TSGS#27(05)0036

Source: **SA5 (Telecom Management)**

Title: **Rel-6 TS 32333-100 Notification Log IRP CORBA SS - for SA Approval**

Document for: **Approval**

Agenda Item: **7.5.3**

3GPP TSG-SA5 (Telecom Management) **S5-056025**
Meeting #41, Lisbon, PORTUGAL, 24 - 28 January 2005

Presentation of Technical Specification to TSG SA

Presentation to: **TSG SA Meeting #27**
Document for presentation: **TS 32.333, Version 1.0.0**
Notification Log IRP: CORBA Solution Set
Presented for: **Approval**

Abstract of document:

This TS implements in CORBA the requirements and information service defined in 32.331 and 32.332 for the Notification Log IRP.

Work done against the WID contained in SP-040789 (Work Item ID: OAM-NIM).

This is part of the Notification Log IRP, which consists of:

Number	Title
32.331	Notification Log Integration Reference Point (IRP): Requirements
32.332	Notification Log Integration Reference Point (IRP): Information Service
32.333	**Notification Log Integration Reference Point (IRP): CORBA Solution Set**
32.334	Notification Log Integration Reference Point (IRP): CMIP Solution Set
32.335	Notification Log Integration Reference Point (IRP): XML definitions

The purpose of this set of specifications is to provide a Notification Log mechanism enabling the network manager to log and retrieve logged notifications in the managed systems.

Changes since last presentation to TSG SA :
New

Outstanding Issues:
None

Contentious Issues:
None

3GPP TS 32.333 V1.0.0 (2005-03)

Technical Specification

3rd Generation Partnership Project;
Technical Specification Group Services and System Aspects
Telecommunication management;
Notification Log (NL) Integration Reference Point (IRP):
Common Object Request Broker Architecture (CORBA)
Solution Set (SS)
(Release 6)

Keywords
telecom management

3GPP

Postal address

3GPP support office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet
http://www.3gpp.org

Contents

Foreword

This Technical Specification 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 a member 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 32.333 "Notification Log (NL) Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)";

TS 32.334 "Notification Log (NL) Integration Reference Point (IRP): Common Management Information Protocol (CMIP) Solution Set (SS)";

TS 32.335 "Notification Log (NL) Integration Reference Point (IRP): eXtensible Markup Language (XML) definitions";

1 Scope

The present document specifies the CORBA Solution Set for the IRP whose semantics are specified in 3GPP TS 32.332 [6] Notification Log IRP: Information Service.

Clause 1 to 3 provides background information. Clause 4 provides key architectural features supporting the SS. Clause 5 defines the mapping of operations, notification, parameters and attributes defined in IS to their SS equivalents. Clause 6 describes the notification interface containing the push method. Annex A contains the IDL specification.

This Solution Set specification is related to TS 32.332 V6.0.X.

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] OMG TC Document telecom/98-11-01: "OMG Notification Service". http://www.omg.org/technology/documents/

[2] OMG CORBA Services: "Common Object Services Specification, Update: November 22, 1996" (Clause 4 contains the Event Service specification). http://www.omg.org/technology/documents/

[3] 3GPP TS 32.311: "Telecommunication management; Generic Integration Reference Point (IRP) Management; Requirements".

[4] 3GPP TS 32.302: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Information Service (IS)".

[5] 3GPP TS 32.303: " Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)".

[6] 3GPP TS 32.332: "Telecommunication management; Notification Log Integration Reference Point: Information Service (IS)".

[7] 3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) Management; Information Service (IS)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 32.332 [6] apply.

IRP document version number string (or "IRPVersion"): See 3GPP TS 32.311 [3] subclause 3.1.

3.2 Abbreviations

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

CORBA	Common Object Request Broker Architecture
IDL	Interface Definition Language
IRP	Integration Reference Point
MOC	Managed Object Class
MOI	Managed Object Instance
NE	Network Element
OMG	Object Management Group

4 Architectural Features

The overall architectural feature of Notification Log IRP is specified in 3GPP TS 32.332 [6]. This clause specifies features that are specific to the CORBA SS.

4.1 Notification Services

In implementations of CORBA SS, IRPAgent conveys Notification Log notifications to IRPManager via OMG Notification Service (OMG Notification Service [1]).

OMG Event Service [2] provides event routing and distribution capabilities. OMG Notification Service provides, in addition to Event Service, event filtering and Quality Of Service (QOS).

