TSGS#26(04)0784

Source: SA5 (Telecom Management)

Title: 3 Rel-6 CR 32.412/3 Performance Management (PM) IRP Information Service (IS) / CORBA SS

Document for: Approval

Agenda Item: 7.5.3

Doc1stevel	Specific a	CR	R	Phase	Subject	Ca	VersCu	Doc2ndLev	WorkitemsI D
SP-040784	32.412	007	--	Rel-6	Correct ambiguous precondition statement related to createThresholdMonitor operation	F	6.2.0	S5-046977	OAM-PM
SP-040784	32.412	008	--	Rel-6	Correct definition of ObjectClass and ObjectInstance in “notifyMeasurementJobStatusChanged” and “notifyThresholdMonitorStatusChanged”	F	6.2.0	S5-047128	OAM-PM
SP-040784	32.413	004	--	Rel-6	Add missing exception & filter to PM IRP CORBA SS, Align with IDL Style Guide in 32.150	F	6.2.0	S5-047116	OAM-PM

CR-Form-v7

CHANGE REQUEST

⌘ 32.412 CR 007 ⌘rev - ⌘ Current version: 6.2.0 ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network X Core Network X

Title:	⌘	Correct ambiguous precondition statement related to createThresholdMonitor operation
Source:	⌘	**SA5** (mikael.rutanen@nokia.com)
Work item code:	⌘	OAM-PM
Date:	⌘	01/10/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)

Reason for change:	⌘	CreateThresholdMonitor operation is allowed in two cases: 1) the IRPManager shall first start a MeasurementJob to monitor the measurementType(s). Then a ThresholdMonitor for these measurementType(s) can be created 2) a ThresholdMonitor can be accepted for measurementType(s) regardless of existing MeasurementJob(s). The dependency between ThresholdMonitor and MeasurementJob(s) after successful createThresholdMonitor operation is unclear in option 2)
Summary of change:	⌘	The clarification of independency between ThresholdMonitor and MeasurementJob added to clause 7.4.1.1. Creating a ThresholdMonitor without existing MeasurementJob for target measurementType(s) does not create a MeasurementJob visible to IRPManager.
Consequences if not approved:	⌘	The behaviour of IRPAgent remains ambiguous.

Clauses affected:	⌘	7.4.1.1

		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	

Other comments:	⌘	

Change in Clause 7.4.1.1

7.4.1.1 Definition

This operation supports `IRPManager`'s request to create a `ThresholdMonitor` that defines the thresholds for some specific measurementTypes. If the threshold defined is (a) crossed or (b) reached, the related performance alarms will be emitted to subscribed `IRPManager(s)`.

Two cases are allowed:

- One case only accepts threshold monitoring of measurementType(s) that are already under monitoring by an existing `MeasurementJob`. This kind of `PMIRP` will not monitor a measurementType for (a) threshold-crossing or (b) threshold reaching or for clearing if that measurementType is not already subject to a `MeasurementJob` monitoring. This kind of `PMIRP` will only determine the (a) threshold crossing or (b) threshold reaching and clearing events when the related `MeasurementJob(s)` are in Active states. The `IRPManager`, when interacting with this kind of `PMIRP`, must first start a `MeasurementJob` to monitor the measurementTypes and then invoke this operation for the same measurementTypes.

- The other case is that it can accept threshold monitoring of measurementType(s) regardless if they are already under monitoring by existing `MeasurementJob(s)`. `ThresholdMonitor` is independent of any existing `MeasurementJob`(s) and no new `MeasurementJob`(s) are created for threshold monitoring purposes.

