

# 3GPP TS 32.106-2 V3.3.0 (2001-03)

---

*Technical Specification*

**3rd Generation Partnership Project;  
Technical Specification Group Services and System Aspects;  
Telecommunication Management;  
Configuration Management;  
Part 2: Notification Integration Reference Point:  
Information Service Version 1  
(Release 1999)**

---



The present document has been developed within the 3<sup>rd</sup> Generation Partnership Project (3GPP™) and may be further elaborated for the purposes of 3GPP.

The present document has not been subject to any approval process by the 3GPP Organisational Partners and shall not be implemented.

This Specification is provided for future development work within 3GPP only. The Organisational Partners accept no liability for any use of this Specification.

Specifications and reports for implementation of the 3GPP™ system should be obtained via the 3GPP Organisational Partners' Publications Offices.

---

Keywords

---

UMTS, 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>

---

**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.

© 2001, 3GPP Organizational Partners (ARIB, CWTS, ETSI, T1, TTA, TTC).  
All rights reserved.

# Contents

Foreword .....	4
Introduction .....	4
1 Scope .....	5
2 References .....	5
3 Definitions and abbreviations .....	6
3.1 Definitions .....	6
3.2 Abbreviations .....	6
4 System overview .....	7
4.1 System context for Notification .....	7
5 Modelling approach .....	8
6 IRP Information Service .....	8
6.1 Interfaces .....	8
6.1.1 NotificationIRPOperation Interface .....	9
6.1.1.1 Operation subscribe (M) .....	9
6.1.1.2 Operation unsubscribe (M) .....	10
6.1.1.3 Operation getNotificationIRPVersion (M) .....	10
6.1.1.4 Operation getSubscriptionStatus (O) .....	11
6.1.1.5 Operation getSubscriptionIds (O) .....	11
6.1.1.6 Operation changeSubscriptionFilter (O) .....	12
6.1.1.7 Operation suspendSubscription (O) .....	12
6.1.1.8 Operation resumeSubscription (O) .....	12
6.1.1.9 Operation getNotificationCategories (O) .....	12
6.1.2 NotificationIRPNotification Interface .....	13
6.1.2.1 Notification notify .....	13
6.1.2.2 Notification Attributes .....	13
6.1.2.2.1 managedObjectClass (M) .....	14
6.1.2.2.2 managedObjectInstance (M) .....	14
6.1.2.2.3 notificationId (O) .....	14
6.1.2.2.4 eventTime (M) .....	14
6.1.2.2.5 systemDN (C) .....	14
6.1.2.2.6 eventType (M) .....	14
6.1.2.2.7 extendedEventType (M) .....	15
6.1.3 Behaviour .....	15
6.1.3.1 IRP Agent supports multiple subscriptions with one IRPManager .....	15
6.1.3.2 Support of packing multiple notifications .....	16
6.1.3.3 IRP Agent supports emission of multiple Notification categories .....	16
6.1.3.4 Subscription list loss .....	16
6.1.3.5 Notification ordering .....	16
<b>Annex A (informative): Change history .....</b>	<b>17</b>

---

## Foreword

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

The present document is part 2 of a multi-part TS covering the 3<sup>rd</sup> Generation Partnership Project: Technical Specification Group Services and System Aspects; Telecommunication Management; Configuration Management, as identified below:

- Part 1: “3G Configuration Management: Concept and Requirements”;
- Part 2: “Notification Integration Reference Point: Information Service Version 1”;**
- Part 3: “Notification Integration Reference Point: CORBA Solution Set Version 1:1”;
- Part 4: “Notification Integration Reference Point: CMIP Solution Set Version 1:1”;
- Part 5: “Basic Configuration Management IRP Information Model (including NRM) Version 1”;
- Part 6: “Basic Configuration Management IRP CORBA Solution Set Version 1:1”;
- Part 7: “Basic Configuration Management IRP CMIP Solution Set Version 1:1”;
- Part 8: “Name Convention for Managed Objects”.

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

Configuration Management (CM), in general, provides the operator with the ability to assure correct and effective operation of the 3G network as it evolves. CM actions have the objective to control and monitor the actual configuration on the Network Elements (NEs) and Network Resources (NRs), and they may be initiated by the operator or by functions in the Operations Systems (OSs) or NEs.

CM actions may be requested as part of an implementation programme (e.g. additions and deletions), as part of an optimisation programme (e.g. modifications), and to maintain the overall Quality Of Service (QOS). The CM actions are initiated either as a single action on a NE of the 3G network or as part of a complex procedure involving actions on many NEs.

