

Guidelines for Using the OMG Notification and Telecom Log Service

Using the Notification Service

Basic Concepts of the OMG Notification Service

This interface uses the notification service specification identified by the OMG as

formal/02-08-04: Notification Service, version 1.0.1
Date: August 2002
Version 1.0.1

This is accessible at http://www.omg.org/technology/documents/formal/notification_service.htm.

Overview

An overview for explanation of the basic concepts is shown in Figure 1.

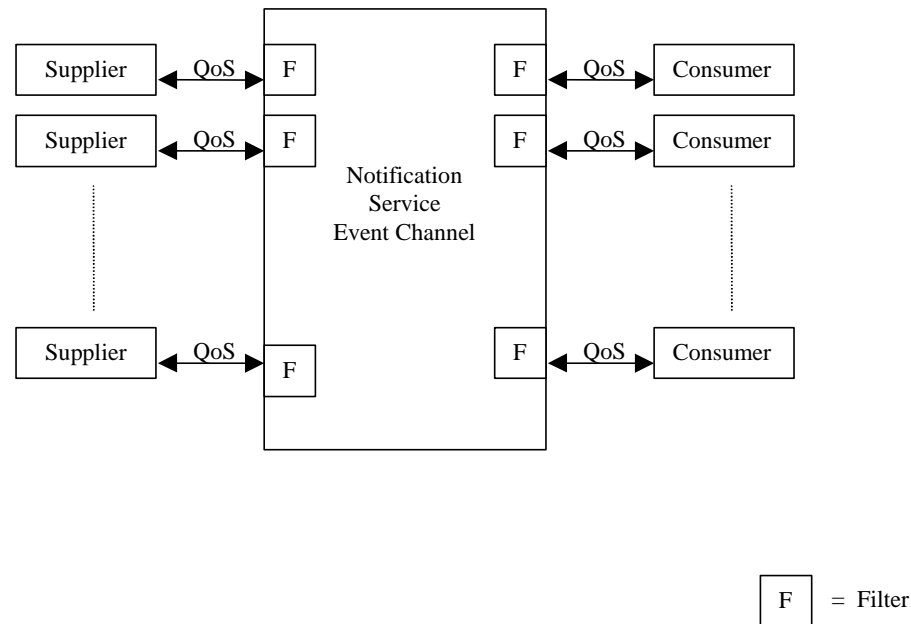


Figure 1. Overview Notification Service (acc. to OMG 2.2 specification)

The notification service is an extension (by inheritance) of the simple OMG COS::Event Service which can be used for event forwarding. It was proven that for Telecom applications the COS::Event Service could not be used since for example it does not allow for events to be filtered at the supplier side.

Therefore OMG has defined the Notification Service which adds the following features to the OMG event channel:

- Selection of a wide range of QoS (Quality of Service) and administrative properties for reliability and event queue management.
- Structured Event Types and powerful filter constraint language.

The basic event forward concept of an event channel provides a full de-coupling of communication between supplier and consumer. An event is sent by a supplier and transferred to any number of consumers by the channel. A Supplier needs no knowledge about connected consumers and a consumer has no knowledge of which supplier has generated an event. The channel allows consumers to register interest in receiving events. The channel gets events from connected suppliers and forwards them to registered consumers.

Supplier, consumer and notification channel are implemented as ORB applications with well defined IDL interfaces in the OMG COS::NotificationService standard.

Two models for event forwarding are supported:

- Push Model
- Pull Model

In the Push model the consumer waits passively until it gets an event. The supplier actively generates events and sends them to the channel which invokes the consumer.

In the Pull Model a consumer actively requests an event. (Supplier waits until it gets an event).

The channel provides proxy objects for each consumer or supplier (not shown in the figure). Filter objects can be associated to proxy objects for suppliers and consumers (see next chapter).

Structured Events

The most important extension of the event service is the support of the new Structured Event Types and event filtering. The contents of a structured event are shown in Figure 2. It consists of an event header and an event body. The event header has a fixed header with information that identifies the event and a variable header with a per event QoS setting, a header with filter settings and the event data in the event body which is composed of a filterable part a remaining body (can not be filtered).

