE provides the scheduling capability (which is optional).

6.4.2 Test management related basic services

This subclause summarises in a table the mapping between the management functions defined in subclause 6.4.1, the managed object classes and the OSI management services (called 'basic services' in the present document), needed to support the management functions. The mapping of the basic services to the supporting CMIS services is presented as well. Note that for the testing, the functions to achieve the service are equivalent to the basic services. For the testing service component the OSI management functions are sufficient and there is no need to define new specific telecommunications management functions.

Table 3: Test management functions, managed objects and basic services

Function(s)	Managed Object (s)	Basic Service(s)	CMIS service(s)
Controlled Test Request	MO(TARR)*, MORT, AO**, TO	Controlled Test Request	M-ACTION
Uncontrolled Test Request	MO(TARR)*, MORT, AO**	Uncontrolled Test Request	M-ACTION
Suspend/Resume Test	MO(TARR)*, MORT, AO**, TO	Suspend/Resume Test	M-ACTION
Terminate Test	MO(TARR)*, MORT, AO**, TO	Terminate Test	M-ACTION
Test Result	MO(TARR)*, MORT, AO**, TO	Test Result	M-EVENT-REPORT
Scheduling Conflict Report	MO(TARR)*, MORT, AO**, TO	Scheduling Conflict Report	M-EVENT-REPORT

* MO(TARR) means an MO of the NE having the TARR functionality.
** AOs are always optional.

The Test Management may also use services, like PT-GET and PT-SET to retrieve and modify the attributes of the MOs involved in the tests, and PT-DELETE to abort the controlled tests. Refer to X.745 [13] for further details.

7 BSS specific fault management functions

While many of the objectives and requirements for Fault Management of the BSS can be regarded as common to several network elements, some may be specific to the BSS. The reason may be due to the BSS being a system supporting radio-based communications or because the BSS is, itself, a distributed network element.

Thus, in addition to the general requirements for fault management capabilities specified in the previous clauses, there are some BSS specific requirements that shall be supported. These are specified in the following subclauses.

7.1 BSS specific alarm surveillance functions

There is no functionality for alarm surveillance removed from or added to the BSS compared to what is identified in the previous clauses. It is expected that additional BSS specific functions will be addressed in a future GSM Phase 2+ version of this document.

BSS specific probable cause values in table 4 of Annex A should be used as far as possible for all types of BSS alarm reports. For more details regarding the probable causes, please refer to Annex A.2.

7.2 BSS specific fault localisation functions

Alarm and/or test failure notifications shall contain information localising the failure to at least the BSC, BTS, TRX, Channel, TRAU, or Abis connection resource that has failed.

7.3 BSS specific fault correction functions

The recovery actions related to software faults are covered by software management functions defined in GSM 12.06 [21] and GSM 12.20 [22].

In the case of a BCCH or SDCCH failure, the BSS should attempt to automatically recover the lost broadcast channel(s), even in the event that there are no backup facilities. This automatic recovery should cover the loss of the specific timeslot and, in the case of BCCH, the loss of another timeslot on the same frequency (for frequency hopping systems). The automatic recovery should avoid utilisation of active traffic channels if possible. If it is not possible to avoid loss of these active traffic channels, an attempt may be made to handover these calls. For SDCCH recovery it should not be the case that the result is the loss of all traffic channels. Some sets of traffic channels shall be preserved for such recovery to be useful.

If the BSS takes any automatic recovery action, the BSS shall notify the OS of the changes that were made.

If automatic recovery has been configured and enabled by the OS, it will take place even if communication with the OS has failed when the recovery is required.

Automatic recovery mechanisms may take system optimisation into account. For example, terrestrial circuits may be blocked to account for failed air interfaces.

The BSS shall support a request from the OS to initiate handover for fault management reasons. The request indicates which device (radio channel, carrier) is affected. The request could optionally indicate the preferred handover destination on the indicated device: the same BSS or preferred BSS. Calls in the clearing phase will have the clearing completed. If a preferred handover destination is indicated by the operator, this information is also given to the MSC together with the necessary parameters given by the MS. This maintenance initiated handover is similar to the normal handover procedure, but with the possible difference that a worse destination is accepted, since the call otherwise would be lost. If the maintenance initiated handover is impossible, the OS will be informed.

Under some failure conditions, the BSC may send the RESET message to the MSC in order to clear all calls. All internal system references have to be released (for more information, see GSM 08.08 [17]).

In order to have a resource in a well-defined state, the operator shall be able to shut down and lock the resource before any fault management action is taken upon it. The OS sends a request to the BSS indicating the BSS resources to be shut down. The BSS locks the indicated resource immediately if no traffic is on it. If there is traffic on the specified resource, then the lock is delayed until all traffic is released or until a lock is requested by the OS.

The circuits of the BSS-MSC interface are locked by a request sent from the OS either to the BSS or MSC, which then performs the blocking of circuits described in GSM 08.08.