The Itf-N interface for Configuration Management (CM) is built up by a number of Integration Reference Points (IRPs) and a related Name Convention, which realise the functional capabilities over this interface. The basic structure of the IRPs is defined in 3GPP TS 32.101 [7] and 3GPP TS 32.102 [8]. For CM, a number of IRPs (and the Name Convention) are defined herein, used by this as well as other specifications for Telecom Management produced by 3GPP. All these are included in 3GPP TS 32.106 from Part 2 and onwards.

The present document is Part 2 of 3GPP TS 32.106 (3GPP TS 32.106-2) - Notification IRP Information Service.

---

# 1 Scope

Network Elements (NEs) under management generate events to inform event receivers about occurrences within the network that may be of interest to event receivers. There are a number of categories of events. Alarm, as specified in Alarm IRP: Information Service 3GPP TS 32.111-2 [1], is one member of this category.

The purpose of Notification IRP is to define an interface through which an IRPManager (typically a network management system) can subscribe to IRPAgent (typically an Element Manager (EM) or a NE) for receiving network events. It also specifies attributes carried in the network events. These attributes are common among all event categories. Attributes that are specific to a particular event category are not part of the present document. For example, `perceivedSeverity` is an attribute specific for alarm event category. This attribute is not defined the present document but in Alarm IRP 3GPP TS 32.111-2 [1].

---

# 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 32.111-2: "Alarm IRP: Information Service".
- [2] Void
- [3] ITU-T Recommendation X.734 (09/92): "Information technology - Open Systems Interconnection - Systems management: Event report management function".
- [4] 3GPP TS 32.106-8: "Name Convention for Managed Objects".
- [5] Void
- [6] OMG: "OMG Notification Service".
- [7] 3GPP TS 32.101: "3G Telecom Management principles and high level requirements".
- [8] 3GPP TS 32.102: "3G Telecom Management architecture".
- [9] 3GPP TS 32.106-1: "3G Configuration Management: Concept and Requirements".
- [10] ITU-T Recommendation X.730: "Object Management Function".
- [11] ITU-T Recommendation X.731: "State Management Function".
- [12] Void
- [13] ITU-T Recommendation X.733: "Alarm Reporting Function".
- [14] ITU-T Recommendation X.736: "Security Alarm Reporting Function".

## 3 Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the following terms and definitions apply. For terms and definitions not found here, please refer to 3GPP TS 32.101 [7], 3GPP TS 32.102 [8] and 3GPP TS 32.106-1 [9].

**IRPAgent:** See 3GPP TS 32.102 [8].

**IRPManager:** See 3GPP TS 32.102 [8].

**Event:** It is an occurrence that is of significance to network operators, the NEs under surveillance and network management applications. Events can indicate many types of network management information, such as network alarms, network configuration change information and network performance data.

**Extended Event Type:** ITU-T TMN defines event types. Examples are: Object Creation, Object Deletion, Attribute Value Change, State Change, Communications Alarm, Processing Error Alarm, Environmental Alarm, Quality of Service Alarm, Equipment Alarm, Integrity Violation, Security Violation, Time Domain Violation, Operational Violation, Physical Violation. Valid values of this set are controlled by ITU-T.

The 3GPP Working Group SA5's (Telecommunication Management) work on IRP requires definitions beyond those ITU-T defined event types. Examples are:

- Indicate alarm acknowledgement state changes;
- Indicate Alarm List (defined in Alarm IRP: IS 3GPP TS 32.111-2 [1]) has rebuilt successfully.

This set is called `extendedEventType`. Valid values for this set are specified by this IRP.

**Notification:** It refers to the transport of events from event producer to consumer (receiver). In this IRP, notification is used to carry network events from IRPAgent to IRPManager. Producer sends notifications to consumers as soon as there are new events occur. Consumer does not need to check ("pull") for events.

It may be reused if there is no requirement that the previous notification using that Notification identifier be correlated with future notifications. Generally, IRPAgent should choose it to ensure uniqueness over as long a time as is feasible for the IRPAgent.

**Notification Category:** One Notification Category defines the set of all event types and all extended event types specified by one IRP. Neither an event type nor an extended event type may belong to more than one Notification Category.

**Qualifiers:** Qualifiers for operations, notifications and attributes (whether they are Mandatory(M)/ Conditional(C)/ Optional(O)) are defined in the present (Information Service) document, but not for corresponding parameters in the solution set documents (as they are meaningless there). Mandatory and Conditional qualifiers shall always be the same in other IRPs using items from the Notification IRP IS (the present document), but Optional qualifiers may in the other IRPs be set to either Optional or Mandatory.