A necessary and sufficient subset of OMG Notification Services shall be used to support Notification Log notifications as specified in 3GPP TS 32.332 [6].

These operation are classified as <<AgentInternal-usage>> in GPP TS 32.332 [6].

5. Mapping

5.1 Operation and Notification mapping

3GPP TS 32.332 [6] defines semantics of operations and notifications visible across the Notification Log IRP.
The following table indicates the mapping of these operations and notifications to their equivalents defined in this SS.

Table 5.1.1: Mapping from IS Notification/Operation to SS equivalents

IS Operation/ notification 3GPP TS 32.332 [6]	SS Method	Qualifier
subscribeLog	subscribe_log	M
unsubscribeLog	unsubscribe_log	M
exportLogRecords	export_log_records	O
getLogRecords	get_log_records	O
getLogSubscriptionIds	get_log_subscription_ids	O
getLogSubscriptionStatus	get_log_subscription_status	O
notifyLogSubscribed	push_structured_event (note 1). See clause 4.1. See interface NotifyLogSubscribed	M
notifyLogUnsubscribed	push_structured_event (note 1). See clause 4.1. See interface NotifyLogUnubscribed	M
notifyOccupancyLevelCrossed	push_structured_event (note 1). See clause 4.1. See interface NotifyLogOccupancyLevelCrossed	O
notifyLoggingResumed	push_structured_event (note 1). See clause 4.1. See interface NotifyLoggingResumed	O

NOTE: OMG Notification Service OMG Notification Service [1] defines this method.

5.2 Operation parameter mapping

3GPP TS 32.332 [6] defines semantics of parameters carried in operations across the Notification Log IRP.
The following tables indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table 5.2.1: Mapping from IS `subscribeLog` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
logSubscriptionId	NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId	M
loggingEndTIme	GenericIRPManagementConstDefs::IRPTime loggingEndTimeAsked	O
notificationCategories	GenericIRPManagementConstDefs::NotificationCategorySet notificationCategorySet	O
filter	GenericIRPManagementConstDefs::StringTypeOpt filter	O
logManagerToken	NotificationLogIRPConstDefs::LogManagerTokenOpt logManagerToken;	O
loggingEndTime	GenericIRPManagementConstDefs::loggingEndTimeGiven	O
status	GenericIRPManagementConstDefs::Signal Exceptions: SubscribeLog, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::ValueNotSupported, InvalidLogSubscriptionId, UnknownLogManagerToken.	M

Table 5.2.2: Mapping from IS `unsubscribeLog` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
logSubscriptionId	NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId	M
logManagerToken	NotificationLogIRPConstDefs::LogManagerTokenOpt logManagerToken	O
status	GenericIRPManagementConstDefs::Signal Exceptions: UnsubscribeLog, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ParameterNotSupported, UnknownLogSubscriptionId, UnknownLogManagerToken).	M

Table 5.2.3: Mapping from IS `exportLogRecords` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
logSubscriptionId	NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId	M
notificationCategories	NotificationLogIRPConstDefs::NotificationCategorySetOpt notificationCategorySet	O
filter	GenericIRPManagementConstDefs::StringTypeOpt filter	O
fileLocation (editor note 1)	string fileLocation	M
status	GenericIRPManagementConstDefs::Signal Exceptions: ExportLogRecords, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::OperationNotSupported, UnknownLogSubscriptionId.	M

Editor Note 1: The use of filename is not yet aligned with current IS and require further discussion.

Table 5.2.4: Mapping from IS `getLogRecords` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
logSubscriptionId	NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId	M
notificationCategories	NotificationLogIRPConstDefs::NotificationCategorySetOpt notificationCategories,	O
filter	GenericIRPManagementConstDefs::StringTypeOpt filter	O
getLogRecordsResult	Some or all of the information contained in getLogRecordsResult will be returned via the return value of type DsLogAdmin::RecordList.	M
status	Exceptions: GetLogRecords, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported, UnknownLogSubscriptionId.	M

Table 5.2.5: Mapping from IS `getLogSubscriptionIds` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
logSubscriptionIds	return value of type DsLogAdmin::LogIdList	M
status	Exceptions: GetLogSubscriptionIds, GenericIRPManagementSystem::OperationNotSupported.	M