After successful repair/replacement actions have been taken on a resource, it shall be restored to service. If the service personnel is able to request this locally at the BSS, the BSS shall inform the OS indicating the status of the resource concerned.

7.4 BSS specific testing function

It is expected that BSS specific testing functions will be addressed in a future GSM phase 2+ version of this document.

7.5 BSS-OS communication failure

A temporary buffering capability shall be provided by the BSS. This decreases the negative effects of communication failures between the NE and OS and peaks of message flow occurring during normal operation of the NE. For these reasons, the following recommendations and requirements apply to the BSS:

- During OS-NE communication link disruption, the BSS should continue to provide the telecommunication services as far as possible. This implies that failures on the BSS shall be recovered autonomously by the BSS itself, without any support from the OS. During this time, alarm reports, state change reports and any other event reports shall be temporarily stored and automatically forwarded to the managing OS later, when the communication link is restored.

- When the communication is restored, the BSS shall provide all the information necessary to the managing OS to regain control of the BSS. This means that the BSS shall provide general information about lost messages.

- It is expected that the management aspects of handling the BSS-OS communications link failure will be addressed in a future GSM phase 2+ version of this document.

Annex A (normative): Management Information Model

A.1 Management Information Model

A.1.1 Object Classes

A.1.1.1 thresholdManager

```
thresholdManager MANAGED OBJECT CLASS
DERIVED FROM "ITU-T Rec. X.738 | ISO/IEC 10164-13:1995":metricScanner;
CHARACTERIZED BY
thresholdManagerPackage PACKAGE
    BEHAVIOUR
    thresholdManagerBehaviour BEHAVIOUR
    DEFINED AS "The thresholdManager MOC permits the management of counter and
                gauge thresholds, in order to generate defined alarm
                notifications as a result of a value change crossing the
                threshold level of a counter or gauge.
                The behaviour of the thresholdManager MOC deviates from the
                behaviour of the metricScanner (from which it is derived) in
                that it does not emit the Scan Report notification.
                If the conditional managedObjectInstanceSelectionPackage is
                used by the OS, then it shall refer to the observed object
                instances and not to measurement function object instances.
                See GSM 12.11 clause 5.1.2.1";;
    ATTRIBUTES
    thresholdingStatus   GET-REPLACE;
    NOTIFICATIONS
    thresholdAlarm;
CONDITIONAL PACKAGES
    counterThresholdPackage  PRESENT IF "counter threshold is required and
                                gaugeThresholdPackage is not present ",
    gaugeThresholdPackage  PRESENT IF   "gauge threshold is required and
                                counterThresholdPackage is not present ",
REGISTERED AS {GSM1211TypeModule.gsm1211ObjectClass 10};

counterThresholdPackage PACKAGE
    BEHAVIOUR
    counterThresholdPackageBehaviour BEHAVIOUR
    DEFINED AS  "This package supports the management of counter thresholds.
                See GSM 12.11 clause 5.1.2.2";;
    ATTRIBUTES
    counterThreshold GET-REPLACE;
REGISTERED AS {GSM1211TypeModule.gsm1211Package 10};

gaugeThresholdPackage PACKAGE
    BEHAVIOUR
    gaugeThresholdPackageBehaviour BEHAVIOUR
    DEFINED AS  "This package supports the management of gauge thresholds.
                See GSM 12.11 clause 5.1.2.3";;
    ATTRIBUTES
    gaugeThreshold GET-REPLACE;
REGISTERED AS {GSM1211TypeModule.gsm1211Package 20};
```

A.1.1.2 thresholdAlarmRecord

```
thresholdAlarmRecord MANAGED OBJECT CLASS
DERIVED FROM "ITU-T Rec. X.721: 1992": eventLogRecord;
CHARACTERIZED BY
thresholdAlarmRecordPackage PACKAGE
    BEHAVIOUR
    thresholdAlarmRecordBehaviour BEHAVIOUR
    DEFINED AS " This MOC specifies the format of the thresholdAlarm.
                The identifier value for the eventType attribute shall be thresholdAlarm ";;
    ATTRIBUTES
    observedObjectInstance                      GET,
    observedObjectClass                         GET,
    counterOrGaugeIdentifier                    GET,
    thresholdLevel                              GET,
    "ITU-T Rec. X.721:1992":perceivedSeverity   GET,
```

```
    "ITU-T Rec. X.721:1992":probableCause       GET;
CONDITIONAL PACKAGES
    "ITU-T Rec. X.721:1992":notificationIdentifierPackage   PRESENT IF "the     notification
Identifier parameter is present in the notification",
    "ITU-T Rec. X.721:1992":correlatedNotificationsPackage PRESENT IF "the correlatedNotifications
parameter is present in the notification",
    "ITU-T Rec. X.721:1992":additionalTextPackage PRESENT IF "the Additional text parameter is
present in the notification",
    "ITU-T Rec. X.721:1992":additionalInformationPackage PRESENT IF "the Additional information
parameter is present in the notification ";
REGISTERED AS {GSM1211TypeModule.gsm1211ObjectClass 20};
```