### 3.2 Abbreviations

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

C	Conditional
CM	Configuration Management
CORBA	Common Object Request Broker Architecture
DN	Distinguished Name
EM	Element Manager
FM	Fault Management
IDL	Interface Definition Language

IRP	Integration Reference Point
IS	Information Service
ITU-T	International Telecommunication Union, Telecommunication Standardisation Sector
M	Mandatory
MIB	Management Information Base
MIM	Management Information Model
MOC	Managed Object Class
MOI	Managed Object Instance
NE	Network Element
NM	Network Manager
NR	Network Resource
NRM	Network Resource Model
O	Optional
OMG	Object Management Group
SS	Solution Set
TMN	Telecommunications Management Network
UML	Unified Modelling Language (OMG)

## 4 System overview

### 4.1 System context for Notification

Figure 1 and Figure 2 identify System contexts of Notification IRP in terms of implementations called IRPAgent and IRPManager.

“IRPManager” depicts a process that interacts with IRPAgent for the purpose of receiving network Notifications via this IRP. IRPAgent detects network events. IRPAgent sends IRPManagers notifications carrying the events. Examples of IRPManagers can be a process running supporting network Notification logging device or supporting network Notification viewing devices (such as a local craft terminal) or a process running within a Network Manager (NM) as shown in Figure 1 and Figure 2. IRPAgent implements and supports this IRP. IRPAgent can run within one Element Manager (EM) with one or more NEs (see Figure 1) or run within one NE (see Figure 2). In the former case, the interfaces (represented by a thick dotted line) between the EM and the NEs are not subject of this IRP. Whether EM and NE share the same hardware system is not relevant to this IRP either. By observing the interaction across the IRP, one cannot deduce if EM and NE are integrated in a single system or if they run in separate systems.

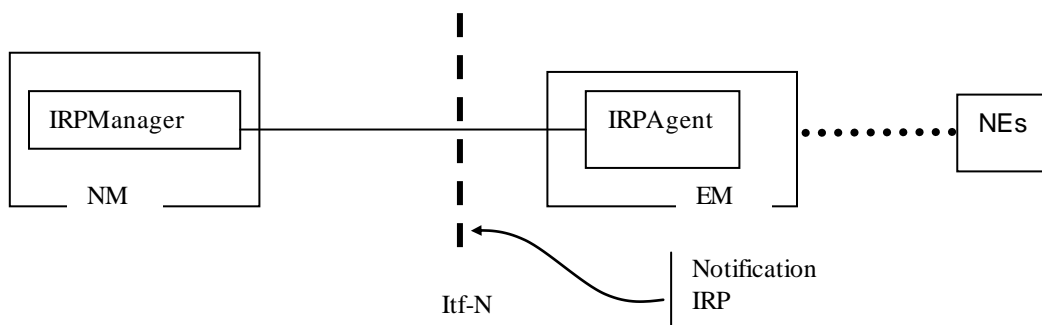
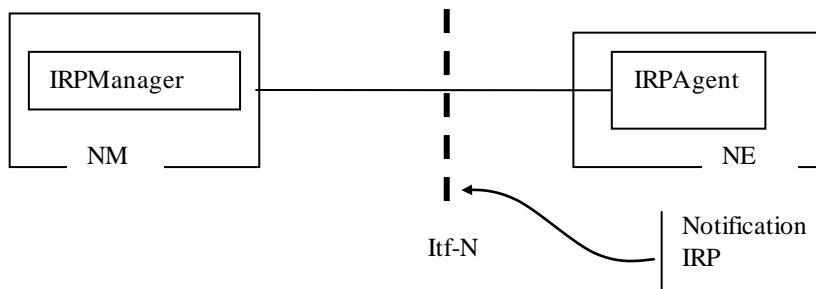


Figure 1: System Context A



**Figure 2: System Context B**

This interface supports the following implementation strategies.

- One IRPAgent supports emission of different categories of Notifications, such as alarms (as specified in 3GPP TS 32.111-2 [1]) and others.
- One IRPAgent supports emission of one specific category of Notification. For example, one IRPAgent implementation only emits alarms specified in 3GPP TS 32.111-2 [1]. Another IRPAgent implementation emits configuration status change notifications.
- IRPManager can specify the categories of notifications it wants to receive using `subscribe` operation. In the case IRPManager does not specify the notification category in `subscribe`, IRPAgent will then emit all categories of notifications that IRPAgent handles. This implementation is SS dependent.
- IRPManager can query the categories of notification supported by IRPAgent. This implementation is Solution Set (SS) dependent.

The Notification IRP defines attributes, carried in notifications that are common in all categories of notifications.

Attributes specific to a particular category of notification shall be specified in corresponding IRP (such as Alarm IRP IS – see 3GPP TS 32.111-2 [1]) using the Notification IRP. Those IRP also define the protocol interaction via which IRPManager receives the notifications.