Table 5.2.6: Mapping from IS `getLogSubscriptionStatus` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
logSubscriptionId	NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId	M
logAttributeList	NotificationLogIRPConstDefs::LogAttributeList logAttributeList	M
status	GenericIRPManagementConstDefs::Signal Exceptions: GetLogSubscriptionStatus, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::OperationNotSupported.	M

5.3 Notification parameter mapping

3GPP TS 32.332 [6] defines semantics of parameters carried in notifications. The following tables indicate the mapping of these parameters to their OMG CORBA Structured Event (defined in OMG Notification Service [1]) equivalents. The composition of OMG Structured Event, as defined in the OMG Notification Service [1], is:

```
Header
      Fixed Header
           domain_name
           type_name
           event_name
      Variable Header
Body
      filterable_body_fields
      remaining_body
```

The following tables list all OMG Structured Event attributes in the second column. The first column identifies the 3GPP TS 32.332 [6] defined notification parameters.

Table 5.3.1: Mapping for `notifyLogSubscribed`

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding SS attribute.	domain_name	M	It carries the IRP document version number string. See sub-clause 3.3. It indicates the syntax and semantics of the Structured Event as defined by this specification.
notificationType	type_name	M	This is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: NOTIFY_LOG_SUBSRIBED.
There is no corresponding IS attribute.	event_name	M	Null-string
There is no corresponding SS attribute.	variable Header		
objectClass, objectInstance	One NV (note 1) pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: MANAGED_OBJECT_INSTANCE. Value of NV pair is a string.
notificationId	One NV pair of remaining_body	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: NOTIFICATION_ID. Value of NV pair is a long.
eventTime	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: EVENT_TIME. Value of NV pair is of type GenericIRPManagementConstDefs::IRPTime.
systemDN	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: SYSTEM_DN Value of NV pair is a string.
logSubscriptionId	One NV pair of remaining_body	M	Name of NV pair is the NotificationLogIRPNotifications:: NotifyLogSubscribed::LOG_SUBSCRIPTION_ID. Value of NV pair is of type NotificationLogIRPConstDefs::LogSubscriptionId.
loggingEndTime	One NV pair of remaining_body	O	Name of NV pair is the NotificationLogIRPNotifications::NotifyLogSubscribed::LOGGING_END_TIME. Value of NV pair is a string.
notificationCategories	One NV pair of remaining_body	O	Name of NV pair is the NotificationLogIRPNotifications:: NotifyLogSubscribed::NOTIFICATION_CATEGORIES. Value of NV pair is of type GenericIRPManagementConstDefs::VersionNumberSet.
filter	One NV pair of remaining_body	O	Name of NV pair is notificationLogIRPNotifications:: NotifyLogSubscribed:: FILTER Value of NV pair is a string.

NOTE: NV stands for name-value pair. Order arrangement of NV pairs is not significant. The name of NV-pair is always encoded in string. This note is applicable to all NV-pairs and all mapping tables for all notifications.

Table 5.3.2: Mapping for `notifyLogUnsubscribed`

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding IS attribute.	domain_name	M	It carries the IRP document version number string. See sub-clause 3.3. It indicates the syntax and semantics of the Structured Event as defined by this specification.
notificationType	type_name	M	This is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: NOTIFY_LOG_UNSUBSCRIBED.
There is no corresponding IS attribute.	event_name	M	Null-string.
	variable Header		
objectClass, objectInstance	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: MANAGED_OBJECT_INSTANCE. Value of NV pair is a string.
notificationId	One NV pair of remaining_body	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: NOTIFICATION_ID. Value of NV pair is a long.
eventTime	One NV pair of One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: EVENT_TIME. Value of NV pair is a IRPTime of module GenericIRPManagementConstDefs.
systemDN	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: SYSTEM_DN Value of NV pair is a string.
logSubscriptionId	One NV pair of filterable_body_fields (editor note 2)	M	Name of NV pair is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: LOG_SUBSCRIPTION_ID. Value of NV pair is of type NotificationLogIRPConstDefs::LogSubscriptionId.

Editor note 2: The placement of this parameter in filterable_body_fields so that it is filterable is not yet aligned with current IS and require further discussion.