A.1.2 Attributes

A.1.2.1 thresholdingStatus

```
thresholdingStatus  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX ThresholdingStatus;
    MATCHES FOR EQUALITY ;;
    BEHAVIOUR
    thresholdingStatusBehaviour BEHAVIOUR
    DEFINED AS "This attribute identifies whether the thresholding mechanism is
                in the activated or deactivated state.
                See GSM 12.11 clause 6.1.1";;
REGISTERED AS {GSM1211TypeModule.gsm1211Attribute 10};
```

A.1.2.2 counterThreshold

```
counterThreshold    ATTRIBUTE
    WITH ATTRIBUTE SYNTAX CounterThreshold;
    MATCHES FOR EQUALITY ;
    BEHAVIOUR
    counterThresholdBehaviour BEHAVIOUR
    DEFINED AS "This attribute identifies the characteristics of a counter
                threshold (such as levels, severity and the notification
                generation switch).  See GSM 12.11 clause 6.1.1";;
REGISTERED AS {GSM1211TypeModule.gsm1211Attribute 20};
```

A.1.2.3 gaugeThreshold

```
gaugeThreshold   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX GaugeThreshold;
    MATCHES FOR EQUALITY ;
    BEHAVIOUR
    gaugeThresholdBehaviour BEHAVIOUR
    DEFINED AS "This attribute identifies the characteristics of a gauge
                threshold for the high and low values (such as levels, severities
                and notification generation switches).
                See GSM 12.11 clause 6.1.1";;
REGISTERED AS {GSM1211TypeModule.gsm1211Attribute 30};
```

A.1.2.4 observedObjectClass

```
observedObjectClass  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX ObjectClass;
    MATCHES FOR EQUALITY ;
    BEHAVIOUR
    observedObjectClassBehaviour BEHAVIOUR
    DEFINED AS "This attribute identifies the object class of the observed object
                for which a counter or gauge threshold crossing
                resulted in the threshold alarm notification.";;
REGISTERED AS {GSM1211TypeModule.gsm1211Attribute 40};
```

A.1.2.5 observedObjectInstance

```
observedObjectInstance   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX ObjectInstance;
    MATCHES FOR EQUALITY ;
    BEHAVIOUR
    observedObjectInstanceBehaviour BEHAVIOUR
    DEFINED AS "This attribute identifies the object instances of the observed
                object for which a counter or gauge threshold crossing resulted
                in the threshold alarm notification.";;
REGISTERED AS {GSM1211TypeModule.gsm1211Attribute 50};
```

A.1.2.6 counterOrGaugeIdentifier

```
counterOrGaugeIdentifier     ATTRIBUTE
    WITH ATTRIBUTE SYNTAX AttributeId;
    MATCHES FOR EQUALITY ;
    BEHAVIOUR
    counterOrGaugeIdentifierBehaviour BEHAVIOUR
    DEFINED AS "This attribute identifies the counter or gauge which resulted in
                the threshold alarm notification.";;
REGISTERED AS {GSM1211TypeModule.gsm1211Attribute 60};
```

A.1.2.7 thresholdLevel

```
thresholdLevel   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX ObservedValue;
    MATCHES FOR EQUALITY ;
    DEFINED AS "This attribute identifies the level of the counter or gauge
                which resulted in the threshold alarm notification.";;
REGISTERED AS {GSM1211TypeModule.gsm1211Attribute 70};
```

A.1.3 Notifications

```
thresholdAlarm NOTIFICATION
    BEHAVIOUR
    thresholdAlarmBehaviour BEHAVIOUR
    DEFINED AS "This notification is emitted as a result of a value change crossing
                the threshold level of a counter or gauge.
                See GSM 12.11 clause 5.1.2.1";;
    WITH INFORMATION SYNTAX
    ThresholdAlarmInformation
    AND ATTRIBUTE IDS
    observedObjectClass         observedObjectClass,
    observedObjectInstance      observedObjectInstance,
    counterOrGaugeIdentifier    counterOrGaugeIdentifier,
    thresholdLevel              thresholdLevel,
    severity                    "ITU-T Rec. X.721:1992":perceivedSeverity,
        probableCause               "ITU-T Rec. X.721:1992":probableCause,
        notificationIdentifier       ITU-T Rec. X.721:1992":notificationIdentifier,
        correlatedNotifications     "ITU-T Rec. X.721:1992":correlatedNotifications,
        additionalText              "ITU-T Rec. X.721:1992":additionalText,
        additionalInformation       "ITU-T Rec. X.721:1992":additionalInformation;
REGISTERED AS {GSM1211TypeModule.gsm1211Notification 10};
```

A.2 Probable causes

The probable cause is a parameter contained in the alarm notification. It specifies the most probable cause of the alarm. The syntax of the probable cause parameter is defined in ITU-T X.721 [6], and the semantics (use and meaning) in X.733.

The actual value assignments for the possible (and recommended) probable causes for the scope of GSM fault management are defined in X.721/X.733, M.3100 and this annex.