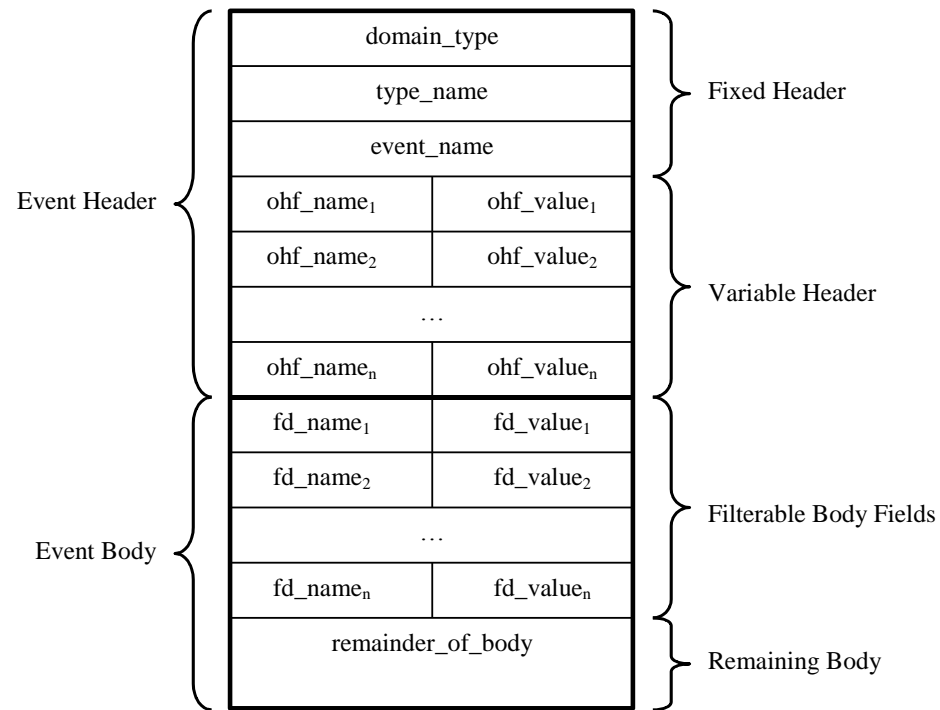


Figure 2. Structure event definition (OMG)

The `type_name` and `event_name` can be used for simple type match filtering. For more complex filters the notification service filtering language can be used which is basically an extension of the OMG Trader Constraint Language. (For example “`$type_name = 'Alarm'` and `$probableCause = 'AIS'`”.)

An event (for instance an alarm) can be mapped to the filterable body part or to the `remainder_of_body` (which can not be filtered). The Mapping is described in the next chapter of the document. It can be shown by measurements that the performance of the notification service is very much dependent on this mapping and the way that filters are applied to structured events.

Recommendations for Usage of Notification Service

General Recommendations

The following recommendations are given:

- Use only Push Model in EMS.
- Use one channel to avoid ordering problems.
- Use one notification service per EMS.

QoS and Admin Parameters

The following parameters are to be used in order to provide an efficient way of using the notification service (proven by test-runs):

- Use Persistency for EventReliability and ConnectionReliability parameter, as needed.
- All events shall have the same priority (0) to avoid ordering problems.
- Use FifoOrder for order and discard policy.
- Use FALSE for STARTTIMESUPPORTED.
- Use 0 for maximum batch size, i.e. batching is not used.
- Use 0 for Pacing Interval, i.e. batching is not used.
- The RejectNewEvents admin property should be set to TRUE so that the EMS will know that the notification channel is not accepting new events anymore.
- The TimeOut QoS property should be set to 30 minutes for NT_TCA and NT_ALARM events; it should be set to 24 hours for all other events.

StructuredEvent

This document describes how to use the structures and fields of this notification service in the context of this interface.

A structured event is divided in two main parts which in turn are subdivided in two parts as shown in figure 2. The event header consists of the fixed and the variable event header. The event body is composed of the filterable body and the remaining body.

The structured event must be filled out as follows:

header:

fixed_header:

event_type:

domain_name:

Must be set to "tmf_mtnm"

type_name:

Must be set to one of:

"NT_OBJECT_CREATION"
"NT_OBJECT_DELETION"
"NT_ATTRIBUTE_VALUE_CHANGE"
"NT_STATE_CHANGE"
"NT_ROUTE_CHANGE"
"NT_PROTECTION_SWITCH"
"NT_EPROTECTION_SWITCH"
"NT_TCA"
"NT_ALARM"
"NT_FILE_TRANSFER_STATUS"
"NT_PM_SC"
"NT_BACKUP_STATUS"
"NT_HEARTBEAT"

"NT_PULL_FILE_TRANSFER_STATUS"

The NMS should ignore unrecognized notifications. This is required to provide forward compatibility.

event_name:

Must be set to empty string

variable_header

The variable event header is meant for QoS parameter and will not contain any parts of the notifications defined so far in the IDL.

filterable_data:

The content of the event body depends on the event type. Thus there is no need to define a union of notifications but a body for every event type only. Note that the filterable body is composed of name/value pairs where the value is of type ANY. (Test runs show that unions should be avoided in order to avoid performance problems).

When a field is marked optional, it can be omitted from the name/value pair list, which indicates that it does not apply, that its value is unknown, or that it is unsupported.

Every member of the filterable body can be filtered on in any combination. However for some filters there may be considerable performance penalties.

The NMS should ignore unrecognized fields of filterable_data in a notification. This is required for forward compatibility.

The contents of this field are dependant on the **type_name**. The field must be filled out as follows:

For **type_name** = "NT_OBJECT_CREATION"

Holds details of object which is being created.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is

Name	Type	Description
		supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the object being created.
"objectType"	notifications::ObjectType_T	Identifies the type of object being created. This parameter is only used for object types define in v2.1 and earlier.
"objectTypeQualifier"	notifications::ObjectTypeQualifier_T	Identifies the type of object being created. This parameter is only used for new object types defined in v3.0 and beyond.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"edgePointRelated"	boolean	TRUE if this event relates to a PTP that is an edge point or to a PG that contains a PTP that is an edge point. FALSE otherwise; this field is optional in this case.

For **type_name** = "NT_OBJECT_DELETION"

Holds details of object which is being deleted

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the object being deleted.
"objectType"	notifications::ObjectType_T	Identifies the type of object being deleted. This parameter is only used for object types defined in v2.1 and earlier.
"objectTypeQualifier"	notifications::ObjectTypeQualifier_T	Identifies the type of object being deleted. This parameter is only used for new object types defined in v3.0 and beyond.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case

Name	Type	Description
		where the NE does not report time, this field is optional or an empty string can be reported.
"edgePointRelated"	boolean	TRUE if this event relates to a PTP that is an edge point or to a PG that contains a PTP that is an edge point. FALSE otherwise; this field is optional in this case.