Table 5.3.3: Mapping for `notifyOccupancyLevelCrossed`

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding SS attribute.	domain_name	M	It carries the IRP document version number string. See sub-clause 3.3. It indicates the syntax and semantics of the Structured Event as defined by this specification.
notificationType	type_name	M	This is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: NOTIFY_LOG_OCCUPANCY_LEVEL_CROSSED.
There is no corresponding SS attribute	event_name	M	Null-string
There is no corresponding SS attribute.	variable Header		
objectClass, objectInstance	One NV pair of filterable_body_fields	M	NV stands for name-value pair. Order arrangement of NV pairs is not significant. The name of NV-pair is always encoded in string. Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: MANAGED_OBJECT_INSTANCE. Value of NV pair is a string.
notificationId	One NV pair of remaining_body	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: NOTIFICATION_ID. Value of NV pair is a long.
eventTime	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: EVENT_TIME. Value of NV pair is a IRPTime of module GenericIRPManagementConstDefs.
systemDN	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: SYSTEM_DN Value of NV pair is a string.
logSubscriptionId	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: LOG_SUBSCRIPTION_ID. Value of NV pair is of type NotificationLogIRPConstDefs::LogSubscriptionId.
currentOccupancyLevel	One NV pair of One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: CURRENT_OCCUPANCY_LEVEL. Value of NV pair is an unsigned short (i.e. DsLogAdmin:: Threshold).
logFullAction	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: LOG_FULL_ACTION. Value of NV pair is of type NotificationLogIRPConstDefs:: LogFullActionType.

Table 5.3.4: Mapping for `notifyLoggingResumed`

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding IS attribute.	domain_name	M	It carries the IRP document version number string. See sub-clause 3.3. It indicates the syntax and semantics of the Structured Event as defined by this specification.
notificationType	type_name	M	This is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: NOTIFY_LOGGING_RESUMED.
There is no corresponding IS attribute.	event_name	M	Null-string.
There is no corresponding IS attribute.	variable Header		
objectClass, objectInstance	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: MANAGED_OBJECT_INSTANCE. Value of NV pair is a string.
notificationId	One NV pair of remaining_body	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: NOTIFICATION_ID. Value of NV pair is a long.
eventTime	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: EVENT_TIME. Value of NV pair is a IRPTime of module GenericIRPManagementConstDefs.
systemDN	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationIRPConstDefs:: AttributeNameValue:: SYSTEM_DN Value of NV pair is a string.
logSubscriptionId	One NV pair of filterable_body_fields	M	Name of NV pair is the NotificationLogIRPNotifications:: NotifyLogSubscribed:: LOG_SUBSCRIPTION_ID. Value of NV pair is of type NotificationLogIRPConstDefs::LogSubscriptionId.

6 NotificationLogIRPNotifications Interface

OMG CORBA Notification push operation is used to realise the notification of NotificationLogIRPNotifications. All the notifications in this interface are implemented using this push_structured_event method.

6.1 Method push (M)

```
module CosNotifyComm {
…
Interface SequencePushConsumer : NotifyPublish {
             void push_structured_events(
                    in CosNotification::EventBatch notifications)
          raises( CosEventComm::Disconnected);
             …
      }; // SequencePushConsumer
      …
}; // CosNotifyComm
```

NOTE 1: The push_structured_events method takes an input parameter of type EventBatch as defined in the OMG CosNotification module (OMG Notification Service [1]). This data type is the same as a sequence of Structured Events. Upon invocation, this parameter will contain a sequence of Structured Events being delivered to IRPManager by IRPAgent to which it is connected.

NOTE 2: The maximum number of events that will be transmitted within a single invocation of this operation is controlled by IRPAgent wide configuration parameter.

NOTE 3: The amount of time the supplier (IRPAgent) of a sequence of Structured Events will accumulate individual events into the sequence before invoking this operation is controlled by IRPAgent wide configuration parameter as well.

NOTE 4: IRPAgent may push EventBatch with only one Structured Event.