In a network element, the managed objects which generate alarm notifications shall in their respective MOCs' definitions include the relevant alarm notifications defined in ITU-T X.721 or, alternatively, the general packages equipmentRelatedAlarmPackage and/or functionalRelatedAlarmPackage defined in GSM 12.20 [22]. Further, they should include the list of the probable cause values that they use in their behaviour definitions.

The syntax for the definition of the probable causes is a choice of two ASN.1 types: 'global value' and 'local value'.

- The global value type probable cause is defined as a unique object identifier. A list of global values is defined by ITU-T X.721. Additional global values may be defined and registered following the standard procedures defined in ITU-T X.208.

- The local value type probable cause is defined as an integer. To assure the interoperability within the TMN networks, the ITU-T M.3100 [2] has defined standard values for the local value probable cause to be used within the TMN application context, and all additional probable causes defined in any other specifications must use the global value type. The TMN application context defined in ITU-T M.3100 has (by definition) all the capabilities of the Systems management application context defined by ITU-T X.701 [5] and, in addition, the ITU-T M.3100 definition of the local value type probable cause.

Further, some of the generic probable causes defined in X.721/X.733 and M.3100 are suitable for use in GSM/BSS alarm notifications, but these probable causes alone are insufficient to support all the needs of a GSM/BSS network element. Therefore, some more GSM/BSS specific probable causes have been defined in this annex, of the global value type as specified in M.3100.

Finally, all probable causes identified as relevant for use within the scope of GSM fault management, referring to X.721/733, M.3100 or GSM 12.11 definitions, are listed in the table below. The columns 2-4 identify in which specifications these probable causes are defined. The formal value assignments are specified in the ASN.1 module; see section A.3.

Where the probable cause is defined in M.3100 as well as X.721/X.733 or GSM 12.11, then the present document recommends the use of the probable cause from either the X.721/X.733 or GSM 12.11 specification.

Table 4: Probable causes

Probable Cause	GSM 12.11	X.721/ X.733	M.3100	Example Object Classes (See notes 1 and 2)	Alarm Type (see note 3)
A-bis to BTS interface failure	X			lapdLink, pcmCircuit	EQP
A-bis to TRX interface failure	X			lapdLink, pcmCircuit	EQP
Antenna problem	X		X	bts, equipmentMOC	EQP
Battery breakdown	X			bts, basebandTransceiver, radioCarrier, equipmentMOC	EQP
Battery charging fault	X		X	bts, basebandTransceiver, radioCarrier, equipmentMOC	EQP
Clock synchronisation problem	X			bsc, bts, transcoder, equipmentMOC	EQP
Combiner problem	X			bts, equipmentMOC	EQP
Disk problem	X		X	bsc, bts, transcoder, equipmentMOC	EQP
Equipment failure	X			bsc, bts, transcoder, equipmentMOC	EQP
Excessive receiver temperature	X			radioCarrier	EQP
Excessive transmitter output power	X			radioCarrier	EQP
Excessive transmitter temperature	X			radioCarrier	EQP
Frequency hopping degraded	X			bts, frequencyHoppingSystem	EQP
Frequency hopping failure	X		X	bts, frequencyHoppingSystem	EQP
Frequency redefinition failed	X			bts, frequencyHoppingSystem	EQP
I/O device error		X	X	bsc, bts, transcoder, equipmentMOC	EQP
Line interface failure	X			bsc, bts, transcoder, pcmCircuit equipmentMOC	EQP
Link failure	X			bsc, bts, transcoder, pcmCircuit, lapdLink , equipmentMOC	EQP
Loss of synchronisation	X		X	bts, btsSiteManager, transcoder	EQP
Lost redundancy	X		X	bts	EQP
Mains breakdown with battery back-up	X			bts, basebandTransceiver, radioCarrier, equipmentMOC	EQP
Mains breakdown without battery back-up	X			bts, basebandTransceiver, radioCarrier, equipmentMOC	EQP
Power supply failure	X		X	bts, basebandTransceiver, radioCarrier, equipmentMOC	EQP
Processor problem		X		bsc, bts, transcoder, equipmentMOC	EQP
Receiver antenna fault	X		X	radioCarrier	EQP
Receiver failure	X			radioCarrier	EQP
Receiver multicoupler failure	X			radioCarrier	EQP
Receiver problem		X		bsc, bts, transcoder, equipmentMOC	EQP
Reduced transmitter output power	X			radioCarrier	EQP
Signal quality evaluation fault	X		X	bts, basebandTransceiver, radioCarrier	EQP
Timeslot hardware failure	X			lapdlink, transcoder, pcmCircuit, channel	EQP
Timing problem		X		bsc, bts, transcoder, equipmentMOC	EQP
Transceiver problem	X		X	bsc, bts, transcoder, equipmentMOC	EQP
Transcoder problem	X			bsc, bts, transcoder, equipmentMOC	EQP
Transcoder or rate adapter problem	X			transcoder, equipmentMOC	EQP