For **type_name** = "NT_ATTRIBUTE_VALUE_CHANGE"

Holds details of an attribute value that has changed.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the object reporting an AVC.
"objectType"	notifications::ObjectType_T	Identifies the type of object reporting

Name	Type	Description
		an AVC. This parameter is only used for object types defined in v2.1 and earlier.
"objectTypeQualifier"	notifications::ObjectTypeQualifier_T	Identifies the type of object for which an AVC is being reported. This parameter is used only for new object types defined in v3.0 and beyond.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"edgePointRelated"	boolean	TRUE if this event relates to a PTP that was or became an edge point, to a CTP that is contained in PTP that is an edge point, or to a PG that contains a PTP that is an edge point. FALSE otherwise; this field is optional in this case.
"attributeList"	notifications::NVList_T	The list of names of attributes that have changed and their complete new values in the modified object structure. Special behavior is defined for attributes "transmissionParams" and "additionalInfo", see below. See note below for details of which attributes must be supported.

In an AVC notification, the "transmissionParams" attribute's value may not contain the complete value of the attribute. Rather, it only indicates transmission parameters that have changed. The value always contains a complete list of the TP's layer rates; but for each layer, only those transmission parameters that have been deleted, changed, or added are specified. A deleted transmission parameter is indicated by a "-" value; a changed or added transmission parameter has its new value specified. All transmission parameters not listed have not changed.

Similarly, in an AVC notification, the "additionalInfo" attribute's value may not contain the complete value of the attribute. Rather, it only indicates NameAndStringValue_Ts that have been deleted, changed, or added. A deleted NameAndStringValue_T is indicated by a "-" value; a changed or added NameAndStringValue_T has its new value specified. All NameAndStringValue_Ts not listed have not changed.

For **type_name** = "NT_STATE_CHANGE"

Holds details of a state attribute value that has changed.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the object reporting a State Change.

Name	Type	Description
"objectType"	notifications::ObjectType_T	Identifies the type of object reporting a State Change. This parameter is only used for object types defined in v2.1 and earlier.
"objectTypeQualifier"	notifications::ObjectTypeQualifier_T	Identifies the type of object for which a SC notification is being reported. This parameter is only used for new object types defined in v3.0 and beyond.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"edgePointRelated"	boolean	TRUE if this event relates to a PTP that is an edge point, to a CTP that is contained in a PTP that is an edge point, or to a PG that contains a PTP that is an edge point. FALSE otherwise; this field is optional in this case.
"attributeList"	notifications::NVList_T	The list of names of state attributes that have changed and their complete new values in the modified object structure. See note below for details of which attributes must be supported.

For **type_name** = "NT_ROUTE_CHANGE"

Holds details of the route change.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the SNC object reporting a route change.
"objectType"	notifications::ObjectType_T	Identifies the SNC object reporting a route change.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"routeChangeEvent"	subnetworkConnection::	This identifies which state the notification

Name	Type	Description
	RerouteChangeEvent_T	is emitted against.
"route"	subnetworkConnection::Route_T	This identifies the new route that is chosen, if any; this field is optional otherwise.

For **type_name** = "NT_PROTECTION_SWITCH"

Holds details of a protection switch.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular groupName or protectedTP throughout the time that correlation is significant.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report

Name	Type	Description
		time, this field is optional or an empty string can be reported.
"ProtectionType"	protection::ProtectionType_T	The type of the protection.
"switchReason"	protection::switchReason_T	The reason the switch occurred.
"layerRate"	transmissionParameters::LayerRate_T	The layer which this switch is relevant to.
"groupName"	globaldefs::NamingAttributes_T	Identifies the protectionGroup emitting the switch. If protectionType is SNCP, this field is optional or an empty name can be reported.
"protectedTP"	globaldefs::NamingAttributes_T	Identifies the TP that was protected when the switch occurred. For an SNCP, this is always the reliable TP. For a 2F BLSR ring switch notification, this is the TP that is/was inactive during the switch. For a 4FMSSP ring switch notification, this is the worker TP that is/was inactive during the switch. For a 1:N MSP switch notification, this is the worker TP for which the protection switch occurred. For a revertive 1+1 MSP, this is always the worker TP. For a non-revertive 1+1 MSP switch notification, this is the TP that was active before the switch (note that

Name	Type	Description
		after the switch the protected TP changes).
"switchAwayFromTP"	globaldefs::NamingAttributes_T	Identifies the TP being switched away from. For a 2F MSSPRING ring switch, this is the TP that switched. For a 4-Fiber MSSPRING span switch, this is one of the TPs in the MSP 1:N groups (worker or protection). For a 4-Fiber ring switch, this is the worker TP of the span that switched; for example if the East span switches to the West span then this TP is the East-Worker.
"switchToTP"	globaldefs::NamingAttributes_T	Identifies the TP which is being switched to. This identifies the TP that is the active source after the switch, or currently active if no protection switch is currently active. For a 2F BLSR, when the state returns to normal, this is the TP that had switched. For a 4F BLSR ring switch, when the state returns to normal, this is the worker TP of the span that had switched; for example, if the East span had switched to the West span, then this is the East worker TP.

