

# ETSI TS 132 332 V6.0.0 (2004-12)

---

*Technical Specification*

**Digital cellular telecommunications system (Phase 2+);  
Universal Mobile Telecommunications System (UMTS);  
Telecommunication management;  
Notification Log (NL) Integration Reference Point (IRP):  
Information Service (IS)  
(3GPP TS 32.332 version 6.0.0 Release 6)**

---



---

Reference

DTS/TSGS-0532332v600

---

Keywords

GSM, 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**

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

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

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

[http://portal.etsi.org/chaicor/ETSI\\_support.asp](http://portal.etsi.org/chaicor/ETSI_support.asp)

---

**Copyright Notification**

No part may be reproduced except as authorized by written permission.  
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2004.  
All rights reserved.

**DECT**<sup>TM</sup>, **PLUGTESTS**<sup>TM</sup> and **UMTS**<sup>TM</sup> are Trade Marks of ETSI registered for the benefit of its Members.  
**TIPHON**<sup>TM</sup> and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members.  
**3GPP**<sup>TM</sup> is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

---

## 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 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 (<http://webapp.etsi.org/IPR/home.asp>).

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.

---

## Foreword

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, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

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

# Contents

Intellectual Property Rights .....	2
Foreword.....	2
Foreword.....	6
Introduction .....	6
1 Scope .....	7
2 References .....	7
3 Definitions and abbreviations.....	7
3.1 Definitions .....	7
3.2 Abbreviations .....	8
4 System overview .....	8
4.1 System context .....	8
4.2 Compliance rules.....	9
5 Information Object Classes .....	9
5.1 Information entities imported and local labels .....	9
5.2 Class diagram .....	9
5.2.1 Attributes and relationships .....	9
5.2.2 Inheritance .....	11
5.3 Information Object Class definitions.....	11
5.3.1 NLIRP.....	11
5.3.1.1 Definition .....	11
5.3.1.2 Attributes.....	11
5.3.2 Log.....	12
5.3.2.1 Definition .....	12
5.3.2.2 Attributes.....	12
5.3.2.3 State diagram.....	12
5.3.3 LogRecord .....	12
5.3.3.1 Definition .....	12
5.3.3.2 Attributes.....	13
5.4 Information relationship definitions .....	13
5.4.1 Relation-nLIRP-log (M) .....	13
5.4.1.1 Definition .....	13
5.4.1.2 Role.....	13
5.4.1.3 Constraint.....	13
5.4.2 Relation-log-logRecord (M) .....	13
5.4.2.1 Definition .....	13
5.4.2.2 Role .....	13
5.4.2.3 Constraint.....	13
5.4.3 Relation-logRecord-notificationIRPNotification (M).....	13
5.4.3.1 Definition .....	13
5.4.3.2 Role.....	14
5.4.3.3 Constraint.....	14
5.5 Information attribute definition .....	15
5.5.1 Definition and legal values .....	15
5.5.2 Constraints .....	16
6 Interface definition .....	17
6.1 Class diagram .....	17
6.2 Generic rules .....	17
6.3 NLIRPOperations_1 Interface (M).....	18
6.3.1 Operation subscribeLog (M).....	18
6.3.1.1 Definition .....	18
6.3.1.2 Input parameters.....	18