(continued)

Table 4 (continued): Probable causes

Probable Cause	GSM 12.11	X.721/ X.733	M.3100	Example Object Classes (See notes 1 and 2)	Alarm Type (see note 3)
Transmitter antenna failure	X			radioCarrier	EQP
Transmitter antenna not adjusted	X			radioCarrier	EQP
Transmitter failure	X			radioCarrier	EQP
Transmitter low voltage or current	X			radioCarrier	EQP
Transmitter off frequency	X			radioCarrier	EQP
Transmitter problem		X		bsc, bts, transcoder, equipmentMOC	EQP
Application subsystem failure		X	X	replaceableSoftwareUnit	PROC
Configuration or customisation error		X	X	replaceableSoftwareUnit	PROC
Corrupt data		X		replaceableSoftwareUnit	PROC
CPU cycles limit exceeded		X		replaceableSoftwareUnit, equipmentMOC	PROC
Database inconsistency	X		X	replaceableSoftwareUnit	PROC
File error		X	X	replaceableSoftwareUnit	PROC
File system call unsuccessful	X			replaceableSoftwareUnit	PROC
Input parameter out of range	X			replaceableSoftwareUnit	PROC
Invalid parameter	X		X	replaceableSoftwareUnit	PROC
Invalid pointer	X			replaceableSoftwareUnit	PROC
Message not expected	X			replaceableSoftwareUnit	PROC
Message not initialised	X			replaceableSoftwareUnit	PROC
Message out of sequence	X			replaceableSoftwareUnit	PROC
Out of memory		X	X	replaceableSoftwareUnit, equipmentMOC	PROC
Software error		X	X	replaceableSoftwareUnit	PROC
Software program abnormally terminated		X		replaceableSoftwareUnit	PROC
Storage capacity problem		X		replaceableSoftwareUnit, equipmentMOC	PROC
System call unsuccessful	X			replaceableSoftwareUnit	PROC
Timeout expired	X		X	replaceableSoftwareUnit	PROC
Underlying resource unavailable		X	X	replaceableSoftwareUnit, equipmentMOC	PROC
Variable out of range	X			replaceableSoftwareUnit	PROC
Version mismatch		X	X	replaceableSoftwareUnit, equipmentMOC	PROC
Watch dog timer expired	X			replaceableSoftwareUnit, equipmentMOC	PROC
Cooling system failure	X		X	btsSiteManager, bts, bsc, transcoder	ENV
Enclosure door open		X		btsSiteManager, bts, bsc, transcoder	ENV
External equipment failure	X		X	btsSiteManager, bts, bsc, transcoder	ENV
External power supply failure	X			btsSiteManager, bts, bsc, transcoder	ENV
External transmission device failure	X			btsSiteManager, bts, bsc, transcoder	ENV
Fan failure	X			btsSiteManager, bts, bsc, transcoder	ENV
Fire detected		X		btsSiteManager, bts, bsc, transcoder	ENV
Flood detected		X		btsSiteManager, bts, bsc, transcoder	ENV
High humidity	X			bts, btsSiteManager, bsc, transcoder	ENV
High temperature	X			bts, btsSiteManager, bsc, transcoder	ENV
Humidity unacceptable		X		bts, btsSiteManager, bsc, transcoder	ENV
Intrusion detected	X			btsSiteManager, bts, bsc, transcoder	ENV
Low humidity	X			bts, btsSiteManager, bsc, transcoder	ENV
Low temperature	X			bts, btsSiteManager, bsc, transcoder	ENV
Smoke detected	X			btsSiteManager, bts, bsc, transcoder	ENV
Temperature unacceptable		X		bts, btsSiteManager, bsc, transcoder	ENV

(continued)

Table 4 (concluded): Probable causes

Probable Cause	GSM 12.11	X.721/ X.733	M.3100	Example Object Classes (See notes 1 and 2)	Alarm Type (see note 3)
Bandwidth reduced		X	X	btsSiteManager, bts, bsc, transcoder	QOS
Congestion		X	X	btsSiteManager, bts, bsc, transcoder	QOS
Excessive error rate			X	btsSiteManager, bts, bsc, transcoder	QOS
Excessive response time		X	X	btsSiteManager, bts, bsc, transcoder	QOS
Excessive retransmission rate		X	X	btsSiteManager, bts, bsc, transcoder	QOS
Reduced alarm reporting	X			eventForwardingDiscriminator	QOS
Reduced event reporting	X			eventForwardingDiscriminator	QOS
Reduced logging capability	X		X	log	QOS
System resources overload	X		X	btsSiteManager, bts, bsc, transcoder	QOS
Broadcast channel failure	X		X	bts	COM
Call establishment error		X		bsc	COM
Connection establishment error	X		X	bsc, bts, transcoder	COM
Framing error		X		pcmCircuit, radioCarrier, bts, transcoder	COM
Invalid message received	X		X	btsSiteManager, bts, bsc, transcoder	COM
Invalid MSU received	X			bsc	COM
LAPD link protocol failure	X			lapdLink, btsSiteManager, bts, bsc, transcoder	COM
Local alarm indication	X			pcmCircuit	COM
Local node transmission error		X	X	bsc	COM
Loss of frame		X		pcmCircuit, radioCarrier	COM
Loss of signal		X		pcmCircuit, radioCarrier	COM
Remote alarm indication	X			pcmCircuit	COM
Remote node transmission error		X	X	bsc	COM
Routing failure	X		X	btsSiteManager, bts, bsc, transcoder	COM
SS7 protocol failure	X			bsc	COM
Transmission error	X			btsSiteManager, bts, bsc, transcoder, lapdLink, equipmentMOC, radioCarrier	COM

