ETSI TS 132 332 V16.0.0 (2020-08)

TECHNICAL SPECIFICATION

**Digital cellular telecommunications system (Phase 2+) (GSM);
Universal Mobile Telecommunications System (UMTS);
LTE;
Telecommunication management;
Notification Log (NL) Integration Reference Point (IRP);
Information Service (IS)
(3GPP TS 32.332 version 16.0.0 Release 16)**

Reference
RTS/TSGS-0532332vg00

Keywords
GSM,LTE,UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - 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

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables 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 ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

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 ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

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

Version x.y.z

where:

    x the first digit:

        1 presented to TSG for information;

        2 presented to TSG for approval;

        3 or greater indicates TSG approved document under change control.

    y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

    z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

The present document is part of a TS-family covering the 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; as identified below:

    TS 32.331 "Notification Log (NL) Integration Reference Point (IRP): Requirements"

    TS 32.332 "Notification Log (NL) Integration Reference Point (IRP): Information Service (IS)"

    TS 32.336 "Notification Log (NL) Integration Reference Point (IRP); Solution Set (SS) definitions"

The present document describes the requirements and information model necessary for Telecommunication Management (TM). The TM principles and TM architecture are specified in 3GPP TS 32.101 [1] and 3GPP TS 32.102 [2].

A communications system is composed of a multitude of Network Elements (NE) of various types and, typically, different vendors, which inter-operate in a co-ordinated manner in order to satisfy the network users' communication requirements.

The occurrence of faults in an NE may cause deterioration or loss of this NE's function. Fault Management is the functional area, which allows the operator to detect the occurrence of faults in the network in real-time. Configuration Management and Performance Management are two more functional areas, which require the operator to be alerted to certain conditions in the network.

A standard general-purpose mechanism for the management of logs containing selected notifications from the network is required to provide an ability to perform historical analysis on faults and conditions, which occurred in the network. The TS 32.33x-series, constituting the Notification log IRP, sets forth such a mechanism - and the present document contains the IS definition.

1 Scope

The present document specifies the Information Service for the Notification Log Integration Reference Point (NLIRP) as it applies to Itf-N.

This IRP IS defines the semantics of operations (and their parameters) visible across the Itf-N in a protocol and technology neutral way. It does not define the syntax or encoding of the operations and their parameters.

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. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1]   3GPP TS 32.101: "Telecommunication management; Principles and high level requirements".

[2]   3GPP TS 32.102: "Telecommunication management; Architecture".

[3]   3GPP TS 32.302: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Information Service (IS)".

[4]   3GPP TS 32.622: "Telecommunication management; Configuration Management (CM); Generic network resources Integration Reference Point (IRP): Network Resource Model (NRM)".

[5]   3GPP TS 32.111-2: "Telecommunication management; Fault Management; Part 2: Alarm Integration Reference Point (IRP): Information Service (IS)".

[6]   3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management; Information Service (IS)".

[7]   3GPP TS 32.331: "Telecommunication management; Notification Log (NL) Integration Reference Point (IRP): Requirements".

[8]   3GPP TS 32.342 "Telecommunication management; File Transfer (FT) Integration Reference Point (IRP): Information Service (IS)".

[9]   3GPP TS 32.150: "Telecommunication management; Integration Reference Point (IRP) Concept and definitions".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 32.101 [1], 3GPP TS 32.102 [2], 3GPP TS 32.111-2 [5] and 3GPP TS 32.331 [7] apply.

notification category: it refers to the set of notifications of one 3GPP IRP Information Service specification
A Notification Category is identified by the name of the IRP specification and the IRP specification version number.

3.2 Abbreviations

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

DN   Distinguished Name
EM   Element Manager
FT   File Transfer
IOC   Information Object Class
IRP   Integration Reference Point
M   Mandatory
NE   Network Element
NL   Notification Log
NLIRP   Notification Log Integration Reference Point
NM   Network Manager
O   Optional
UML   Unified Modelling Language

4   System overview

The general definition of the System Context for the present IRP is found in 3GPP TS 32.150 [9] clause 4.7.

In addition, the set of related IRP(s) relevant to the present IRP is shown in the two diagrams below
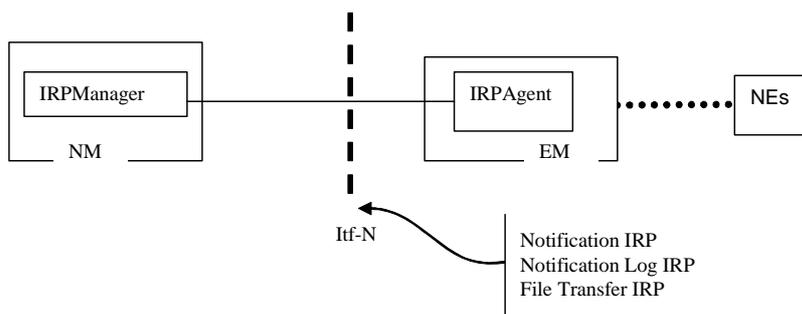
Figure 4.1: System Context A
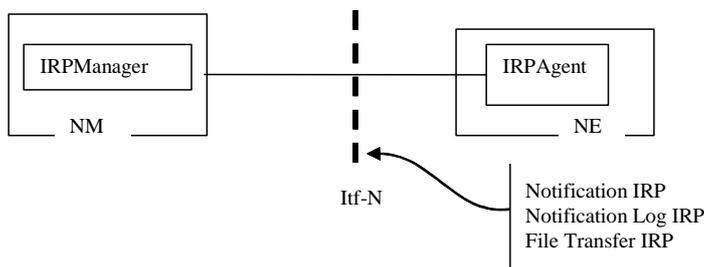
Figure 4.2: System Context B

4.2   Compliance rules

For general definitions of compliance rules related to qualifiers (Mandatory/Optional/Conditional) for *operations*, *notifications and parameters* (of operations and notifications) please refer to 3GPP TS 32.102 [2].