6.3.1.3	Output parameters .....	18
6.3.1.4	Pre-condition .....	18
6.3.1.5	Post-condition .....	19
6.3.1.6	Exceptions .....	19
6.3.2	Operation unsubscribeLog (M) .....	19
6.3.2.1	Definition .....	19
6.3.2.2	Input parameters .....	19
6.3.2.3	Output parameters .....	19
6.3.2.4	Pre-condition .....	20
6.3.2.5	Post-condition .....	20
6.3.2.6	Exceptions .....	20
6.3.3	Operation exportLogRecords (O) .....	20
6.3.3.1	Definition .....	20
6.3.3.2	Input parameters .....	20
6.3.3.3	Output parameters .....	20
6.3.3.4	Pre-condition .....	21
6.3.3.5	Post-condition .....	21
6.3.3.6	Exceptions .....	21
6.3.4	Operation getLogRecords (O) .....	21
6.3.4.1	Definition .....	21
6.3.4.2	Input parameters .....	21
6.3.4.3	Output parameters .....	21
6.3.4.4	Pre-condition .....	22
6.3.4.5	Post-condition .....	22
6.3.4.6	Exceptions .....	22
6.4	NLIRPOperations_2 Interface (O) .....	22
6.4.1	Operation getLogSubscriptionIds (M) .....	22
6.4.1.1	Definition .....	22
6.4.1.2	Input parameters .....	22
6.4.1.3	Output parameters .....	22
6.4.1.4	Pre-condition .....	23
6.4.1.5	Post-condition .....	23
6.4.1.6	Exceptions .....	23
6.4.2	Operation getLogSubscriptionStatus (M) .....	23
6.4.2.1	Definition .....	23
6.4.2.2	Input parameters .....	23
6.4.2.3	Output parameters .....	23
6.4.2.4	Pre-condition .....	23
6.4.2.5	Post-condition .....	23
6.4.2.6	Exceptions .....	23
6.5	NLIRPNotifications_1 Interface (M) .....	24
6.5.1	Notification notifyLogSubscribed (M) .....	24
6.5.1.1	Definition .....	24
6.5.1.2	Input Parameters .....	24
6.5.1.3	Triggering Event .....	24
6.5.1.3.1	From-state .....	24
6.5.1.3.2	To-state .....	24
6.5.2	Notification notifyLogUnsubscribed (M) .....	24
6.5.2.1	Definition .....	24
6.5.2.2	Input Parameters .....	25
6.5.2.3	Triggering Event .....	25
6.5.2.3.1	From-state .....	25
6.5.2.3.2	To-state .....	25
6.6	NLIRPNotifications_2 Interface (O) .....	25
6.6.1	Notification notifyOccupancyLevelCrossed (M) .....	25
6.6.1.1	Definition .....	25
6.6.1.2	Input Parameters .....	26
6.6.1.3	Triggering Event .....	26
6.6.1.3.1	From-state .....	26
6.6.1.3.2	To-state .....	26
6.6.2	Notification notifyLoggingResumed (O) .....	26
6.6.2.1	Definition .....	26

6.6.2.2 Input Parameters .....27  
6.6.2.3 Triggering Event .....27  
6.6.2.3.1 From-state.....27  
6.6.2.3.2 To-state.....27  
**Annex A (informative): Change history .....28**  
History .....29

---

## Foreword

This Technical Specification (TS) has been produced by the 3<sup>rd</sup> 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 3<sup>rd</sup> Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; as identified below:

TS 32.331	"Notification log Integration Reference Point (IRP): Requirements";
<b>TS 32.332</b>	<b>"Notification log Integration Reference Point (IRP): Information Service (IS)";</b>
TS 32.333	" Notification log Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)".
TS 32.334	"Notification log Integration Reference Point (IRP): Common Management Information Protocol (CMIP) Solution Set (SS)".
TS 32.335	"Notification log Integration Reference Point (IRP): eXtensible Markup Language (XML) definitions".

The present document is part of a TS-family, which describes the requirements and information model necessary for Telecommunications Management <sup>TM</sup>. 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 (NL IRP) 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; 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 Integration Reference Point (IRP): Requirements".
- [8] 3GPP TS 32.342 "Telecommunication management; File Transfer (FT) Integration Reference Point (IRP): Information Service (IS)".