## 5 Modelling approach

This clause identifies the modelling approach adopted and used in this IRP.

This IRP bases its design on work captured in ITU-T Recommendation X.734 [3], OMG Notification Service [6]. The central design ideas are:

- Separation of notification Consumers (IRPManagers) from Producers (IRPAgents);
- Notifications are sent to IRPManagers without the need for IRPManagers to periodically check for new notifications.
- Common characteristics related to notifications in all other IRPs are gathered in one IRP (the present document).

## 6 IRP Information Service

### 6.1 Interfaces

Figure 3 illustrates the operations and notifications defined as interfaces implemented and used by IRPAgent and IRPManager. Parameters and return status are not indicated. Interface in IRP Information Service is identical to concept conveyed by stereotype `<<interface>>` of UML.



One interface, called `NotificationIRPOperation`, is defined. This interface defines operations implemented by `IRPAgent` and used (or called by) `IRPManager`.

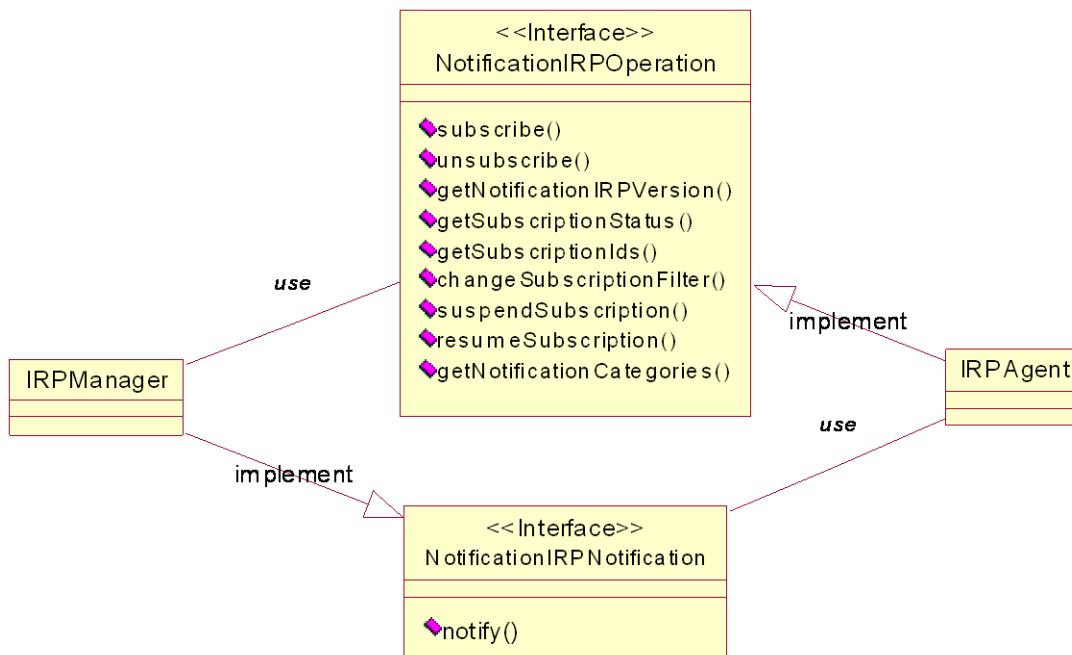


Figure 3: Protocol Independent Interface

### 6.1.1 NotificationIRPOperation Interface

#### 6.1.1.1 Operation subscribe (M)

`IRPManager` invokes this operation to establish subscription to receive network events via notifications, under the filter constraint specified in this operation. How `IRPManager` discovers the `IRPAgent`'s address or reference (so that `IRPManager` can invoke this operation) is outside the scope of the present document.

Table 1: Parameters of `subscribe`

Name	Qualifier	Purpose
<code>managerReference</code>	Input, M	It specifies the reference of <code>IRPManager</code> to which <code>IRPAgent</code> shall send events.
<code>timeTick</code>	Input, O	It specifies the value of a timer hold by <code>IRPAgent</code> for the subject <code>IRPManager</code> . This value defines a time window within which <code>IRPManager</code> intends to invoke <code>getSubscriptionStatus</code> (or <code>subscribe</code> ) operation. <code>IRPAgent</code> shall reset the timer, with <code>timeTick</code> , when it receives the <code>getSubscriptionStatus</code> (or <code>subscribe</code> ) operation from the subject <code>IRPManager</code> . If the timer expires, <code>IRPAgent</code> may delete its resources allocated to the <code>IRPManager</code> and consider <code>IRPManager</code> as if it has invoked <code>unsubscribe</code> operation. In such case, <code>IRPManager</code> will not receive further notification. <code>IRPManager</code> needs to invoke <code>subscribe</code> operation again. The value is in unit of whole minute. If the value is between 1 and 15, <code>IRPAgent</code> considers it to be 15. If the parameter is absent or if the parameter is present but its value is negative or 0, <code>IRPAgent</code> shall treat <code>timeTick</code> value as infinite, i.e., timer will never expire and <code>IRPAgent</code> needs other means to decide when to delete resources allocated to the <code>IRPManager</code> .