5   Information Object Classes (IOCs)

5.1   Information entities imported and local labels

Label reference	Local label
3GPP TS 32.622 [4], information object class, `Top`	`Top`
3GPP TS 32.312 [6], information object class, `ManagedGenericIRP`	`ManagedGenericIRP`
3GPP TS 32.342 [8], information object class, `FileTransferIRP`	`FileTransferIRP`
3GPP TS 32.302 [3], information object class, `NotificationIRP`	`NotificationIRP`
3GPP TS 32.302 [3], information object class, `NotificationIRPNotification`	`NotificationIRPNotification`

5.2   Class diagram

5.2.1   Attributes and relationships

This clause introduces the set of IOCs that encapsulate information within the `IRPAgent`. The intent is to identify the information required for NLIRP operations and notification. This clause provides the overview of all support object classes in UML. Subsequent clauses provide more detailed specification of various aspects of these support object classes.
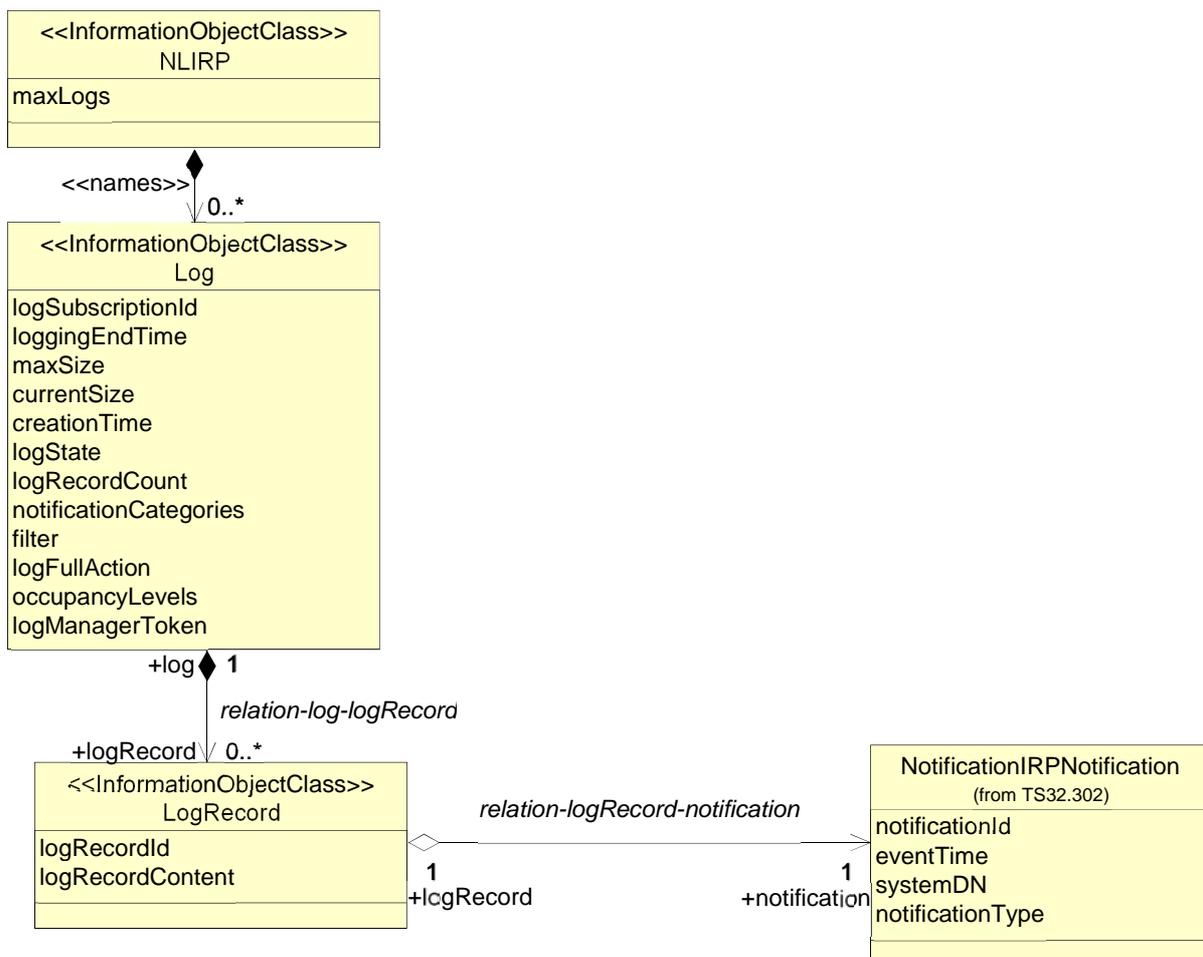
Figure 5.1: Information Object Class (IOC) UML diagram

To change: change logid, add logrecordcontent, remove logrecordsource.