NOTE 1: In this column, for each probable cause there is a list of one or more managed object classes defined in GSM 12.20 [22] that should use that probable cause. This, however, does not put any limitation on the use of the probable causes. In other words, in the BSS any managed object can use any of the above probable causes, as long as it includes the correct notifications.

NOTE 2: The object class "equipmentMOC" stands for any manufacturer-defined MOC derived from the "gsmEquipment" MOC (defined in 12.20), or it is equivalent to the "gsmEquipment" MOC, if directly used for instantiation.

NOTE 3: In this column, an alarm type is associated with each probable cause. According to ITU-T X.733, there are five alarm types: Communications alarm (COM); Quality of Service alarm (QOS); Processing Error alarm (PROC); Equipment alarm (EQP); Environmental alarm (ENV).

A.3 Abstract Syntax Definitions

```
GSM1211TypeModule {ccitt (0) identified-organization (4) etsi (0) mobileDomain (0) gsm-Operation-
Maintenance (3) gsm-12-11 (11) informationModel (0) asn1Module (2) version1 (1)}

DEFINITIONS IMPLICIT TAGS ::=

BEGIN
    IMPORTS
        gsm-12-11
        FROM GSM-DomainDefinitions {ccitt (0)
        identified-organization (4) etsi (0) mobileDomain (0)
        gsm-Operation-Maintenance (3) gsm-12-30 (30)
        informationModel (0) asn1Module (2)
        gsm-OM-DomainDefinitions (0) version1 (1)}

        ProbableCause, PerceivedSeverity, ObservedValue, AdditionalText,
        AdditionalInformation, NotificationIdentifier, CorrelatedNotifications
        FROM Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3)
        part2(2) asn1Module (2) 1}
```

```
        AttributeId, ObjectInstance, ObjectClass
        FROM CMIP-1 { joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3) }
;

-- EXPORTS Everything


    --  Object Identifiers

        --  Information Model Related Identifiers

    gsm1211InformationModel OBJECT IDENTIFIER ::=
            {gsm1211 informationModel (0)}


    gsm1211ObjectClass      OBJECT IDENTIFIER   ::=
            {gsm1211InformationModel managedObjectClass (3)}

    gsm1211Package          OBJECT IDENTIFIER   ::=
            {gsm1211InformationModel package (4)}

    gsm1211Attribute            OBJECT IDENTIFIER   ::=
            {gsm1211InformationModel attribute (7)}

    gsm1211Notification         OBJECT IDENTIFIER   ::=
            {gsm1200InformationModel notification (10)}


        --  Threshold Management Related Identifiers

ThresholdingStatus  ::= BOOLEAN
-- ThresholdingStatus may have the following values:-
--  True (thresholding mechanism is in the activated state), or
--  False (thresholding mechanism is in the deactivated state)

CounterThreshold ::= SET OF SEQUENCE {
                initialComparisonLevel  INTEGER,
                comparisonLevel         INTEGER,
                offsetValue             INTEGER,
                severity                PerceivedSeverity,
                notififyOnOff           BOOLEAN}
--  notififyOnOff may have the following values:-
--  True (the generation of defined notifications is switched on), or
--  False (the generation of defined notifications is switched off)

GaugeThreshold ::= SET OF SEQUENCE {
                notifyLow               NotifyThreshold,
                notifyHigh              NotifyThreshold}

NotifyThreshold ::= SEQUENCE {
                threshold               ObservedValue,
                severity                PerceivedSeverity,
                notifyOnOff             BOOLEAN }
--  notififyOnOff may have the following values:-
--  True (the generation of defined notifications is switched on), or
--  False (the generation of defined notifications is switched off)

ObservedValue ::= CHOICE {
                integer                 INTEGER,
                real                    REAL }

ThresholdAlarmInformation::= SEQUENCE {
                observedObjectClass         ObjectClass,
                observedObjectInstance      ObjectInstance,
                counterOrGaugeIdentifier    AttributeId,
                thresholdLevel              ObservedValue,
                severity                    PerceivedSeverity,
                    probableCause               ProbableCause,
                    notificationIdentifier      [1]NotificationIdentifier OPTIONAL,
                    correlatedNotifications     [2]CorrelatedNotifications OPTIONAL,
                additionalText              [3]AdditionalText OPTIONAL,
                additionalInformation       [4]AdditionalInformation OPTIONAL }


        --  Probable cause value assignments

    gsm1211ProbableCause OBJECT IDENTIFIER ::=
            {gsm1211InformationModel standardSpecificExtension (0)
             probableCause (0)}


    gsmA-BisToBTSInterfaceFailure ProbableCause ::=
```

```
            globalValue : {gsm1211ProbableCause 1}
gsmA-BisToTRXInterfaceFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 2}
gsmAntennaProblem ProbableCause ::=
            globalValue : {gsm1211ProbableCause 3}
gsmBatteryBreakdown ProbableCause ::=
            globalValue : {gsm1211ProbableCause 4}
gsmBatteryChargingFault ProbableCause ::=
            globalValue : {gsm1211ProbableCause 5}
gsmClockSynchronisationProblem ProbableCause ::=
            globalValue : {gsm1211ProbableCause 6}
gsmCombinerProblem ProbableCause ::=
            globalValue : {gsm1211ProbableCause 7}
gsmDiskProblem ProbableCause ::=
            globalValue : {gsm1211ProbableCause 8}
gsmEquipmentFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 9}
gsmExcessiveReceiverTemperature ProbableCause ::=
            globalValue : {gsm1211ProbableCause 10}
gsmExcessiveTransmitterOutputPower ProbableCause ::=
            globalValue : {gsm1211ProbableCause 11}
gsmExcessiveTransmitterTemperature ProbableCause ::=