Annex A (normative): IDL specifications

A.1 IDL specification (file name "NotificationLogIRPConstDefs.idl")

```
// File: NotificationLogIRPConstDefs.idl

#ifndef _NOTIFICATIONLOGIRPCONSTDEFS_IDL_
#define _NOTIFICATIONLOGIRPCONSTDEFS_IDL_

#include <DsLogAdmin.idl>
#include "NotificationIRPConstDefs.idl"
#include "GenericIRPManagementConstDefs.idl"

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

module NotificationLogIRPConstDefs
{

    typedef DsLogAdmin::LogId LogSubscriptionId;
    typedef DsLogAdmin::RecordId RecordId;
    typedef DsLogAdmin::RecordIdList RecordIdList;

    typedef string LogManagerToken;
    /*
    LogManagerTokenOpt is a type carrying an optional parameter.
    If the boolean is TRUE, then the value is present; else absent.
    */
    union LogManagerTokenOpt switch (boolean)
    {
       case TRUE:  LogManagerToken value;
    };


    /*
    IRPTimeOpt is a type carrying an optional parameter.
    If the boolean is TRUE, then the value is present; else absent.
    */
    union IRPTimeOpt switch (boolean)
    {
       case TRUE: GenericIRPManagementConstDefs::IRPTime value;
    };


    typedef GenericIRPManagementConstDefs::VersionNumberSet
        NotificationCategorySet;
    /*
    NotificationCategorySetOpt is a type carrying an optional parameter.
    If the boolean is TRUE, then the value is present; else absent.
    */
    union NotificationCategorySetOpt switch (boolean)
    {
       case TRUE:  NotificationCategorySet value;
    };

    enum LogState { LOGGING, LOGFULL, STOPPED};
```

```
    typedef DsLogAdmin::CapacityAlarmThresholdList
        CapacityAlarmThresholdList;

    typedef unsigned short LogFullActionType;
    const LogFullActionType wrap = 0;
    const LogFullActionType halt = 1;


   /*
   IteratorOpt is a type carrying an optional parameter.
   If the boolean is TRUE, then the value is present; else absent.
   */
   union IteratorOpt switch (boolean)
   {
      case TRUE:  DsLogAdmin::Iterator value;
   };

   struct LogAttributes {
       LogSubscriptionId logSubscriptionId;
       GenericIRPManagementConstDefs::IRPTime loggingEndTime;
       unsigned long long maxSize;
       unsigned long long currentSize;
       LogState logState;
       unsigned long long logRecordCount;
       NotificationIRPConstDefs::NotificationCategorySet notificationCategories;
       string filter;
       LogFullActionType logFullAction;
       CapacityAlarmThresholdList occupancyLevels;
       };

   typedef sequence <LogAttributes> LogAttributeList;


};



#endif // _NOTIFICATIONLOGIRPCONSTDEFS_IDL_
```