5.2.2   Inheritance
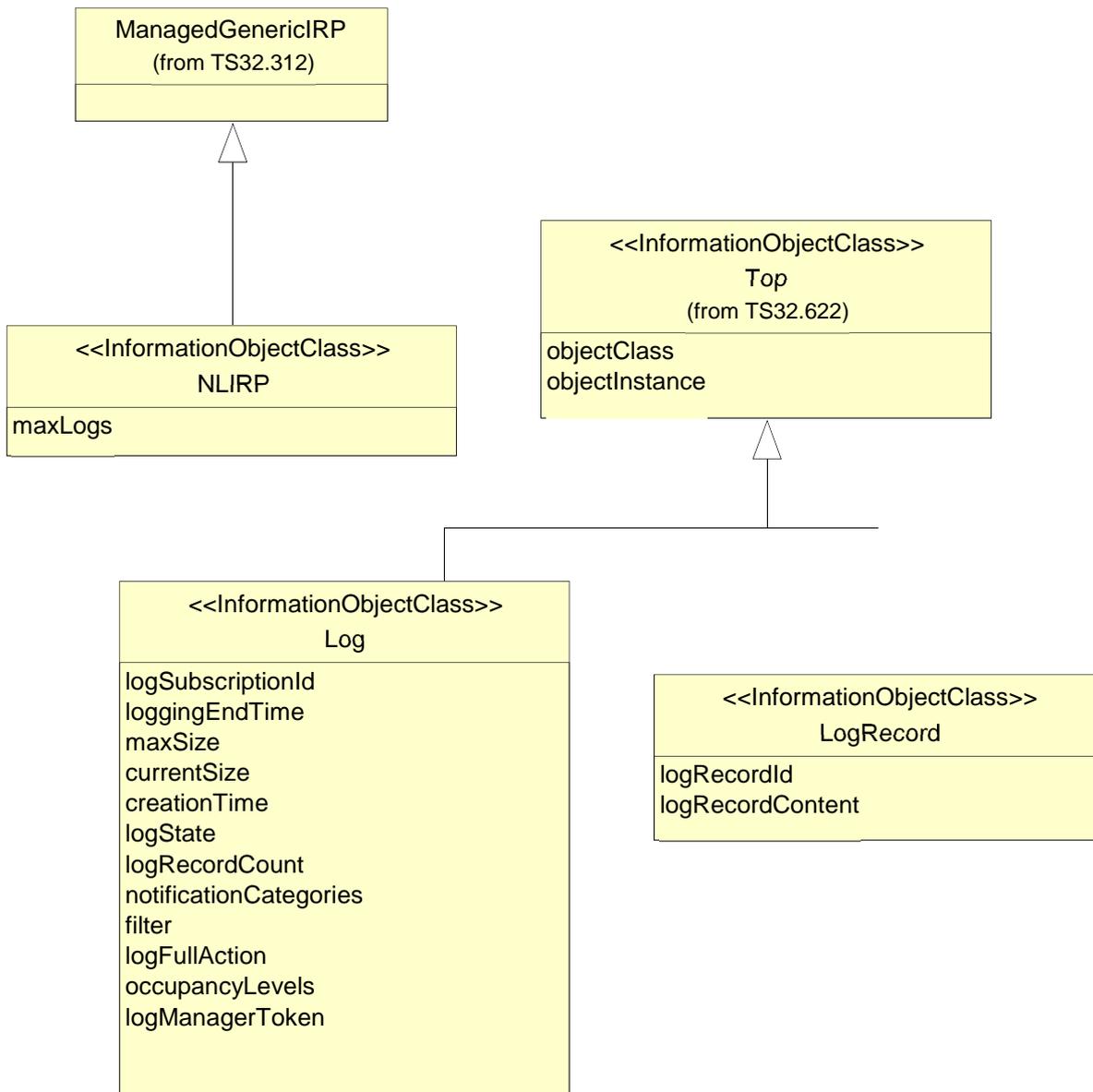
Figure 5.2: Information Object Class (IOC) inheritance UML diagram

5.3 Information Object Class (IOC) definitions

5.3.1 NLIRP

5.3.1.1 Definition

`LogIRP` is the representation of the notification log management capabilities specified by the present document. This IOC inherits from `ManagedGenericIRP` IOC specified in TS 32.312 [6].

5.3.1.2 Attributes

Attribute name	Support Qualifier	Read Qualifier	Write Qualifier
maxLogs	O	M	–

5.3.2 Log

5.3.2.1 Definition

The Log IOC is the representation of a Notification Log.

5.3.2.2 Attributes

Attribute name	Support Qualifier	Read Qualifier	Write Qualifier
logSubscriptionId	M	M	–
loggingEndTime	O	M	–
logManagerToken	O	–	–
maxSize	O	M	–
currentSize	O	M	–
creationTime	O	M	–
logState	M	M	–
logRecordCount	O	M	–
notificationCategories	O	M	–
filter	O	M	–
logFullAction	M	M	–
occupancyLevels	M	M	–

5.3.2.3   State diagram
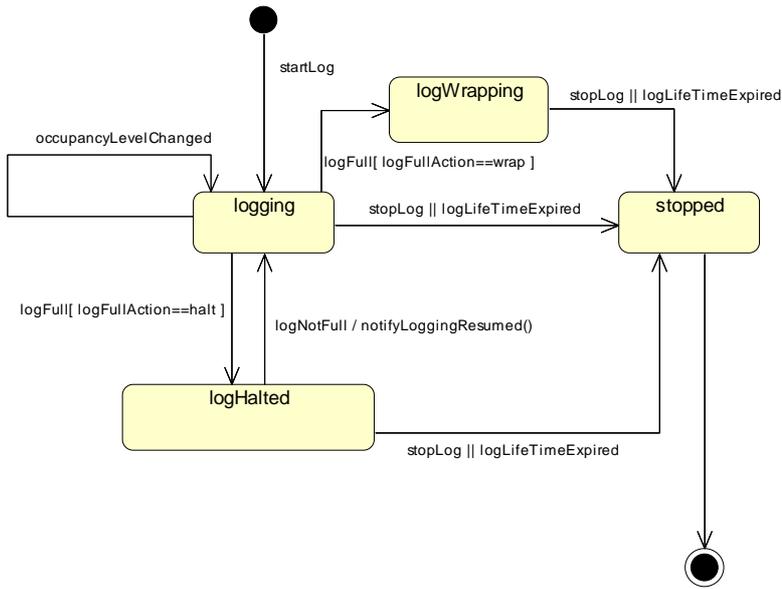
Figure 5.3: State diagram for Notification Log

The disposition of a log that has been stopped, that is, whether the log remains visible across the Itf-N, is left as vendor specific functionality. The time of the deletion of logs is vendor specific.