---

# 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
NM	Network Manager
O	Optional
TS	Technical Specification
UML	Unified Modelling Language

---

## 4 System overview

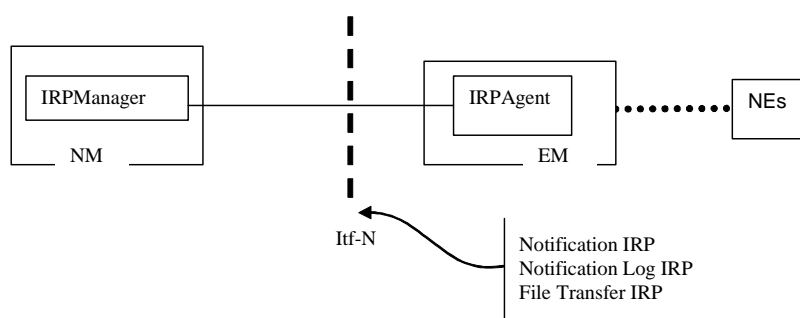
### 4.1 System context

Figures 4.1 and 4.2 identify system contexts of the IRP defined by the present specification in terms of its implementation called IRPAgent and the user of the IRPAgent, called IRPManager. For a definition of IRPManager and IRPAgent, see 3GPP TS 32.102 [2].

The IRPAgent implements and supports this IRP. The IRPAgent can reside in an Element Manager (EM - see figure 4.1) or a Network Element (NE - see figure 4.2). In the former case, the interfaces (represented by a thick dotted line) between the EM and the NEs are not the subject of this IRP.

An IRPManager using this IRP shall choose one of the two System Contexts defined here, for each NE. For instance, if an EM is responsible for managing a number of NEs, the NM shall access this IRP through the EM and not directly to those NEs. For another IRP though, the System Context may be different.

By observing the interaction across this IRP, one cannot deduce if EM and NE are integrated in a single system or if they run in separate systems.



**Figure 4.1: System Context A**

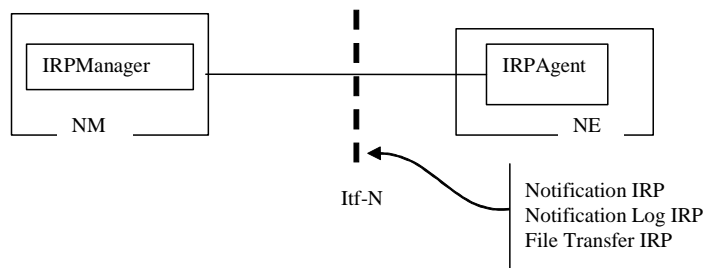


Figure 4.2: System Context B

## 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

## 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 subclause introduces the set of Information Object Classes (IOCs) that encapsulate information within the IRPAgent. The intent is to identify the information required for NL IRP operations and notification. This subclause provides the overview of all support object classes in UML. Subsequent subclauses provide more detailed specification of various aspects of these support object classes.