notificationCategories	Input, O	It identifies one or more Notification Categories (see also definition in subclause 3.1). If the parameter is absent, IRPAgent shall consider IRPManager is subscribing to all notification categories supported by IRPAgent.
filter	Input, O	It specifies a filter constraint that IRPAgent shall use to filter notification of the category specified in notificationCategory parameter. IRPAgent shall notify IRPManagers if the event satisfies the filter constraint. If this parameter is absent, then no filter constraint shall be applied. Valid filter constraint grammars are specified by individual notification IRP SS, e.g. Notification IRP: CORBA SS.
subscriptionId	Output, M	It holds an unambiguous identity of this subscription. IRPManager can invoke operations (e.g., suspendSubscription) using this identity. In normal usage, IRPManager shall not provide this identity to another IRPManager such that the second IRPManager can invoke operations using it.
status	Output, M	(a) Operation succeeded in that the requested subscription has been established successfully AND that IRPAgent is emitting categories of notification specified by IRPManager via the notificationCategory parameter AND that the filter, if present, contains a valid filter constraint. (b) Operation failed because IRPManager is already in subscription, i.e., IRPAgent detects that there is an existing subscription carrying the same managerReference and in subscription for the same notificationCategory. (c) Operation failed because of other specified or unspecified reason.

### 6.1.1.2 Operation unsubscribe (M)

The IRPManager invokes this operation to cancel subscriptions. The IRPManager can cancel one subscription by providing the corresponding subscriptionId or all subscriptions by leaving the subscriptionId parameter absent.

**Table 2: Parameters for unsubscribe**

Name	Qualifier	Purpose
managerReference	Input, M	It specifies the reference of IRPManager. IRPManager shall supply its valid managerReference. This is the necessary requirement for the operation to be successful.
subscriptionId	Input, O	It carries the subscriptionId carried as the OUT parameter in the subscribe operation. IRPManager shall supply a specific subscriptionId if IRPManager wants to unsubscribe that particular subscription. IRPManager shall not supply subscriptionId (the parameter is absent) if it wants to unsubscribe all subscriptions established between IRPAgent and this managerReference.
status	Output, M	(a) Operation succeeded in that subscription is cancelled successfully. (b) Operation failed because of specified or unspecified reason.

### 6.1.1.3 Operation getNotificationIRPVersion (M)

IRPManager wishes to find out the Notification IRP SS versions supported by IRPAgent. IRPAgent shall respond with a list of Notification IRP SS version(s).

**Table 3: Parameters for getNotificationIRPVersion**

Name	Qualifier	Purpose
versionNumberList	Output, M	It indicates one or more SS version numbers supported by the IRPAgent. In Release 99, it shall contain only one version number.

status	Output, M	(a) Operation succeeded in that <code>versionNumberList</code> contains valid result. (b) Operation failed. Output parameter <code>versionNumberList</code> may contain invalid result.
--------	-----------	--

#### 6.1.1.4 Operation `getSubscriptionStatus` (O)

IRPManager invokes this operation to query the subscription status of a particular subscription.

IRPManager can get similar service by invoking `subscribe` operation. However, the following differences are noted.

- Operation `subscribe` uses `managerReference` and this operation uses `subscriptionId`.
- If IRPAgent has lost IRPManager's reference, IRPManager use of `subscribe` operation may result in establishment of another subscription. Using this operation does not establish another subscription.
- IRPManager can use `getSubscriptionStatus` operation to know about the filter constraint in effect, the state of subscription (i.e., if subscription is suspended/inactive or resumed/active), the `timeTick` value that may be set at `subscribe` invocation time and the notificationCategory currently in used in the subscription.