5.3.3   LogRecord

5.3.3.1   Definition

The LogRecord IOC is the representation of an individual Notification Log Record.

Attribute name	Support Qualifier	Read Qualifier	Write Qualifier
logRecordId	M	M	–
logRecordContent	O	M	–

5.4 Information relationship definitions

5.4.1 Relation-nLIRP-log (M)

5.4.1.1 Definition

This represents the relationship between `NLIRP` and the `Log`.

5.4.1.2 Role

Name	Definition
`nLIRP`	It represents the `NLIRP`.
`log`	It represents the `Log`.

5.4.1.3 Constraint

Name	Definition
`uniqueLogSubscriptionId`	The log subscription id must be unique amongst all logs managed by a given NLIRP instance.
`uniqueLogManagerToken`	The log manager token must be unique amongst all managers and logs utilizing logging services from a given `NLIRP` instance.

5.4.2 Relation-log-logRecord (M)

5.4.2.1 Definition

This represents the relationship between `Log` and the `LogRecord`.

5.4.2.2 Role

Name	Definition
`log`	It represents the `Log`.
`logRecord`	It represents the `LogRecord`.

5.4.2.3 Constraint

Name	Definition
`uniqueLogRecordId`	The log record id must be unique amongst all logs records within a given log.

5.4.3 Relation-logRecord-notificationIRPNotification (M)

5.4.3.1 Definition

This represents the relationship between `LogRecord` and the notification header represented by `NotificationIRPNotification`.

5.4.3.2 Role

Name	Definition
logRecord	It represents the `LogRecord`.
notification	It represents the `NotificationIRPNotification`.

5.4.3.3 Constraint

Name	Definition
logRecordIdRelatesNotificationId	Within a given log, there is a one-to-one relationship between Notification Id and Log Id (as each notification can only be recorded once).

5.5 Information attribute definition

5.5.1 Definition and legal values

Attribute Name	Definition	Legal Values
creationTime	The time when the log is created.	YYYYMMDDhhmmss
currentSize	This attribute provides the number of bytes currently utilized by a given log. When taken in conjunction with maxSize, the amount of space remaining in the log can be determined.	Either: - zero - a positive whole number
filter	It specifies a filter constraint that IRPAgent shall use to filter notification of the category specified in notificationCategories parameter. Support of time based filter is mandatory. Support of other filter constraints is optional.	Filter constraint grammar is SS dependent
logFullAction	Indicate the action that will be taken by this instance of NLIRP when the Log.maxSize has been reached.	An ENUM that can have one of the following values: - wrap: The oldest LogRecord(s) in the Log, based on the log time, will be deleted to free resources for the logging of new LogRecord(s). - halt: No more LogRecord(s) will be logged and all incoming events are discarded. LogRecord/s already in the Log will be retained.
loggingEndTime	Defines the date and time when the log stops logging.	If log is created by IRPManager: o value provided by the IRP Manager - the IRP Agent may reject the value provided by the IRP Manager in case this value is excessive o in case no value is provided by the IRP Manager then this value is set by the IRP Agent. If log is created by IRPAgent: o value indicates date and time or o value is expressed as "indefinite".
logManagerToken	This attribute contains the token of a manager utilizing logging services from a given NLIRP instance, assigned by the IRPAgent.	The value of this attribute must be unique amongst all managers and logs utilizing logging services from the given NLIRP instance.
logRecordContent	The notification information of a log record, excluding the notification header information.	The content format varies depending on the type of the notification.
logRecordCount	The number of log records currently logged within a given log.	positive whole number, including zero
logRecordId	This attribute contains the id of a log record within a given log, assigned by the IRPAgent.	The value of this attribute must be unique amongst all log record contained by a given log.
logState	Provides an indication of the current state of a specific log	An ENUM that can have one of the following values: o logging o logFull o stopped
logSubscriptionId	This attribute contains the id of a log subscription within a given NLIRP instance, assigned by the IRPAgent.	The value of this attribute must be unique amongst all log subscriptions managed by a given NLIRP instance.
maxLogs	Defines the maximum number of logs that can be supported by a given Notification Log IRP, assigned by the IRPAgent.	Non-zero, positive whole number
maxSize	This attribute defines the maximum number of bytes that may be utilized by a given log, assigned by the IRPAgent.	Either: o Non-zero, positive whole number o Zero indicates no limit on the number of records is set by the IRPAgent (log full handling will provide limitation information)
notificationCategories	Specifies the notification categories that can be recorded within a given log - SET OF (name of IRP, version of IRP)	It identifies one or more Notification Category (see also Definition in clause 3.1)
occupancyLevels	Case Log.logFullAction == 'halt' and Log.maxSize is non-zero: o this attribute contains a list of 3 values fixed (determined by the IRPAgent and never changed) of percentage, of Log.maxLog o the NLIRP will generate notifyOccupancyLevelCrossed to alert IRPManagers about the Log capacity currently used. Case Log.logFullAction == 'wrap': o this attribute contains 1 value fixed (determined by the IRPAgent and never changed) of percentage of Log.maxLog the NLIRP will generate notifyOccupancyLevelCrossed to alert IRPManagers that the specified percentage of the capacity has been written to the wrapping Log.	Case of 'halt': a set of 3 values. Case of 'wrap': 1 value.