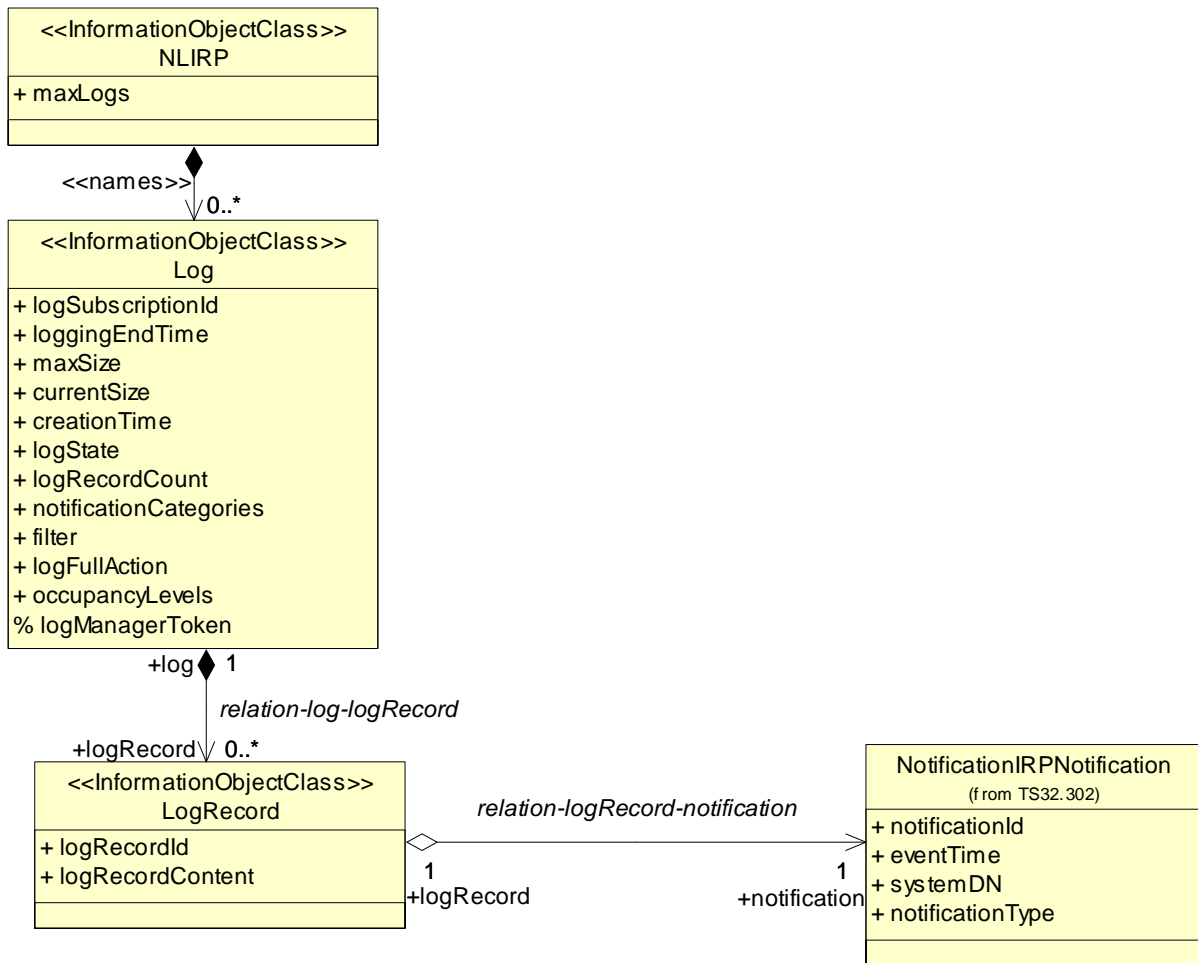


Figure 5.1: Information Object Class UML Diagram

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

## 5.2.2 Inheritance

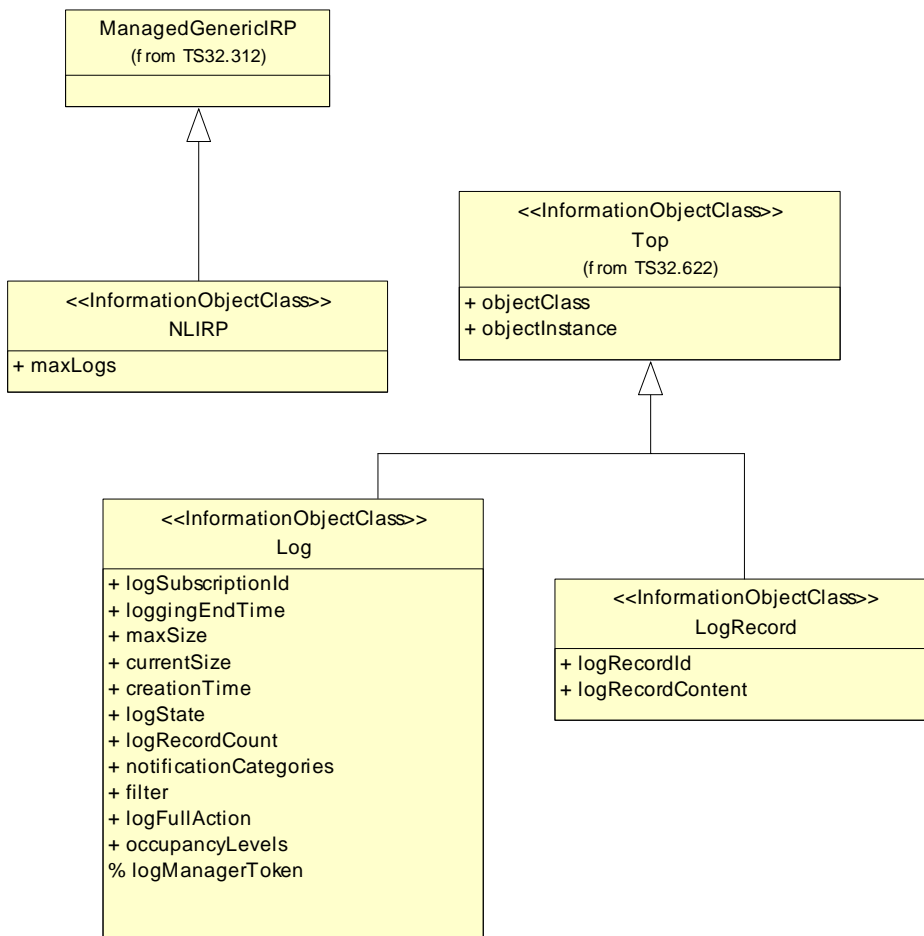


Figure 5.2: Information Object Class Inheritance UML Diagram

## 5.3 Information Object Class 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 3GPP TS 32.312 [6].

#### 5.3.1.2 Attributes

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
maxLogs	+	0	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	Visibility	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