A.2 IDL specification (file name “NotificationLogIRPSystem.idl”)

```
// File: NotificationLogIRPSystem.idl

#ifndef _NOTIFICATIONLOGIPRSYSTEM_IDL_
#define _NOTIFICATIONLOGIRPSYSTEM_IDL_

#include "NotificationLogIRPConstDefs.idl"
#include "GenericIRPManagementSystem.idl"
#include <DsLogAdmin.idl>
#include <TimeBase.idl>

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

module NotificationLogIRPSystem
{
    /*
    System fails to complete the method.  System can provide reason
    to qualify the exception.  The semantics carried in reason
    is outside the scope of this IRP.
    */
    exception SubscribeLog { string reason; };
    exception UnsubscribeLog { string reason; };
    exception ExportLogRecords { string reason; };
    exception GetLogSubscriptionIds { string reason; };
    exception GetLogRecords { string reason; };
    exception GetLogSubscriptionStatus { string reason; };

    exception InvalidLogSubscriptionId { string reason; };
    exception UnknownLogSubscriptionId {};
    exception UnknownLogManagerToken {};
    exception InvalidConstraint { string reason; };



    interface NotificationLogIRP : GenericIRPManagementSystem::
        GenericIRPManagement
    {
        GenericIRPManagementConstDefs::Signal subscribe_log (
            in NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId,
            in NotificationLogIRPConstDefs::IRPTimeOpt loggingEndTimeAsked,
            in NotificationLogIRPConstDefs::NotificationCategorySetOpt
                notificationCategorySet,
            in GenericIRPManagementConstDefs::StringTypeOpt filter,
            out NotificationLogIRPConstDefs::LogSubscriptionId
               logSubscriptionIdOut,
            out NotificationLogIRPConstDefs::LogManagerTokenOpt logManagerToken,
            out NotificationLogIRPConstDefs::IRPTimeOpt loggingEndTimeGiven
        )
        raises ( SubscribeLog,
             GenericIRPManagementSystem::InvalidParameter,
             GenericIRPManagementSystem::ParameterNotSupported,
             GenericIRPManagementSystem::ValueNotSupported,
             InvalidLogSubscriptionId, UnknownLogManagerToken
             );

        GenericIRPManagementConstDefs::Signal unsubscribe_log (
            in NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId,
            in NotificationLogIRPConstDefs::LogManagerTokenOpt logManagerToken
        )
```

```
        raises ( UnsubscribeLog,
              GenericIRPManagementSystem::InvalidParameter,
              GenericIRPManagementSystem::ParameterNotSupported,
              UnknownLogSubscriptionId, UnknownLogManagerToken);

    GenericIRPManagementConstDefs::Signal export_log_records (
        in NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId,
        in NotificationLogIRPConstDefs::NotificationCategorySetOpt
            notificationCategorySet,
        in GenericIRPManagementConstDefs::StringTypeOpt filter,
        out string fileName
        )
        raises ( ExportLogRecords,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::ParameterNotSupported,
            GenericIRPManagementSystem::OperationNotSupported,
            UnknownLogSubscriptionId);


     //  If some but not all of the information is returned via the return value
     //  RecordList, then the rest of the information is returned via the
     //  iterator.  Otherwise, the iterator is absent.
     //
    DsLogAdmin::RecordList get_log_records(
        in NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId,
        in NotificationLogIRPConstDefs::NotificationCategorySetOpt
            notificationCategories,
        in GenericIRPManagementConstDefs::StringTypeOpt filter,
        out NotificationLogIRPConstDefs::IteratorOpt iterator
        )
        raises ( GetLogRecords,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::ParameterNotSupported,
            GenericIRPManagementSystem::ValueNotSupported,
            GenericIRPManagementSystem::OperationNotSupported,
            UnknownLogSubscriptionId);


    DsLogAdmin::LogIdList get_log_subscription_ids()
        raises ( GetLogSubscriptionIds,
             GenericIRPManagementSystem::OperationNotSupported
             );


    GenericIRPManagementConstDefs::Signal get_log_subscription_status (
        in NotificationLogIRPConstDefs::LogSubscriptionId logSubscriptionId,
        out NotificationLogIRPConstDefs::LogAttributeList logAttributeList
        )
        raises ( GetLogSubscriptionStatus,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::OperationNotSupported
            );

    };
};

#endif // _NOTIFICATIONLOGIPRSYSTEM_IDL_
```

A.3 IDL specification (file name "NotificationLogIRPNotifications.idl")

```
// File: NotificationLogIRPNotifications.idl

#ifndef _NOTIFICATIONLOGIRPNOTIFICATIONS_IDL_
#define _NOTIFICATIONLOGIRPNOTIFICATIONS_IDL_

#include "NotificationIRPNotifications.idl"

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

module NotificationLogIRPNotifications
{

    interface NotifyLogSubscribed: NotificationIRPNotifications::Notify
    {
        const string NOTIFY_LOG_SUBSRIBED = "x1";
        const string LOG_SUBSCRIPTION_ID = "id";
        const string LOGGING_END_TIME = "loggingEndTime";
        const string NOTIFICATION_CATEGORIES = "categories";
        const string FILTER = "filter";

    };

    interface NotifyLogUnubscribed: NotificationIRPNotifications::Notify
    {
        const string NOTIFY_LOG_UNSUBSCRIBED = "x2";
        const string LOG_SUBSCRIPTION_ID = "id";
    };

    interface NotifyLogOccupancyLevelCrossed:
        NotificationIRPNotifications::Notify
    {
        const string NOTIFY_LOG_OCCUPANCY_LEVEL_CROSSED = "x3";
        const string LOG_SUBSCRIPTION_ID = "id";
        const string CURRENT_OCCUPANCY_LEVEL = "level";
        const string LOG_FULL_ACTION = "fullAction";
    };

    interface NotifyLoggingResumed: NotificationIRPNotifications::Notify
    {
        const string NOTIFY_LOGGING_RESUMED = "x4";
        const string LOG_SUBSCRIPTION_ID = "id";
    };


};

#endif // _NOTIFICATIONLOGIRPNOTIFICATIONS_IDL_
```

Annex B (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
Mar 2005	S_27	SP-050036	--	--	Submitted to TSG SA#27 for Approval	1.0.0	