5.5.2 Constraints

Name	Definition
`inv_lastModifiedTime`	Time indicated shall be greater than or equal to creationTime
`inv_logRecordCount`	Number indicated shall be less than or equal to maxSize

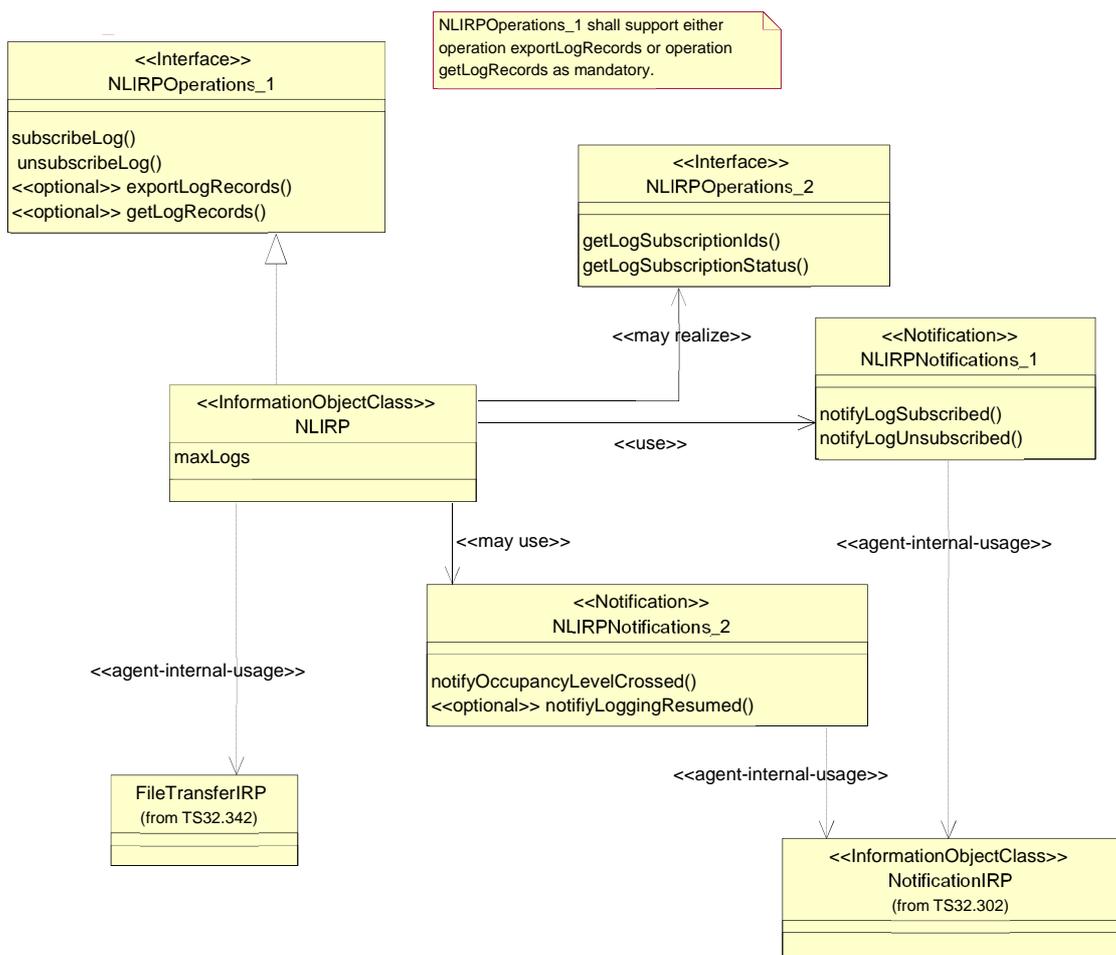# 6 Interface definition

6.1 Class diagram

Figure 6.1: Class diagram

NLIRPOperations_1 shall support either operation exportLogRecords or operation getLogRecords as mandatory.

6.2 Generic rules

Rule 1: Each operation with at least one input parameter supports a pre-condition valid_input_parameter which indicates that all input parameters shall be valid with regards to their information type. Additionally, each such operation supports an exception operation_failed_invalid_input_parameter which is raised when pre-condition valid_input_parameter is false. The exception has the same entry and exit state.

Rule 2: Each operation with at least one optional input parameter supports a set of pre-conditions supported_optional_input_parameter_xxx where "xxx" is the name of the optional input parameter and the pre-condition indicates that the operation supports the named optional input parameter. Additionally, each such operation supports an exception operation_failed_unsupported_optional_input_parameter_xxx which is raised when (a) the pre-condition supported_optional_input_parameter_xxx is false and (b) the named optional input parameter is carrying information. The exception has the same entry and exit state.

Rule 3: Each operation shall support a generic exception operation_failed_internal_problem which is raised when an internal problem occurs and that the operation cannot be completed. The exception has the same entry and exit state.

6.3 NLIRPOperations_1 Interface (M)

6.3.1 Operation subscribeLog (M)

6.3.1.1 Definition

Using this operation, an IRPManager is initiating the logging of notifications. Resulting from this operation an IRPAgent shall start logging of notifications and, if necessary, also create an associated log.

6.3.1.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
logSubscriptionId	M	Log.logSubscriptionId	See clause 5.5.1 If empty, then IRPAgent shall create a new log and return the logSubscriptionId as output parameter. However not all IRPAgents will provide this capability, i.e. IRPManager must provide an existing logSubscriptionId.
loggingEndTime	O	Log.loggingEndTime	See clause 5.5.1
notificationCategories	O	Log.notificationCategories	See clause 5.5.1; if notificationCategories is absent than all notifications are logged
filter	O	Log.ntfFilterfilter Filter constraint grammar is SS dependent	See clause 5.5.1; if this parameter is absent, then no filter constraint shall be applied.