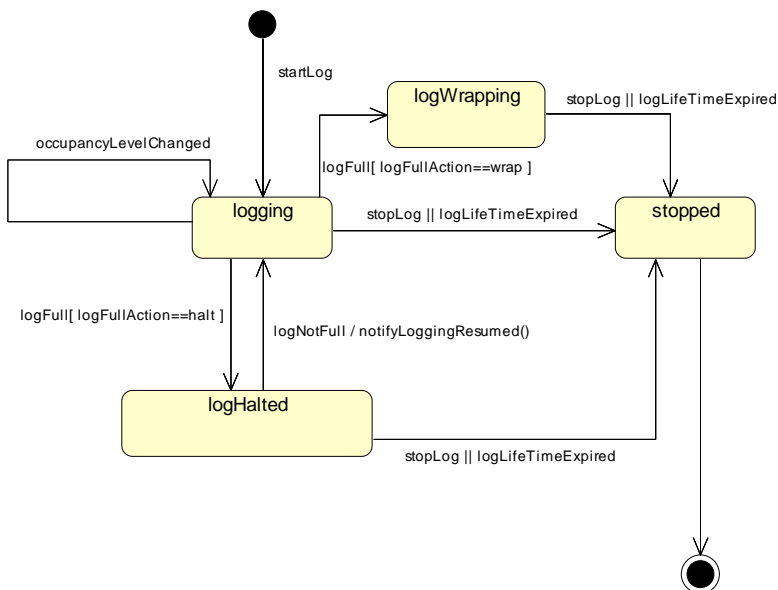


Figure 5.3: State Diagram for Notification Log

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.

### 5.3.3.2 Attributes

Attribute name	visibility	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 NL IRP instance.
uniqueLogManagerToken	The log manager token must be unique amongst all managers and logs utilizing logging services from a given NL IRP 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 <code>LogRecord</code> .
notification	It represents the <code>NotificationIRPNotification</code> .