            globalValue : {gsm1211ProbableCause 12}
gsmFrequencyHoppingDegraded ProbableCause ::=
            globalValue : {gsm1211ProbableCause 13}
gsmFrequencyHoppingFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 14}
gsmFrequencyRedefinitionFailed ProbableCause ::=
            globalValue : {gsm1211ProbableCause 15}
gsmLineInterfaceFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 16}
gsmLinkFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 17}
gsmLossOfSynchronisation ProbableCause ::=
            globalValue : {gsm1211ProbableCause 18}
gsmLostRedundancy ProbableCause ::=
            globalValue : {gsm1211ProbableCause 19}
gsmMainsBreakdownWithBatteryBackUp ProbableCause ::=
            globalValue : {gsm1211ProbableCause 20}
gsmMainsBreakdownWithoutBatteryBackUp ProbableCause ::=
            globalValue : {gsm1211ProbableCause 21}
gsmPowerSupplyFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 22}
gsmReceiverAntennaFault ProbableCause ::=
            globalValue : {gsm1211ProbableCause 23}
gsmReceiverFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 24}
gsmReceiverMulticouplerFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 25}
gsmReducedTransmitterOutputPower ProbableCause ::=
            globalValue : {gsm1211ProbableCause 26}
gsmSignalQualityEvaluationFault ProbableCause ::=
            globalValue : {gsm1211ProbableCause 27}
gsmTimeslotHardwareFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 28}
gsmTransceiverProblem ProbableCause ::=
            globalValue : {gsm1211ProbableCause 29}
gsmTranscoderProblem ProbableCause ::=
            globalValue : {gsm1211ProbableCause 30}
gsmTranscoderOrRateAdapterProblem ProbableCause ::=
            globalValue : {gsm1211ProbableCause 31}
gsmTransmitterAntennaFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 32}
gsmTransmitterAntennaNotAdjusted ProbableCause ::=
            globalValue : {gsm1211ProbableCause 33}
gsmTransmitterFailure ProbableCause ::=
            globalValue : {gsm1211ProbableCause 34}
gsmTransmitterLowVoltageOrCurrent ProbableCause ::=
            globalValue : {gsm1211ProbableCause 35}
gsmTransmitterOffFrequency ProbableCause ::=
            globalValue : {gsm1211ProbableCause 36}
gsmDatabaseInconsistency ProbableCause ::=
            globalValue : {gsm1211ProbableCause 37}
gsmFileSystemcallUnsuccessful ProbableCause ::=
            globalValue : {gsm1211ProbableCause 38}
gsmInputParameterOutOfRange ProbableCause ::=
            globalValue : {gsm1211ProbableCause 39}
gsmInvalidParameter ProbableCause ::=
            globalValue : {gsm1211ProbableCause 40}
gsmInvalidPointer ProbableCause ::=
            globalValue : {gsm1211ProbableCause 41}
gsmMessageNotExpected ProbableCause ::=
            globalValue : {gsm1211ProbableCause 42}
```

```
    gsmMessageNotinitialized ProbableCause ::=
                globalValue : {gsm1211ProbableCause 43}
    gsmMessageOutOfSequence ProbableCause ::=
                globalValue : {gsm1211ProbableCause 44}
    gsmSystemCallUnsuccessful ProbableCause ::=
                globalValue : {gsm1211ProbableCause 45}
    gsmTimeoutExpired ProbableCause ::=
                globalValue : {gsm1211ProbableCause 46}
    gsmVariableOutOfRange ProbableCause ::=
                globalValue : {gsm1211ProbableCause 47}
    gsmWatchDogTimerExpired ProbableCause ::=
                globalValue : {gsm1211ProbableCause 48}
    gsmCoolingSystemFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 49}
    gsmExternalEquipmentFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 50}
    gsmExternalPowerSupplyFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 51}
    gsmExternalTransmissionDeviceFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 52}
    gsmFanFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 53}
    gsmHighHumidity ProbableCause ::=
                globalValue : {gsm1211ProbableCause 54}
    gsmHighTemperature ProbableCause ::=
                globalValue : {gsm1211ProbableCause 55}
    gsmIntrusionDetected ProbableCause ::=
                globalValue : {gsm1211ProbableCause 56}
    gsmLowHumidity ProbableCause ::=
                globalValue : {gsm1211ProbableCause 57}
    gsmLowTemperature ProbableCause ::=
                globalValue : {gsm1211ProbableCause 58}
    gsmSmokeDetected ProbableCause ::=
                globalValue : {gsm1211ProbableCause 59}
    gsmExcessiveErrorRate ProbableCause ::=
                globalValue : {gsm1211ProbableCause 60}
    gsmReducedAlarmReporting ProbableCause ::=
                globalValue : {gsm1211ProbableCause 61}
    gsmReducedEventReporting ProbableCause ::=
                globalValue : {gsm1211ProbableCause 62}
    gsmReducedLoggingCapability ProbableCause ::=
                globalValue : {gsm1211ProbableCause 63}
    gsmSystemResourcesOverload ProbableCause ::=
                globalValue : {gsm1211ProbableCause 64}
    gsmBroadcastChannelFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 65}
    gsmConnectionEstablishmentError ProbableCause ::=
                globalValue : {gsm1211ProbableCause 66}
    gsmInvalidMessageReceived ProbableCause ::=
                globalValue : {gsm1211ProbableCause 67}
    gsmInvalidMSUReceived ProbableCause ::=
                globalValue : {gsm1211ProbableCause 68}
    gsmLAPDLinkProtocolFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 69}
    gsmLocalAlarmIndication ProbableCause ::=
                globalValue : {gsm1211ProbableCause 70}
    gsmRemoteAlarmIndication ProbableCause ::=
                globalValue : {gsm1211ProbableCause 71}
    gsmRoutingFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 72}
    gsmSS7ProtocolFailure ProbableCause ::=
                globalValue : {gsm1211ProbableCause 73}
    gsmTransmissionError ProbableCause ::=
                globalValue : {gsm1211ProbableCause 74}


END
```