For **type_name** = "NT_EPROTECTION_SWITCH"

Holds details of an equipment protection switch.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular equipment protection group or protected equipment throughout the time that correlation is significant.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"eProtectionGroupType"	protection::EProtectionGroupType_T	The type of the equipment protection. If omitted, M:N is assumed.
"eSwitchReason"	protection::ESwitchReason_T	The reason the switch occurred.
"groupName"	globaldefs::NamingAttributes_T	Identifies the equipment protection

Name	Type	Description
		group emitting the switch.
"protectedE"	globaldefs::NamingAttributes_T	Identifies the equipment that was protected when the switch occurred. For an M:N group, protectedE always identifies the worker equipment instance for which the switch occurred.
"switchAwayFromE"	globaldefs::NamingAttributes_T	Identifies the equipment instance being switched away from.
"switchToE"	globaldefs::NamingAttributes_T	Identifies the equipment instance which is being switched to.

For **type_name** = "NT_TCA"

Holds details of a threshold crossing alert.

The combination of the objectName, layerRate, granularity, pmParameterName, pmLocation, and thresholdType fields must uniquely identify a TCA. Together, these six fields contain information such that any clear of a TCA would correlate to the correct TCA. This means that if a TCA is raised, cleared, raised again, and cleared again, and if the original clear and second TCA were missed, the second clear would clear the first TCA.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the object reporting a TCA.
"nativeEMSName"	string	Identifies the object as portrayed on the EMS user interface.
"objectType"	notifications::ObjectType_T	Identifies the type of object reporting a TCA. This parameter is only used for object types defined in v2.1 and earlier.
"objectTypeQualifier"	notifications::ObjectTypeQualifier_T	Identifies the type of object for which the TCA is being reported. This parameter is only used for new object types defined in v3.0 and beyond.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.

Name	Type	Description
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"isClearable"	boolean	true if the event is clearable (or is itself a clear); false otherwise.
"perceivedSeverity"	notifications::PerceivedSeverity_T	Indicates the severity of the TCA.
"layerRate"	transmissionParameters::LayerRate_T	Carries the details of the threshold that has been crossed
"granularity"	Granularity_T	Carries the details of the threshold that has been crossed
"pmParameterName"	PMPParameterName_T	Carries the details of the threshold that has been crossed
"pmLocation"	PMLocation_T	Carries the details of the threshold that has been crossed
"thresholdType"	PMThresholdType_T	Carries the details of the threshold that has been crossed
"value"	float	Carries the details of the threshold that has been crossed, if known. This parameter is optional.

Name	Type	Description
"unit"	string	Carries the details of the threshold that has been crossed, if known. This parameter is optional.
"acknowledgeIndication"	notifications::AcknowledgeIndication_T	<p>The possible values of acknowledgeIndication are as follows:</p> <p>EVENT_ACKNOWLEDGED - provided in case of manual or auto acknowledgement</p> <p>EVENT_UNACKNOWLEDGED - provided if the event has not been acknowledged but the EMS supports acknowledgement for the event</p> <p>NA - provided in case the EMS does not support acknowledgement for this event</p> <p>Absence of the acknowledgeIndication field in Version 2.0 or 2.1 TCAs shall be interpreted as NA.</p>

For **type_name** = "NT_ALARM"

Holds details of an alarm

An alarm must be reported against the correct object if it is modeled by the EMS.

Alarms against entities which are not modeled by the interface should still be reported using the "AID" objectType. The EMS should ensure that all such entities have a unique value for the "AID". As an example:

```
sourceName[0].name = "EMS"  
sourceName[0].value = "CompanyName/EMSName"  
sourceName[1].name = "ManagedElement"  
sourceName[1].value = "NEName"  
sourceName[2].name = "AID"  
sourceName[2].value = "IdentifierValue"
```

The combination of the objectName, layerRate, probableCause, and probableCauseQualifier fields must uniquely identify an alarm. It is acceptable that the probableCauseQualifier be an empty string, provided that uniqueness (using the remaining three fields) is still guaranteed. Together, these four fields contain information such that any clear of an alarm would correlate to the correct alarm. This means that if an alarm is raised, cleared, raised again, and cleared again, and if the original clear and second alarm were missed, the second clear would clear the first alarm.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the object reporting an alarm
"nativeEMSName"	string	Identifies the object as portrayed on the EMS user interface.
"nativeProbableCause"	string	Identifies the probableCause as portrayed on the EMS user interface.
"objectType"	notifications::ObjectType_T	Identifies the type of object reporting an alarm. This parameter is only used for object types defined in v2.1 or earlier.
"objectTypeQualifier"	notifications::ObjectTypeQualifier_T	Identifies the type of object for which the alarm is being reported. This parameter is only used for new object types defined in v3.0 and beyond.