**Table 4: Parameters for `getSubscriptionStatus`**

Name	Qualifier	Purpose
<code>subscriptionId</code>	Input, M	It carries the <code>subscriptionId</code> carried as the output parameter in the <code>subscribe</code> operation.
<code>notificationCategoryList</code>	Output, M	It identifies the notificationCategory or notificationCategories supported in this subscription.
<code>filterInEffect</code>	Output, O	It contains the filter constraint currently active. If it is absent, IRPManager shall not apply any filter constraint to notifications emitted towards the subject IRPManager.
<code>subscriptionState</code>	Output, O	It indicates if the subscription is in "suspended" or "not-suspended".
<code>timeTick</code>	Output, O	It carries the same value as the one in <code>subscribe</code> operation.
status	Output, M	(a) Operation is successful and IRPAgent has valid values for all output parameters. (b) Operation is unsuccessful in that IRPAgent has no knowledge of the subscription.

#### 6.1.1.5 Operation `getSubscriptionIds` (O)

IRPManager invokes this operation to get the values of all still valid `subscriptionIds` assigned by IRPAgent as result of previously `subscribe` operations performed by this IRPManager.

**Table 5: Parameters for `getSubscriptionIds`**

Name	Qualifier	Purpose
<code>managerReference</code>	Input, M	It specifies the reference of IRPManager that requests the list of identifiers of active subscriptions related to this IRPManager.
<code>subscriptionIdList</code>	Output, M	It carries a list of the <code>subscriptionId</code> , each assigned as OUT parameter in previous <code>subscribe</code> operations invoked by the current IRPManager. This value should contain no information if the IRPManager did not yet subscribed to that System or System lost all subscription related information.
status	Output, M	(a) Operation succeeded in that the value contained in OUT parameter is valid. (b) Operation failed because subscription information is lost or IRPAgent cannot complete the operation for other reasons. In this case, the OUT parameter shall contain no information.

### 6.1.1.6 Operation `changeSubscriptionFilter` (O)

IRPManager invokes this operation to replace the present filter constraint with a new one.

**Table 6: Parameters for `changeSubscriptionFilter`**

Name	Qualifier	Purpose
<code>subscriptionId</code>	Input, M	It carries the <code>subscriptionId</code> carried as the OUT parameter in the <code>subscribe</code> operation.
<code>filter</code>	Input, M	See description of Table 1: Parameters of <code>subscribe</code> .
<code>status</code>	Output, M	(a) Operation succeeded in that IRPAgent is now producing events based on the new filter constraint. (b) Operation failed in that, for unspecified reason, the new filter constraint cannot be installed. The old filter constraint, if present before this operation, is still in effect.

### 6.1.1.7 Operation `suspendSubscription` (O)

IRPManager invokes this operation to request IRPAgent to stop emission of notifications. IRPAgent may lose notification(s) if subscription is suspended.

**Table 7: Parameters for `suspendSubscription`**

Name	Qualifier	Purpose
<code>subscriptionId</code>	Input, M	It carries the <code>subscriptionId</code> carried as the OUT parameter in the <code>subscribe</code> operation.
<code>status</code>	Output, M	(a) Operation succeeded in that IRPAgent has suspended emission of notifications. (b) Operation failed in that, for unspecified reason, IRPAgent has not suspended emission of events.

### 6.1.1.8 Operation `resumeSubscription` (O)

IRPManager invokes this operation to request IRPAgent to resume emission of notifications. If the `Subscription State` is "not-suspended", IRPAgent shall return status successful and ignore this invocation. If `Subscription State` is "suspended", IRPAgent shall return status successful, change the `Subscription State` to "not-suspended" and resume emission of notifications.

**Table 8: Parameters for `resumeSubscription`**

Name	Qualifier	Purpose
<code>subscriptionId</code>	Input, M	It carries the <code>subscriptionId</code> carried as the OUT parameter in the <code>subscribe</code> operation.
<code>status</code>	Output, M	(a) Operation succeeded in that IRPAgent is has resumed emission of events. (b) Operation failed in that, for unspecified reason, IRPAgent cannot resume emission of events.

### 6.1.1.9 Operation `getNotificationCategories` (O)

IRPManager invokes this operation to query the categories of notification supported by IRPAgent. IRPManager does not need to be in subscription to invoke this operation.