6.3.1.3   Output parameters

Parameter Name	Qualifier	Matching Information	Comment
logSubscriptionId	M	Log.logSubscriptionId	See clause 5.5.1 This parameter contains either o the logSubscriptionId of the log created resulting from this operation or o the value of the input parameter logSubscriptionId.
logManagerToken	O	Log.logManagerToken	See clause 5.5.1 See also comment 6.3.2.2. Note: Security Management IRP may provide capabilities that make this parameter redundant.
loggingEndTime	O	Log.loggingEndTime	See clause 5.5.1 If supported by the Log IOC (5.3.2) and not already provided by the IRP Manager then the output of this operation must support this parameter.
status	M	ENUM (OperationSucceeded, OperationFailed)	If loggingEndTime is valid and notificationCategoriesis valid or absent, status = OperationSucceeded. If operation_failed is true, status = OperationFailed. If loggingEndTime excessive, status = OperationFailed.

6.3.1.4   Pre-condition

logsNotMaxed

Assertion Name	Definition
logsNotMaxed	The number of logs is less than the maximum number of logs allowed.

6.3.1.5   Post-condition

logStarted

Assertion Name	Definition
logStarted	A log is started with the specified characteristics (lifetime and notificationCategories).

6.3.1.6   Exceptions

Exception Name	Definition
logSubscriptionId_required	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.
maxLogs_reached	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.
operation_failed	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.

6.3.2 Operation `unsubscribeLog` (M)

6.3.2.1 Definition

Using this operation, the IRPManager that started a specific log is able to stop this log via unsubscribing. Note stopping a log implies that the log becomes invisible across Itf-N, independent from the loggingEndTime; therefore the IRPManager should retrieve log information of interest before using this `unsubscribeLog` operation, e.g. through the `exportLogRecords` operation.

The disposition of a log that has been stopped, that is, whether the log remains visible across the Itf-N, is left as vendor specific functionality. The time of the deletion of logs is vendor specific.

In consideration of a multi-manager environment, the log can only be stopped by the creating IRPManager if this IRPManager provided a related token while initiating subscribeLog.

6.3.2.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
logSubscriptionId	M	Log.logSubscriptionId	See clause 5.5.1
logManagerToken	O	Log.logManagerToken	See clause 5.5.1 If subscribeLog returned a logManagerToken for a log with logSubscriptionId="X", then logManagerToken must be provided as input parameter of this operation to successfully stop log with logSubscriptionId="X". If subscribeLogdid not return a logManagerToken for a log with logSubscriptionId="X", then logManagerToken must contain NULL value to successfully stop log with logSubscriptionId="X".

6.3.2.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM (OperationSucceeded, OperationFailed)	If logSubscriptionId is valid and logManagerToken (if supported) is matching, status = OperationSucceeded. If operation_failed is true, status = OperationFailed.

6.3.2.4 Pre-condition

There are no pre-conditions, other than those established by the generic rules (see clause 6.2).

6.3.2.5 Post-condition

`logStopped`

Assertion Name	Definition
logStopped	The specified log is stopped.

6.3.2.6 Exceptions

Exception Name	Definition
operation_failed	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.

6.3.3   Operation `exportLogRecords` (O)

Support for this operation is mandatory if getLogRecords is not supported.

6.3.3.1   Definition

Using this operation, an IRPManager can initiate export of all or part of a log into a file. This file then is being transferred to the IRPManager using the File Transfer IRP (note also that the FTIRP is providing the file location to the IRPManager).

6.3.3.2   Input parameters

Parameter Name	Qualifier	Information type	Comment
logSubscriptionId	M	Log.logSubscriptionId	See clause 5.5.1
notificationCategories	O	Log.notificationCategories	See clause 5.5.1
filter	O	Log.filterLog.filter	See clause 5.5.1

6.3.3.3   Output parameters

Parameter Name	Qualifier	Matching Information	Comment
invocationId	M	--	This parameter carries an identifier that NLIRP assigns to identify the request if the status is OperationSucceeded. This parameter carries no information if the status is not OperationSucceeded. One identifier can at most identify one request. NLIRP should not reuse identifier values. This identifier shall also appear in the file name of the exported log file as <specificIRP_extension> (as part specified in Appendix A: File Name Convention of [8]). When the exported log file is ready, FTIRP issues notifyFileReady notification which carries the exported log file name.

6.3.3.4   Pre-condition

There are no pre-conditions, other than those established by the generic rules (see clause 6.2).

6.3.3.5   Post-condition

`logRecordsExported`

Assertion Name	Definition
logRecordsExported	The specified log records have been exported as requested. In case the log is empty or in case that all the log records do not satisfy the criteria of input parameters notificationCategories and filter, this post-condition is true.

6.3.3.6   Exceptions

Exception Name	Definition
export_failed	The NLIRP was unable to export the specified records.
operation_failed	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.

6.3.4 Operation `getLogRecords` (O)

Support for these operations is mandatory if `exportLogRecords` is not supported.

6.3.4.1 Definition

Using this operation, an IRPManager can retrieve one or more log records from a certain log.

Note that this operation might be preferred for retrieval of small amounts of log records, while operation `exportLogRecords` might be preferred for retrieval of medium to large amounts of log records, as providing a more efficient bulk transfer mechanism.

6.3.4.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
`logSubscriptionId`	M	Log.logSubscriptionId	See clause 5.5.1
`notificationCategories`	O	Log.notificationCategories	See clause 5.5.1
`filter`	O	Log.filter	See clause 5.5.1

6.3.4.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
`getLogRecordsResult`	M	List of logRecord's	List of logRecords; each entry of the list holds all supported attributes of IOC LogRecord (see 5.3.3.2).
`status`	M	ENUM (OperationSucceeded, OperationFailed)	If logSubscriptionId is valid and (logRecordIdList is empty or logRecordIdList contains valid Id's), status = OperationSucceeded. If operation_failed is true, status = OperationFailed.

6.3.4.4 Pre-condition

There are no pre-conditions, other than those established by the generic rules (see clause 6.2).