Name	Type	Description
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"isClearable"	boolean	Indicates if the event is clearable (or is itself a clear).
"layerRate"	transmissionParameters::LayerRate_T	The layer which this alarm is relevant to.
"probableCause"	string	No other string than the ones defined by the TM Forum should be used for this field.
"probableCauseQualifier"	string	Used with objectName, layerRate, and probableCause to uniquely identify an alarm.
"perceivedSeverity"	notifications::PerceivedSeverity_T	Indicates the severity of the alarm.
"serviceAffecting"	notifications::ServiceAffecting_T	Indicates whether the alarm has affected service

Name	Type	Description
"affectedTPList"	globaldefs::NamingAttributesList_T	A list of affected TPs. Contained CTPs are not listed. This field is optional (indicated by an empty string) for all alarms except for alarms on equipment. This is used to indicate a list of TPs affected by an equipment failure for example. If the alarm is an alarm on an equipment that supports PTPs, then the ports (PTPs) supported by this equipment will be listed in this field (irrespective of whether the alarm is Service Affecting or not). The list should be ordered by PTP names (ASCII order).
"additionalText"	string	More information about the alarm. such as, "Unit is mis-mounted".
"additionalInfo"	globaldefs::NVSLIST_T	This allows the communication from the EMS to the NMS of additional information which isn't explicitly modeled. See "Other Parameters into Additional Information Parameter" below. This field is optional, and is now obsolete . The "X.733::*" fields described below or proprietary fields should be used instead.

Name	Type	Description
"X.733::EventType"	string	Classifies the alarm into one of the five basic categories specified in ITU-T X.733. Value is one of the following: <ul style="list-style-type: none">• "communicationsAlarm",• "environmentalAlarm",• "equipmentAlarm",• "processingErrorAlarm",• "qualityofServiceAlarm" This field is optional.
"X.733::SpecificProblems"	notifications::SpecificProblemList_T	Identifies further refinements to the Probable cause of the alarm. (Similar to ProbableCauseQualifier, but this parameter is designed to be human readable and compatible with ITU usage.) This field is optional.
"X.733::BackedUpStatus"	string	Specifies whether or not the object emitting the alarm has been backed-up, and services provided to the user have, therefore, not been disrupted. Value is one of the following: <ul style="list-style-type: none">• "BACKED_UP",• "NOT_BACKED_UP" This field is optional.

Name	Type	Description
"X.733::BackUpObject"	globaldefs::NamingAttributes_T	Specifies the object that is providing back-up services for the object about which the notification pertains. This parameter shall be present when the X733::BackedupStatus parameter is present and has the value "BACKED_UP"; this field is optional otherwise.
"X.733::TrendIndication"	string	Specifies the current severity trend of the object. If present it indicates that there are one or more alarms ("outstanding alarms") which have not been cleared, and pertain to the same object as that to which this alarm ("current alarm") pertains. Value is one of the following: <ul style="list-style-type: none">• "MORE_SEVERE",• "NO_CHANGE",• "LESS_SEVERE" This field is optional.
"X.733::CorrelatedNotifications"	notifications::CorrelatedNotificationList_T	Contains a set of Notification identifiers and, if necessary, their associated object names. This set is defined to be the set of all notifications to which this notification is considered to be correlated. This field is optional.

Name	Type	Description
"X.733::MonitoredAttributes"	notifications::NVList_T	Defines one or more attributes of the managed object and their corresponding values at the time of the alarm. This field is optional.
"X.733::ProposedRepairActions"	notifications::ProposedRepairActionList_T	Used if the cause is known and the system being managed can suggest one or more solutions (such as switch in standby equipment, retry, replace media). This field is optional.
"X.733::AdditionalInfo"	notifications::NVList_T	Allows the inclusion of a set of additional information. (For consistency with X.733). This field is optional.
"rcailIndicator"	boolean	The RCAI indicator has two values, i.e., TRUE (meaning that the alarm is a Root Cause Alarm Indication) or FALSE (meaning the alarm is a raw alarm). Absence of the rcailIndicator in Version 2.0 or 2.1 alarms shall be interpreted as a false value for the RCAI attribute.

Name	Type	Description
"acknowledgeIndication"	notifications::AcknowledgeIndication_T	<p>The possible values of acknowledgeIndication are as follows:</p> <p>EVENT_ACKNOWLEDGED - provided in case of manual or auto acknowledgement</p> <p>EVENT_UNACKNOWLEDGED - provided if the event has not been acknowledged but the EMS supports acknowledgement for the event</p> <p>NA - provided in case the EMS does not support acknowledgement for this event</p> <p>Absence of the acknowledgeIndication field in Version 2.0 or 2.1 alarms shall be interpreted as NA.</p>

For **type_name** = "NT_FILE_TRANSFER_STATUS"

Holds details of file being transferred, transfer status, percent complete and failure reason.

The number of events indicating FT_IN_PROGRESS that will be transferred is EMS dependent, however, at least one event indicating FT_COMPLETED with percentComplete=100, or FT_FAILED with failureReason filled in is mandatory.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed.
"fileName"	string	Name of the file being transferred (this will include the path name). This field should be exactly what was specified in the getHistoryPMDData() request.
"transferStatus"	FileTransferStatus_T	Indicates file transfer status.
"percentComplete"	short	Indicates percent of file transfer complete must be in the range 0..100. Only applies if transferStatus is FT_IN_PROGRESS or FT_COMPLETED; this field is optional otherwise.
"failureReason"	string	This string will contain the failure reason. Only applies if transferStatus is FT_FAILED; this field is optional otherwise.

For **type_name** = "NT_PM_SC"

Holds details of a State Change (SC) to one or more Performance Monitoring Points (PMPs).