## 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: <ul style="list-style-type: none"> <li>• zero</li> <li>• a positive whole number,</li> </ul>
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: <ul style="list-style-type: none"> <li>• 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).</li> <li>• halt: No more LogRecord(s) will be logged and all incoming events are discarded. LogRecord/s already in the Log will be retained.</li> </ul>
loggingEndTime	Defines the date and time when the log stops logging.	If log is created by IRPManager: <ul style="list-style-type: none"> <li>✚ value provided by the IRP Manager - the IRP Agent may reject the value provided by the IRP Manager in case this value is excessive</li> <li>✚ in case no value is provided by the IRP Manager then this value is set by the IRP Agent.</li> </ul> If log is created by IRPAgent: <ul style="list-style-type: none"> <li>✚ value indicates date and time or</li> <li>✚ value is expressed as 'indefinite'.</li> </ul>
logManagerToken	This attribute contains the token of a manager utilizing logging services from a given NL IRP instance, assigned by the IRPAgent.	The value of this attribute must be unique amongst all managers and logs utilizing logging services from the given NL IRP 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: <ul style="list-style-type: none"> <li>• logging</li> <li>• logFull</li> <li>• stopped</li> </ul>
logSubscriptionId	This attribute contains the id of a log subscription within a given NL IRP instance, assigned by the IRPAgent.	The value of this attribute must be unique amongst all log subscriptions managed by a given NL IRP 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: <ul style="list-style-type: none"> <li>• Non-zero, positive whole number</li> <li>• Zero indicates no limit on the</li> </ul>



Attribute Name	Definition	Legal Values
		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 subclause 3.1)
occupancyLevels	<p>Case Log.logFullAction == "halt" and Log.maxSize is non-zero:</p> <ul style="list-style-type: none"> <li>✚ this attribute contains a list of 3 values fixed (determined by the IRPAgent and never changed) of percentage, of Log.maxLog</li> <li>✚ the NLIRP will generate notifyOccupancyLevel to alert IRPManagers about the Log capacity currently used.</li> </ul> <p>Case Log.logFullAction == "wrap":</p> <ul style="list-style-type: none"> <li>✚ this attribute contains 1 value fixed (determined by the IRPAgent and never changed) of percentage of Log.maxLog</li> <li>✚ the NLIRP will generate notifyOccupancyLevel to alert IRPManagers that the specified percentage of the capacity has been written to the wrapping Log.</li> </ul>	<p>Case of "halt": a set of 3 values.</p> <p>Case of "wrap": 1 value.</p>