6.3.4.5 Post-condition

`logRecordsRetrieved`

Assertion Name	Definition
`logRecordsRetrieved`	The specified log records have been retrieved as requested. If the log is empty or all the log records do not satisfy the criteria of input parameters notificationCategories and filter, this post-condition is true.

6.3.4.6 Exceptions

Exception Name	Definition
`operation_failed`	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.

6.4 NLIRPOperations_2 Interface (O)

6.4.1 Operation getLogSubscriptionIds (M)

6.4.1.1 Definition

Using this operation, an IRPManager can query the NLIRP for all available log subscriptions.

6.4.1.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
--	--	--	--

6.4.1.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
logSubscriptionIds	M	List of logSubscriptionId's	See clause 5.5.1 & 6.4.1.2. Note that empty list is a valid value.
status	M	ENUM (OperationSucceeded, OperationFailed)	If operation is successful, status = OperationSucceeded. If operation_failed is true, status = OperationFailed.

6.4.1.4 Pre-condition

There are no pre-conditions, other than those established by the generic rules (see clause 6.2).

6.4.1.5 Post-condition

There are no post-conditions. Querying of log subscription Id's does not result in any changes within the IRP Agent.

6.4.1.6 Exceptions

Exception Name	Definition
operation_failed	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.

6.4.2 Operation getLogSubscriptionStatus (M)

6.4.2.1 Definition

Using this operation, an IRPManager can query the NLIRP for available log status information of an individual log.

6.4.2.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
logSubscriptionId	M	Log.logSubscriptionId	See clause 5.5.1

6.4.2.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
logAttributeList	M	Attributes of related Log IOC except logManagerToken	List of all supported attributes of IOC Log except logManagerToken (see clause 5.3.2.2).
status	M	ENUM (OperationSucceeded, OperationFailed)	If logSubscriptionId is valid status = OperationSucceeded. If operation_failed is true, status = OperationFailed.

6.4.2.4 Pre-condition

There are no pre-conditions, other than those established by the generic rules (see clause 6.2).

6.4.2.5 Post-condition

There are no post-conditions. Querying of log attributes does not result in any changes within the IRPAgent.

6.4.2.6 Exceptions

Exception Name	Definition
operation_failed	**Condition:** Pre-condition is true AND post-condition is false. **Returned Information:** The output parameter status. **Exit state:** Entry state.

6.5 NLIRPNotifications_1 Interface (M)

6.5.1 Notification `notifyLogSubscribed` (M)

6.5.1.1 Definition

Using this notification, an IRPAgent informs all subscribed IRPManagers that a log subscription has been successfully initiated.

6.5.1.2 Input Parameters

Parameter Name	Qualifier	Matching Information	Comment
objectClass	M,Y	Log.objectClass	Notification header - see [3]. It shall carry the `Log` class name.
objectInstance	M,Y	Log.objectInstance	Notification header - see [3]. It shall carry the DN of `Log`.
notificationId	M,N	--	Notification header - see [3].
eventTime	M,Y	--	Notification header - see [3].
notificationType	M,Y	"notifyLogSubscribed"	Notification header - see [3].
systemDN	C,Y	--	Notification header - see [3].
logSubscriptionId	M,N	Log.logSubscriptionId	See clause 5.5.1
loggingEndTime	O,N	Log.loggingEndTime	See clause 5.5.1
notificationCategories	O,N	Log.notificationCategories	See clause 5.5.1; if absent than all notifications are being logged
filter	O,N	Log.filter	See clause 5.5.1

6.5.1.3 Triggering Event

6.5.1.3.1 From-state

`subscribeLog`

Assertion Name	Definition
subscribeLog	An IRPManager requests that a new log be started.

6.5.1.3.2 To-state

`logStarted`

Assertion Name	Definition
logStarted	The NLIRP has started the requested logging activity.

6.5.2 Notification `notifyLogUnsubscribed` (M)

6.5.2.1 Definition

Using this notification, an IRPAgent informs all subscribed IRPManager that a log subscription has stopped.

6.5.2.2 Input Parameters

Parameter Name	Qualifier	Matching Information	Comment
objectClass	M,Y	Log.objectClass	See table 6.5.1.2.
objectInstance	M,Y	Log.objectInstance	See table 6.5.1.2.
notificationId	M,N	--	See table 6.5.1.2.
eventTime	M,Y	--	See table 6.5.1.2.
notificationType	M,Y	"notifyLogUnsubscribed"	See table 6.5.1.2.
systemDN	C,Y	--.	See table 6.5.1.2.
logSubscriptionId	M,N	Log.logSubscriptionId	See table 6.5.1.2.

6.5.2.3 Triggering Event

6.5.2.3.1 From-state

`unsubscribeLog` OR `loggingEndTimeReached`

Assertion Name	Definition
unsubscribeLog	The IRPManager that started the log requests that the log be unsubscribed.
loggingEndTimeReached	The logging end timespecified for the log in subscribeLog has been reached.

6.5.2.3.2 To-state

`logStopped`

Assertion Name	Definition
logStopped	The logging activity has stopped.

6.6 NLIRPNotifications_2 Interface (O)

6.6.1 Notification `notifyOccupancyLevelCrossed` (M)

6.6.1.1 Definition

Using this notification, an IRPAgent informs all subscribed IRPManagers about the occupancy level of a certain log according with levels defined in 5.5.1 (in addition also the log full action type of this NLIRP instance is being provided for context purposes).