**Table 9: Parameters for getNotificationCategories**

Name	Qualifier	Purpose
notificationCategoryList	Output, M	It identifies the list of notification categories supported by IRPAgent (see also definition in subclause 3.1). If this parameter value contain no information, then the meaning is that IRPAgent does not support any notification category at the moment.
eventTypeList	Output, O	It contains a list of elements. Each element is a list of eventType. The number of element shall be identical to that of output parameter notificationCategoryList. The n-th element of this list relates to the n-th element of the notificationCategoryList. IRPAgent shall not use arbitrarily any eventType(s) in this n-th element. IRPAgent shall use the same list of eventType(s) specified in the IRP document identified by the n-th element of the notificationCategoryList. If the n-th element contains no information, it implies IRPAgent is not providing explicit identification of eventType(s) of the corresponding notificationCategory. If this parameter is absent or contains no information, it implies that IRPAgent is not providing explicit identification of eventType(s).
extendedEventTypeList	Output, O	It contains a list of element. Each element is a list of extendedEventType. The number of element shall be identical to that of output parameter notificationCategoryList. The n-th element of this list relates to the n-th element of the notificationCategoryList. IRPAgent shall not use arbitrarily any extendedEventType in this n-th element. IRPAgent shall use the same list of extendedEventType specified in the IRP document identified by the n-th element of the notificationCategoryList. If the n-th element contains no information, it implies IRPAgent is not providing explicit identification of extendedEventType(s) of the corresponding notificationCategory. If this parameter is absent or contains no information, it implies that IRPAgent is not providing explicit identification of extendedEventType(s).
status	Output, M	(a) Operation succeeded in that the output parameter contains valid information. (b) Operation failed in that the output parameter does not contain valid information.

## 6.1.2 NotificationIRPNotification Interface

### 6.1.2.1 Notification notify

IRPAgent notifies the subscribed IRPManager that an event has occurred and that the event has satisfies the filter constraints used for this subscription. One event example is the notification defined in Alarm IRP: IS (3GPP TS 32.111-2 [1]).

The present document does not further specify this notify. Other IRPs using the Notification IRP, such as Alarm IRP: IS (3GPP TS 32.111-2 [1]), shall specify this notify, in particular, the specific parameters carried in notification, for use in their context.

The present document shall specify, in subclause 6.1.2.2, attributes commonly carried in parameters of all notifications.

### 6.1.2.2 Notification Attributes

Information about network events is carried in notification containing parameters of multiple attributes. This IRP specifies attributes that are commonly found in notifications defined by other IRPs. Collectively, they are called Notification Header. Other IRPs using the Notification IRP, such as Alarm IRP: IS (3GPP TS 32.111-2 [1]), shall specify the attributes used in the notification including:

- Identification and qualification of notification Header attributes for their use;
- Specification and qualification of other attributes relevant for their use.

#### 6.1.2.2.1 managedObjectClass (M)

This parameter specifies the class of the Managed Object (MO) in which the network event occurred. This attribute is filterable.

#### 6.1.2.2.2 managedObjectInstance (M)

This parameter specifies the instance of the MO in which the network event occurred. This attribute is filterable.

#### 6.1.2.2.3 notificationId (O)

This parameter provides an identifier for the notification, which may be carried in the `correlatedNotifications` parameter (see below) of future notifications. Attribute `notificationId` shall be chosen to be unique across all notifications of a particular managed object throughout the time that correlation is significant.

It uniquely identifies this notification from other notifications generated by the subject MO.

If IRPManager receives notifications from one IRPAgent, IRPManager shall use `notificationId` and `managedObjectInstance` to uniquely identify all received notifications.

If IRPManager receives notifications from multiple IRPAgents and notifications of each MO are reported at most through one IRPAgent, IRPManager shall use `notificationId` and `managedObjectInstance` to uniquely identify all received notifications.

If IRPManager receives notifications from multiple IRPAgents and notifications of one or more MOs are reported through two or more IRPAgents, IRPManager shall use `notificationId`, together with `managedObjectInstance` and the identity of IRPAgent, to uniquely identify all received notifications. Attribute `systemDN`, if present, carries IRPAgent's identify. If `systemDN` is absent, IRPManager needs other means, which are outside the scope of this IRP, to determine the identity of IRPAgent.

If and when the value of this can be re-used is specified in SSs.

This attribute is filterable.

#### 6.1.2.2.4 eventTime (M)

It indicates the event occurrence time. The semantics of Generalised Time specified by ITU-T shall be used here.

This attribute is filterable.

#### 6.1.2.2.5 systemDN (C)

It carries the Distinguished Name (DN) of IRPAgent that detects the network event and generates the notification. See 3GPP TS 32.106-8 [4] for name convention regarding DN.

This attribute is filterable.

#### 6.1.2.2.6 eventType (M)

It carries identification of the type of event reported by the notification. Allowed event types are ITU-T TMN defined event types. Examples of ITU-T TMN event types are:

- Object Creation (ITU-T Recommendation X.730 [10])
- Object Deletion (ITU-T Recommendation X.730 [10])
- Attribute Value Change (ITU-T Recommendation X.731 [11])
- State Change (ITU-T Recommendation X.731 [11])

- Communications Alarm (ITU-T Recommendation X.733 [13])
- Processing Error Alarm (ITU-T Recommendation X.733 [13])
- Environmental Alarm (ITU-T Recommendation X.733 [13])
- Quality of Service Alarm (ITU-T Recommendation X.733 [13])
- Equipment Alarm (ITU-T Recommendation X.733 [13])
- Integrity Violation (ITU-T Recommendation X.736 [14])
- Security Violation (ITU-T Recommendation X.736 [14])
- Time Domain Violation (ITU-T Recommendation X.736 [14])
- Operational Violation (ITU-T Recommendation X.736 [14])
- Physical Violation (ITU-T Recommendation X.736 [14])

Each IRP document using the Notification IRP, such as Alarm IRP: IS (3GPP TS 32.111-2 [1]), identifies which eventType shall be used for that IRP.