This notification is issued against a list of PMP objects as opposed to a second-level MTNM object. The notification can be used in lieu individual SC notifications on a collection of PMPs. The following usage requirements apply:

- The EMS should not use both the NT_PM_SC and individual SC notifications on PMPs to report the same event. It is important to note that EMS could in some cases issue SC notifications on individual PMPs, and at other times, issue SC notifications on a list of PMPs.
- The NT_PM_SC is used to report only one type of change at a time, e.g., PM collection is set to “disabled” for a collection of PMPs, or TCA generation is enabled for a set of PMPs.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed
"pmNameList"	globaldefs::NamingAttributesList_T	This parameter identifies the PMP objects for which the AVC applies.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.

Name	Type	Description
"attributeList"	notifications::NVList_T	<p>The name of the attribute that has changed, and the value to which it has change.</p> <p>three attributes names have been identified:</p> <ul style="list-style-type: none">• clearPMDData (an indication that PM registers have been cleared for a PM select list). This parameter has only one possible value, i.e., TRUE.• monitoringState (an indication that PM data collection has been enabled or disabled for a PM select list). This attribute is either AS_Unlocked (PM data collection enabled), or AS_Locked (PM data collection disabled).• supervisionState (an indication that TCA generation has been enabled or disabled for a PM select list). This attribute is either AS_Unlocked (TCA generation enabled) or AS_Locked (TCA generation disabled).

For **type_name** = "NT_BACKUP_STATUS"

Holds details of a change to a managed element's backup status.

Name	Type	Description
"notificationId"	String	The uniqueness and the sequence of the notificationId are not guaranteed.
"emsTime"	globaldefs::Time_T	The time at which the event was reported by the EMS system.
"neTime"	globaldefs::Time_T	The time provided by the NE. In a case where the NE does not report time, this field is optional or an empty string can be reported.
"backupStatus"	softwareManager::BackupStatus_T	Backup Status of the Managed Element
"meName"	globaldefs::NamingAttributes_T	The name of the Managed Element from which the backup was taken

For **type_name** = "NT_HEARTBEAT"

Holds details of an EMS heartbeat notification.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of the notificationId are not guaranteed. If the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"objectName"	globaldefs::NamingAttributes_T	Identifies the EMS sending the heartbeat.
"objectType"	notifications::ObjectType_T	OT_EMS.
"emsTime"	globaldefs::Time_T	The time at which the heartbeat notification was generated by the EMS system.

For **type_name** = "NT_PULL_FILE_TRANSFER_STATUS"

Holds details of an File ready for PULL notification. When NMS require EMS do a long-time task, and EMS will report when data was ready and saved in file. The notification contains URL for file and status.

Name	Type	Description
"notificationId"	string	The uniqueness and the sequence of

Name	Type	Description
		the notificationId are not guaranteed. If ² the optional parameter "X.733::CorrelatedNotifications" is supported by the EMS, notification identifiers must be chosen to be unique across all notifications from a particular objectName throughout the time that correlation is significant.
"fileType"	string	The File type to be pull, it could be one of the following value: "NRM", "PM", "FM"
"ftpURL"	string	The full path for file in FTP server: include: FTP server IP, Port, and path
"userName"	string	FTP User name for NMS to pull file from EMS server.
"password"	string	FTP password for NMS to pull file from EMS server.
"taskName"	string	Unique task name for operation in EMS * The NMS need to ensure the uniqueness of the task name
"transferStatus"	FileTransferStatus_T	<i>Indicates file transfer status, and could</i>

Name	Type	Description
		<i>be on of the following value:</i> FT_IN_PROGRESS, FT_FAILED, FT_COMPLETED
"percentComplete"	short	Indicates percent of file transfer complete must be in the range 0..100. Only applies if transferStatus is FT_IN_PROGRESS or FT_COMPLETED; this field is optional otherwise.
"failureReason"	string	This string will contain the failure reason. Only applies if transferStatus is FT_FAILED; this field is optional otherwise.

remainder_of_body:

NULL for all events apart from "NT_OBJECT_CREATION".

For object creation notifications remainder_of_body is one of the following types depending on the objectType

objectType	Type carried in remainder_of_body
OT_AID	Not valid
OT_ASAP	aSAP::ASAP_T
OT_CALL	callSNC::Call_T
OT_CONNECTION	subnetworkConnection::SubnetworkConnection_T
OT_CONNECTION_TERMINATION_POINT	Not valid
OT_EMS	Not valid
OT_EPROTECTION_GROUP	Protection ::EprotectionGroup_T
OT_EQUIPMENT	equipment::Equipment_T
OT_EQUIPMENT_HOLDER	equipment::EquipmentHolder_T
OT_FLOW_DOMAIN	flowDomain:: FlowDomain_T
OT_FLOW_DOMAIN_FRAGMENT	flowDomainFragment:: FlowDomainFragment_T
OT_GTP	terminationPoint::GTP_T
OT_MANAGED_ELEMENT	managedElement::ManagedElement_T
OT_MATRIX_FLOW_DOMAIN	flowDomain:: MatrixFlowDomain_T
OT_MULTILAYER_ROUTING_AREA	multiLayerSubnetwork::MultiLayerSubnetwork_T
OT_MULTILAYER_SNPP	mLSNPP:: MultiLayerSNPP_T