## 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

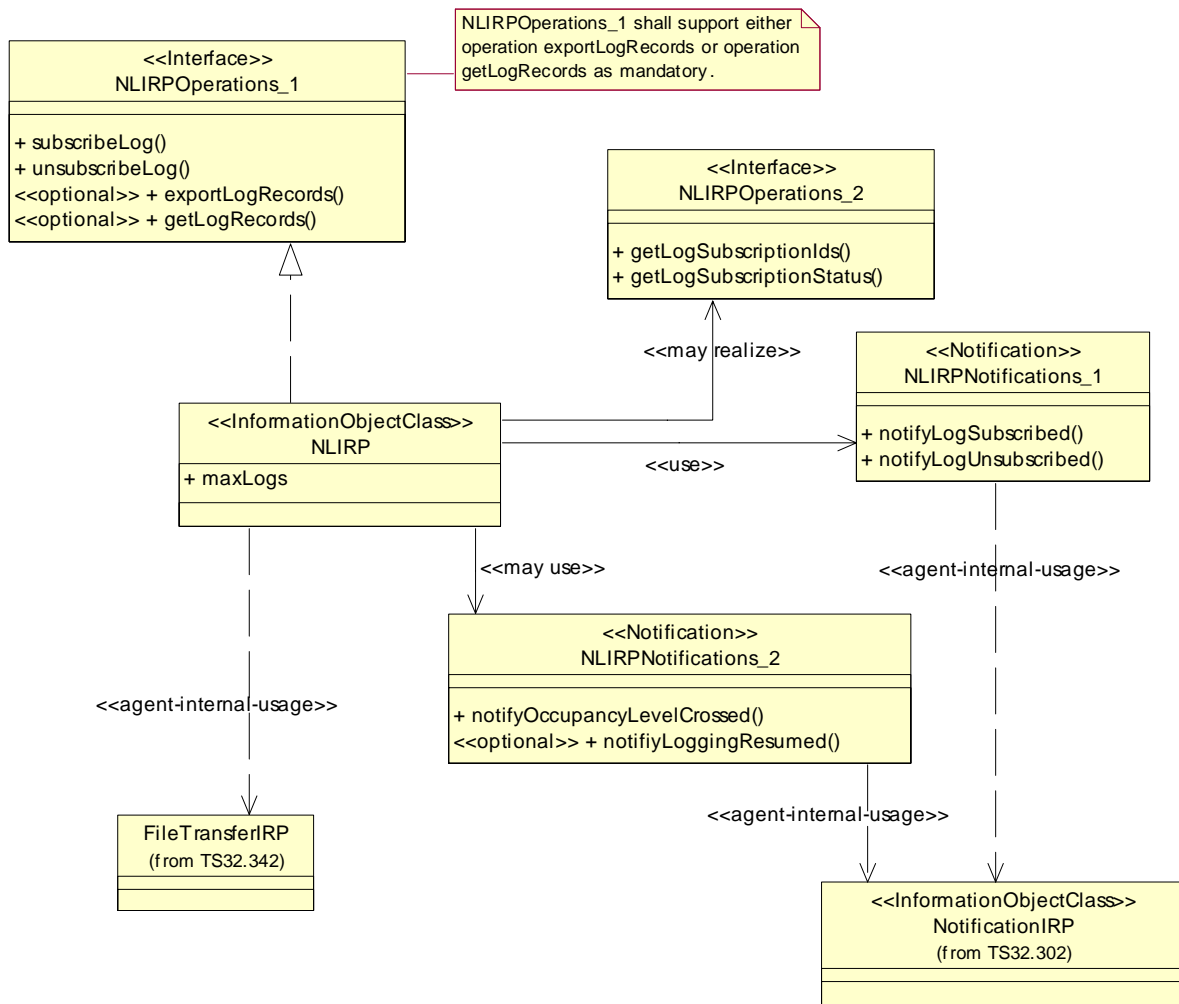


Figure 6.1 Class diagram

Note: 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 subclause 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 subclause 5.5.1
notificationCategories	O	Log.notificationCategories	See subclause 5.5.1; if notificationCategories is absent than all notifications are logged
filter	O	Log.ntfFilter  Filter constraint grammar is SS dependent	See subclause 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 subclause 5.5.1 This parameter contains either <ul style="list-style-type: none"> <li> the logSubscriptionId of the log created resulting from this operation or</li> <li> the value of the input parameter logSubscriptionId.</li> </ul>
logManagerToken	O	Log.logManagerToken	See subclause 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 subclause 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 notificationCategories is 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	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> Entry state.
maxLogs_reached	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> Entry state.
operation_failed	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> 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 subclause 5.5.1
logManagerToken	O	Log.logManagerToken	See subclause 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 subscribeLog did 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 subclause 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	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 6.3.3 Operation exportLogRecords (O)

Support for this operations 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 than is being transferred to the IRPManager using the File Transfer IRP (note also that the FT IRP is providing the file location to the IRP Manger).

### 6.3.3.2 Input parameters

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

### 6.3.3.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM (OperationSucceeded, NoMatchingLogRecordFound, OperationFailed)	If input parameters are valid and operation was successful, status = OperationSucceeded. If input parameters are valid and operation was successful but no matching log records are found, status = NoMatchingLogRecordFound. If operation_failed is true, status = OperationFailed.

### 6.3.3.4 Pre-condition

There are no pre-conditions, other than those established by the generic rules (see subclause 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
exportFailed	The IRPAgent was unable to export the specified records.
operation_failed	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 6.3.4 Operation getLogRecords (O)

Support for this 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 subclause 5.5.1
notificationCategories	O	Log.notificationCategories	See subclause 5.5.1
filter	O	Log.filter	See subclause 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 subclause 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	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> 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 NL IRP 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 subclause 5.5.1 & 6.4.1.2 Note: 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 subclause 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	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> Entry state.