This attribute is filterable.

#### 6.1.2.2.7 extendedEventType (M)

IRPAgent, in certain situations, may generate notifications of types whose semantics are extended beyond those defined by ITU-T event types. Examples are:

- Indicate alarm acknowledgement state changes
- Indicate AlarmList of AlarmIRPAgent has rebuilt successfully.

This attribute carries the required extension.

Each IRP document using the Notification IRP, such as Alarm IRP: IS (3GPP TS 32.111-2 [1]), defines the extended event types required.

This attribute is filterable.

### 6.1.3 Behaviour

#### 6.1.3.1 IRPAgent supports multiple subscriptions with one IRPManager

An IRPManager can have multiple managerReferences. IRPManager can invoke subscribe operations using different managerReferences resulting in multiple subscriptions. As far as IRPAgent is concerned, the IRPAgent is sending alarms to multiple "places".

If IRPManager invokes multiple subscriptions with identical managerReference and notificationCategory combination, all but one subscription shall fail with exception indicating that the IRPManager is already in subscription.

If IRPManager has established subscription by invoking subscribe with notificationCategory parameter absent, subsequent subscribe, either with notificationCategory absent or present, using the same managerReference, shall fail. IRPAgent shall throw exception indicating that the IRPManager is already in subscription.

IRPManager controls the filter constraint via subscribe and changeSubscriptionFilter operations.

### 6.1.3.2 Support of packing multiple notifications

It should be possible to pack multiple notifications together for sending to NM. This provides more efficient use of data communication resources. In order to pack multiple notifications, an EM/NE configurable parameter defines the maximum number of notifications to be packed together. Additionally an EM/NE configurable parameter defines the maximum time delay before the notifications have to be sent.

### 6.1.3.3 IRPAgent supports emission of multiple Notification categories

IRPAgent supporting this IRP may emit multiple categories of notifications. For example, it may emit notification defined in Alarm IRP 3GPP TS 32.111-2 [1]. IRPAgent supports mechanism that IRPManager can use to determine the categories of notifications supported by IRPAgent. IRPAgent also supports mechanism that IRPManager can use to specify the categories of notifications IRPAgent should emit to IRPManager during subscription.

### 6.1.3.4 Subscription list loss

IRPAgent can lose the list of `managerReference` that identifies current IRPManagers under subscription. Under this condition, IRPAgent is incapable of sending events to the affected subscriber(s).

This Notification IRP recommends that IRPManager should invoke the `getSubscriptionStatus` operation periodically to confirm that IRPAgent still has the IRPManager's reference in its list. In case IRPManager does not obtain a positive confirmation, IRPManager should assume that IRPAgent has lost the IRPManager's reference. In this case, IRPManager should invoke `unsubscribe` and then `subscribe` operation again.

This IRP does not recommend the frequency IRPManager should use to invoke `getSubscriptionStatus` operation.

### 6.1.3.5 Notification ordering

Under normal operations, an IRPAgent shall send, to each IRPManager, notifications in the same order they were generated, i.e. in the First-In, First-Out order. Notifications of one Event Type and/or Extended Event Type shall not be given priority over other Event Types and/or Extended Event Types.



## Annex A (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
Mar 2000	S_07	SP-000012	-		Approved at TSG SA #7 and placed under Change Control	2.0.0	3.0.0
Mar 2000		-			cosmetic	3.0.0	3.0.1
Jun 2000	S_08	SP-000242	002		Split of TS - Part 2: Notification Integration Reference Point (IRP): Information Service (IS)	3.0.1	3.1.0
Sep 2000	S_9	SP-000518	001		Consistent description of Event types and Extended event types	3.1.0	3.2.0
Sep 2000	S_9	SP-000518	002		Correction of parameter inconsistency in operation unsubscribe	3.1.0	3.2.0
Mar 2001	S_11	SP-010027	003		Add Information Service QOS specification	3.2.0	3.3.0
Mar 2001	S_11	SP-010027	004		Remove the reference to Relationship Change Notifications (ITU-T X.732)	3.2.0	3.3.0