objectType	Type carried in remainder_of_body
OT_MULTILAYER_SNPPLINK	mLSNPPLink:: MultiLayerSNPPLink_T
OT_MULTILAYER_SUBNETWORK	multiLayerSubnetwork::MultiLayerSubnetwork_T
OT_PHYSICAL_TERMINATION_POINT	terminationPoint::TerminationPoint_T
OT_PMP	performance::PMP_T
OT_PROTECTION_GROUP	protection::ProtectionGroup_T
OT_SUBNETWORK_CONNECTION	subnetworkConnection::SubnetworkConnection_T
OT_TCA_PARAMETER_PROFILE	performance::TCAPparameterProfile_T
OT_TERMINATION_POINT_POOL	terminationPoint::TerminationPoint_T
OT_TOPOLOGICAL_LINK	topologicalLink::TopologicalLink_T
OT_TRAFFIC_CONDITIONING_PROFILE	trafficConditioningProfile:: TCPProfile_T
OT_TRAFFIC_DESCRIPTOR	trafficDescriptor::TrafficDescriptor_T
OT_TRANSMISSION_DESCRIPTOR	transmissionDescriptor::TransmissionDescriptor_T

Using the Telecom Log Service

Note that the Telecom Log Service is an optional feature of the MTNM interface.

Basic Concepts of the OMG Telecom Log Service

This interface uses the log service specification identified by the OMG as

formal/02-11-12: Telecom Log Service, version 1.1

Date: November 2002

Version 1.1

This is accessible at http://www.omg.org/technology/documents/formal/telecom_log_service.htm.

Overview

The OMG Telecom Log Service provides for the logging and/or forwarding of events generated by applications. It supports the functionality specified in ITU-T Recommendation X.735, *Information Technology - Open systems Interconnection Systems Management : Log Control Function*. The OMG Telecom log service extends the OMG Notification service, which in turn extends the OMG Event service.

The OMG Telecom Log service specification defines several types of logs:

- The Log interface serves as an abstract interface from which all other log interfaces inherit. The Log interface has attributes and operations common to all other log interfaces.
- The EventLog interface inherits from the Log and CosEventChannelAdmin::EventChannel interfaces. This interface supports logging and the forwarding of events. There is no support for filtering of event notifications, although there is support for log filtering.
- The NotifyLog interface inherits from the EventLog and CosNotifyChannelAdmin::EventChannel interfaces.

The inheritance relationships among the various log interfaces are shown in Figure 3.

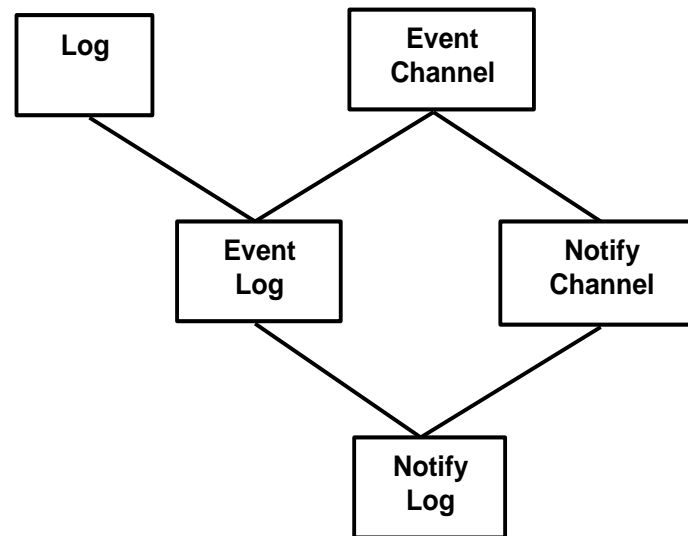


Figure 3. Log Inheritance

An illustration of a NotifyLog is shown in Figure 4. The NotifyLog provides all the functions of a notification channel, **and** has the ability to log events.