Annex B (normative): Coding of fileSubType field

This annex provides a definition for the optional transfer of selected alarm records from the NE to the OS in a file, using the ”Simple File Transfer Control” mechanism (as defined in subclause 5.1.4 and 6.1.2), with the option 'typeOfFile' in requestTransferUp action.

In this case:

- fileType field shall be set to the 'alarmRecords' value;
- fileSubType field may optionally be used to select alarm records according to the following criteria:
  - type of alarms (all alarms or outstanding alarms);
  - severity;
  - period of occurence;

The tables below define the coding of fileSubType field.

Table B.1: FileSubType

8	7	6	5	4	3	2	1
Period of occurence				Severity			Type of alarms

Table B.2: Type of alarms

Bit 1	Type of alarms
0	All alarms
1	Outstanding alarms only

Table B.3: Severity

Bits			Severity
4	3	2	
0	0	0	All alarms whatever the PerceivedSeverity
0	0	1	Alarms with PerceivedSeverity 'critical'
0	1	0	Alarms with PerceivedSeverity 'critical' or 'major'
0	1	1	Alarms with PerceivedSeverity 'critical', 'major' or 'minor'

Table B.4: Period of occurence

Bits				Period of occurence
8	7	6	5	
0	0	0	0	All alarms whatever the time of occurence
0	0	0	1	Alarms that occured during the past 15 minutes
0	0	1	0	Alarms that occured during the past hour
0	0	1	1	Alarms that occured during the past day
0	1	0	0	Alarms that occured during the past week

Note: all values other than the ones specified in the above tables are spare values and can be considered as user definable values.

When the optional fileSubType field is omitted, the default value '0' is assumed which means that all alarms are to be retrieved without filtering.

Annex C (informative): Change History

This annex lists all release 97 change requests approved for the present document by ETSI SMG.

SMG#	SMG tdoc	SMG6 tdoc	VERS	CR	RV	PH	CAT	SUBJECT	Resulting Version
s26	98-0334	98p038	4.1.1	A004		R97	B	BSS fault management Modelling and enhancements	6.0.0
s27	98-0665	98p063	6.0.0	A005		R97	F	Correction of the terminology "Basic Services"	6.1.0

History

Document history		
V6.0.0	July 1998	Publication
V6.1.0	October 1998	Publication

ISBN 2-7437-2670-9
Dépôt légal : Octobre 1998