### 6.4.2 Operation getLogSubscriptionStatus (M)

#### 6.4.2.1 Definition

Using this operation, an IRPManager can query the NL IRP 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 subclause 5.5.1

#### 6.4.2.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
logAttributeList	M	Attribute list of related Log IOC	List of all supported attributes of IOC Log (see subclause 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 subclause 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 IRP Agent.

#### 6.4.2.6 Exceptions

Exception Name	Definition
operation_failed	<b>Condition:</b> Pre-condition is true AND post-condition is false. <b>Returned Information:</b> The output parameter status. <b>Exit state:</b> 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	--	This parameter and objectInstance together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
objectInstance	M	--	This parameter and objectClass together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
notificationId	M	--	Notification header - see 3GPP TS 32.302 [3]
eventTime	M	--	Notification header - see 3GPP TS 32.302 [3]
notificationType	M	"notifyLogSubscribed"	Notification header - see 3GPP TS 32.302 [3]
systemDN	C	IRPAgent.systemDN.	It carries the DN of the IRPAgent that emits this notification. Notification header - see 3GPP TS 32.302 [3]
logSubscriptionId	M,N	Log.logSubscriptionId	See subclause 5.5.1
loggingEndTime	O,N	Log.loggingEndTime	See subclause 5.5.1
notificationCategories	O,N	Log.notificationCategories	See subclause 5.5.1; if absent than all notifications are being logged
filter	O,N	Log.filter	See subclause 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 NL IRP 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	--	This parameter and objectInstance together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
objectInstance	M	--	This parameter and objectClass together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
notificationId	M	--	Notification header - see 3GPP TS 32.302 [3]
eventTime	M	--	Notification header - see 3GPP TS 32.302 [3]
notificationType	M	"notifyLogUnsubscribed"	Notification header - see 3GPP TS 32.302 [3]
systemDN	C	IRPAgent.systemDN.	It carries the DN of the IRPAgent that emits this notification. Notification header - see 3GPP TS 32.302 [3]
logSubscriptionId	M,N	Log.logSubscriptionId	See subclause 5.5.1

## 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 5.1.1 (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	--	This parameter and objectInstance together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
objectInstance	M	--	This parameter and objectClass together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
notificationId	M	--	Notification header - see 3GPP TS 32.302 [3]
eventTime	M	--	Notification header - see 3GPP TS 32.302 [3]
notificationType	M	"notifyOccupancyLevelCrossed"	Notification header - see 3GPP TS 32.302 [3]
systemDN	C	IRPAgent.systemDN.	It carries the DN of the IRPAgent that emits this notification. Notification header - see 3GPP TS 32.302 [3]
logSubscriptionId	M,Y	Log.logSubscriptionId	See subclause 5.1.1
currentOccupancyLevel	M,N	Log.currentSize/ Log.currentSize*100	See subclause 5.1.1
logFullAction	O,N	Log.logFullAction	See subclause 5.1.1

## 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 5.1.1 log.occupancyLevel).

### 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	--	This parameter and objectInstance together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
objectInstance	M	--	This parameter and objectClass together carry the semantics of the DN of Log. Notification header - see 3GPP TS 32.302 [3]
notificationId	M	--	Notification header - see 3GPP TS 32.302 [3]
eventTime	M	--	Notification header - see 3GPP TS 32.302 [3]
notificationType	M	"notifyLoggingResumed"	Notification header - see 3GPP TS 32.302 [3]
systemDN	C	IRPAgent.systemDN.	It carries the DN of the IRPAgent that emits this notification. Notification header - see 3GPP TS 32.302 [3]
logSubscriptionId	M,Y	Log.logSubscriptionId	See subclause 5.1.1

### 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 5.1.1 log.occupancyLevel).

#### 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	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

---

# History

<b>Document history</b>		
V6.0.0	December 2004	Publication