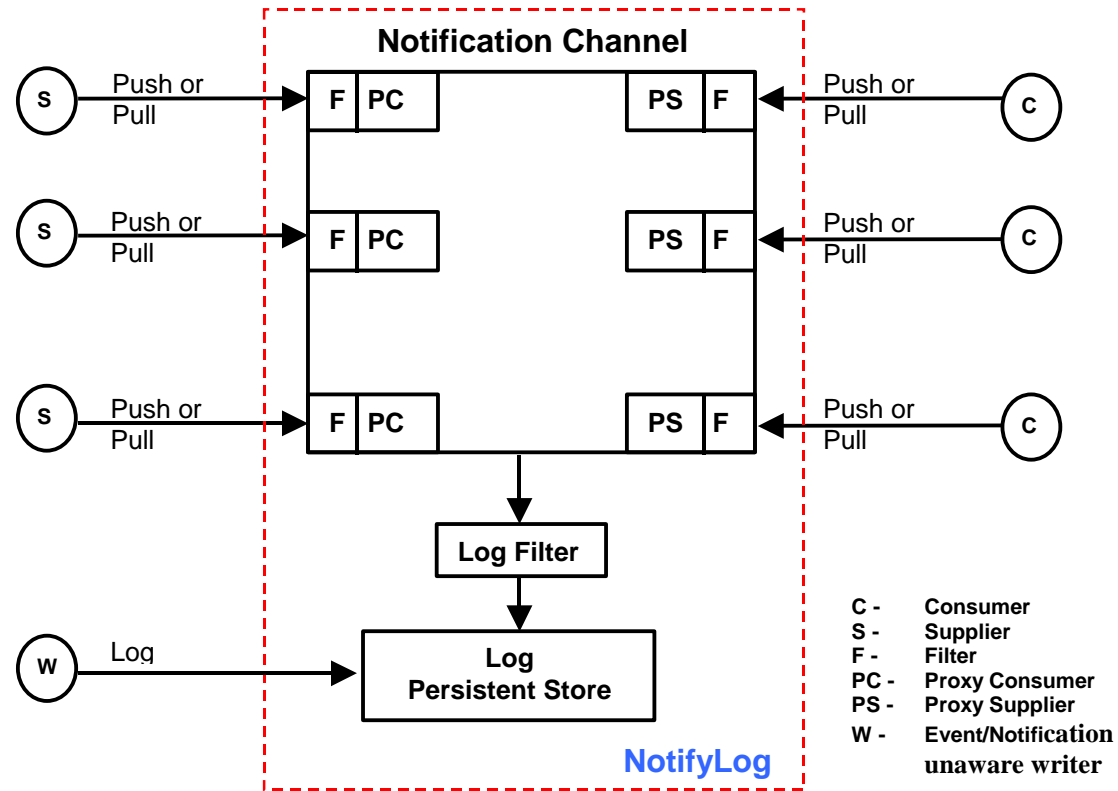


Figure 4. Diagram of a NotifyLog

The OMG Telecom Log Service has the following characteristics:

- support for push and pull interfaces,
- support for multiple suppliers and consumers,

- filtering of events before logging,
- storage of events in Logs,
- forwarding of events (similar to the notification service) to applications or other Logs,
- forwarding of log-related events, i.e., log creation, log deletion, log capacity threshold alarm, log attribute value change, log state change, and log processing alarm,
- the filtering is based on the same default constraint language as specified in the notification service.

The Log interface supports

- the monitoring (i.e., getting) and setting of the Administrative State, Availability Status, Maximum Log Size, Log Full Action, Log Duration, Log Scheduling, Log Capacity Alarm Threshold, Log Record Compaction, and Quality of Service (the flush operation, if supported)
- the monitoring (not setting) of the Operational State, Current Log Size and Quality of Service,
- LogRecord management including the storage, modification, and deletion of log records, and the query and retrieval of log records
- the ability to associate a Filter object with a Log.

Recommendations for Usage of the Telecom Log Service

- 1) Usage of the OMG Telecom Log Service in conjunction with the MTNM interface is optional.
- 2) Since the OMG Telecom Log Service provides a superset of the capabilities of the OMG Notification Service, the log service may be used in lieu of the notification service.
- 3) When using the log service in lieu of the notification service, it is recommended that the NotifyLog interface be used. The NotifyLog interface inherits from the EventLog and the NotifyChannel interfaces.
- 4) Concerning the Log interface, it is recommended that the NMS have access to all capabilities, except the ability directly store, modify and delete log records.

- 5) The log service provides log factory interfaces that allow clients to request the creation of log channels. It is recommended that the log factory interfaces not be accessible by the NMS.
- 6) Setting of log filters by the NMS is implementation dependent. Allowing the NMS to set filters may could problems in cases where multiple NMSs access the same EMS.
- 7) The various log events and alarms, e.g., LogObjectCreation, LogCapacityThresholdAlarm, are defined as untyped events (and alarms) in the OMG Telecom Log Service. As noted in the OMG Telecom Log Service specification, the consumer of log events (in this case, the NMS) may filter the untyped events and alarms via the CosEventChannelAdmin::ConsumerAdmin interface. The OMG defines three types of events: untyped, typed (now deprecated) and structured. The MTNM specification typically uses structured events. However, since there does not appear to be a standard automated way of mapping the untyped log events and alarms to structured events, it is recommended that untyped events be delivered to the NMS unchanged.
- 8) As defined in the OMG Telecom Log Service specification, a log record has the following structure:

```
struct LogRecord {  
    RecordId id;  
    TimeT    time;  
    NVList   attr_list; // attributes, optional  
    any      info; };
```

The info field is used to store the actual log record. Since the MTNM interface is making use of the structured event for notifications, the info field would actually contain a structured event. It is recommended that the optional attributes field not be used.

Revision History

Version	Date	Description of Change
3.0	November 2006	Conversion of iterators into new template
3.1	March 2007	Add definitions for the following new object types: OT_CALL OT_CONNECTION OT_FLOW_DOMAIN OT_FLOW_DOMAIN_FRAGMENT OT_MATRIX_FLOW_DOMAIN OT_MULTILAYER_ROUTING_AREA OT_MULTILAYER_SNPP OT_MULTILAYER_SNPPLINK OT_TRAFFIC_CONDITIONING_PROFILE
3.2	April 2014	Added definition for new notification type: "NT_PULL_FILE_TRANSFER_STATUS"

Acknowledgements

<FirstName> <LastName> <Company>

How to comment on the document

Comments and requests for information must be in written form and addressed to the contact identified below:

Keith Dorking Ciena
Phone: +1 678 867 5007
Fax: +1 678 867 5010

e-mail: kdorking@ciena.com

Please be specific, since your comments will be dealt with by the team evaluating numerous inputs and trying to produce a single text. Thus we appreciate significant specific input. We are looking for more input than wordsmith” items, however editing and structural help are greatly appreciated where better clarity is the result.