6.6.1.2 Input Parameters

Parameter Name	Qualifier	Matching Information	Comment
objectClass	M,Y	Log.objectClass	See table 6.5.1.2.
objectInstance	M,Y	Log.objectInstance	See table 6.5.1.2.
notificationId	M,N	--	See table 6.5.1.2.
eventTime	M,Y	--	See table 6.5.1.2.
notificationType	M,Y	"notifyOccupancyLevelCrossed"	See table 6.5.1.2.
systemDN	C,Y	--	See table 6.5.1.2.
logSubscriptionId	M,Y	Log.logSubscriptionId	See table 6.5.1.2.
currentOccupancyLevel	M,N	The integer portion of ((Log.currentSize/ Log.maxSize)*100)	See clause 5.5.1.
logFullAction	O,N	Log.logFullAction	See clause 5.5.1.

6.6.1.3 Triggering Event

6.6.1.3.1 From-state

`logFull` OR `occupancyLevelCrossed`

Assertion Name	Definition
logFull	The log is full, that is, the number of log records contained within the log has exceeded the maximum number of log records that was established at log startup (log capacity has reached 100 %).
occupancyLevelCrossed	The number of log records within the log has crossed one of the threshold boundaries (see 5.5.1 Log.occupancyLevels).

6.6.1.3.2 To-state

`logWrapping OR (logHalted OR logging)`

Assertion Name	Definition
logWrapping	Applicable for case Log.logFullAction='wrap': The number of log records within the log has exceeded the maximum number of allowed log records (100 % capacity has been crossed). The logging of new records will cause the oldest records within the log to be deleted and replaced (first in, first out). The deletion of records may occur in a block, such that the log transitions back to the "started" state prior to any new records being written.
logHalted	Applicable for case Log.logFullAction='halt': The number of log records within the log has exceeded the maximum number of allowed log records (100 % capacity has been crossed). The logging of new records will be discontinued until the deletion of records has occurred (deletion of records is outside the scope of this IRP).
logging	Applicable for case Log.logFullAction='halt': The number of log records within the log is below the maximum number of allowed records. The generation of the notification is done to inform the subscribed IRPManagers that the log is filling up.

6.6.2 Notification notifyLoggingResumed (O)

6.6.2.1 Definition

Using this notification, an IRPAgent informs all subscribed IRPManagers that the amount of data within a given log has been reduced, allowing logging to resume continue (according to the information provided when the log was created). Note that this notification only applies to Log.logFullAction == 'halt'.

6.6.2.2 Input Parameters

Parameter Name	Qualifier	Matching Information	Comment
objectClass	M,Y	Log.objectClass	See table 6.5.1.2.
objectInstance	M,Y	Log.objectInstance	See table 6.5.1.2.
notificationId	M,N	--	See table 6.5.1.2.
eventTime	M,Y	--	See table 6.5.1.2.
notificationType	M,Y	"notifyLoggingResumed"	See table 6.5.1.2.
systemDN	C,Y	--	See table 6.5.1.2.
logSubscriptionId	M,Y	Log.logSubscriptionId	See table 6.5.1.2.

6.6.2.3 Triggering Event

6.6.2.3.1 From-state

logFull

Assertion Name	Definition
logHalted	The log is full, that is, the number of log records contained within the log has exceeded the maximum number of log records that was established at log startup (log capacity has reached 100 % and Log.logFullAction='halt').
occupancyLevelCrossed	The number of log records within the log has crossed one of the threshold boundaries (see 5.5.1 Log.occupancyLevels).

6.6.2.3.2 To-state

Assertion Name	Definition
logging	The number of log records within the log is below the maximum number of allowed records. The generation of the notification is done to inform the subscribed IRPManagers that the log is filling up.

Annex A (informative): Change history

Change history									
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Cat**	**Old**	**New**
Mar 2004	S_23	SP-040123	--	--	Submitted to SA#23 for Information	--	1.0.0	
Dec 2004	S_26	SP-040798	--	--	Submitted to SA#26 for Approval	--	2.0.0	6.0.0
Jun 2005	S_28	SP-050290	0001	--	Correct mapping info of currentOccupancyLevel and some editorial errors	F	6.0.0	6.1.0
Jun 2005	S_28	SP-050329	0002	--	Apply Generic System Context – Align with TS 32.150	F	6.0.0	6.1.0
Mar 2006	SA_31	SP-060091	0003	--	Add invocationId output parameter of exportLogRecords operation	F	6.1.0	6.2.0
Mar 2006	SA_31	SP-060089	0004	--	Correct ambiguity of object class and object instance usage in notification	F	6.1.0	6.2.0
Dec 2006	SA_34	SP-060709	0005	--	Correct the errors in the definition of occupancyLevels and currentOccupancyLevel	F	6.2.0	6.3.0
Jun 2007	SA_36	--	--	--	Automatic upgrade to Rel-7 (no CR) at freeze of Rel-7. Deleted reference to CMIP SS, discontinued from R7 onwards.	--	6.3.0	7.0.0
Sep 2007	SA_37	SP-070614	0006	--	Remove the use of visibility symbol	C	7.0.0	8.0.0
Dec 2009	-	-	-	-	Update to Rel-9 version (MCC)	-	8.0.0	9.0.0
Mar 2011	-	-	-	-	Update to Rel-10 version (MCC)	-	9.0.0	10.0.0
2012-09	-	-	-	-	Update to Rel-11 version (MCC)		10.0.0	**11.0.0**
2014-10	-	-	-	-	Update to Rel-12 version (MCC)		11.0.0	**12.0.0**
2016-01	-	-	-	-	Update to Rel-13 version (MCC)		12.0.0	**13.0.0**
2017-03	SA#75	-	-	-	Promotion to Release 14 without technical change		13.0.0	**14.0.0**

Change history							
Date	**Meeting**	**TDoc**	**CR**	**Rev**	**Cat**	**Subject/Comment**	**New version**
2018-06						Update to Rel-15 version (MCC)	15.0.0
2020-07	-	-	-	-	-	Update to Rel-16 version (MCC)	**16.0.0**

History

Document history		
V16.0.0	August 2020	Publication