(See 6.3.8.1 for clarification of the alternatives (a) and (b))

End of Change in Clause 7.4.1.1
End of Document

CR-Form-v7

CHANGE REQUEST

⌘	32.412	CR	008	⌘ rev	-	⌘	Current version:	6.2.0	⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: | UICC apps⌘ ☐ ME ☐ Radio Access Network X Core Network X

Title:	⌘	Correct definition of ObjectClass and ObjectInstance in "notifyMeasurementJobStatusChanged" and "notifyThresholdMonitorStatusChanged"
Source:	⌘	**SA5** (Ericsson, ulf.hubinette@ericsson.com; Huawei, afi@huawei.com)
Work item code:	⌘	OAM-PM **Date:** ⌘ 19/11/2004
Category:	⌘	**F** **Release:** ⌘ Rel-6

Use one of the following categories:
- ***F*** *(correction)*
- ***A*** *(corresponds to a correction in an earlier release)*
- ***B*** *(addition of feature),*
- ***C*** *(functional modification of feature)*
- ***D*** *(editorial modification)*

Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	ObjectClass and ObjectInstance together carry the same info as the SystemDN attribute, and are therefore redundant as they both are specified as "M".
Summary of change:	⌘	Change the matching information of ObjectClass and ObjectInstance in the notifications "notifyMeasurementJobStatusChanged" and "notifyThresholdMonitorStatusChanged.
Consequences if not approved:	⌘	Compliant systems are forced to carry redundant information for the notifications and DN of these notifications' issuer will be absent.

Clauses affected:	⌘	7.6.1 and 7.6.2

Other specs affected: ⌘	Y	N	
		X	Other core specifications
		X	Test specifications
		X	O&M Specifications

⌘

Other comments: ⌘

Change in Clause 7.6

7.6 PMIRPNotification_1 Interface (M)

7.6.1 notifyMeasurementJobStatusChanged (M)

7.6.1.1 Definition

The PMIRP Agent notifies all subscribed IRPManagers about the status changes of a MeasurementJob. The status changes in that case include Suspended=>Scheduled, Active=>Suspended, Scheduled=>Suspended, Suspended=>Active, Scheduled=>Active, Active=>Stopped, Suspended=>Stopped, Scheduled=>Stopped.

7.6.1.2 Input parameters

Parameter Name	Qualifier	Matching Information	Comment
objectClass	M	PMIRP.objectClass	This parameter and objectInstance together carry the same semantics of IRPAgent.systemDN. Notification header - see 3GPP TS 32.302 [12].
objectInstance	M	PMIRP.objectInstance	This parameter and objectClass together carry the same semantics of IRPAgent.systemDN. Notification header - see 3GPP TS 32.302 [12].
notificationId	M	--	Notification header - see 3GPP TS 32.302 [12].
eventTime	M	--	Notification header - see 3GPP TS 32.302 [12].
notificationType	M	"notifyMeasurementJobStatusChanged".	Notification header - see 3GPP TS 32.302 [12].
systemDN	C	IRPAgent.systemDN.	It carries the DN of the IRPAgent that emits this notification. Notification header - see 3GPP TS 32.302 [12]
jobId	M	MeasurementJob.jobId	
jobStatus	M	MeasurementJob.jobStatus	The new status of the MeasurementJob.
reason	O	String	It carries one or several of the assertion names of the From-state of Triggering Event.

7.6.1.3 Triggering Event

7.6.1.3.1 From-state

failToReadMeasurementTypesForExtendedProlongPeriod OR internalProblem OR stopMeasurementJob OR stopTimeReached OR resumeMeasurementJob OR suspendMeasurementJob OR startTimeReached OR suspendMeasurementJobBySystem

Assertion Name	Definition
failToRead MeasurementTypesForExtendedProlongPeriod	Because the PMIRP Agent have failed to read the monitored measurementType (s) from managed resources for one or more times, the PMIRP Agent decides that it will not try to read in the future and place the MeasurementJob in "Stopped" state.
internalProblem	Because of an unspecified internal problem, PMIRP Agent decides that it no longer can maintain the MeasurementJob in any jobStatus but "Stopped".
stopMeasurementJob	The stopMeasurementJob returns success.
stopTimeReached	The stop time for MeasurementJob has been reached.
resumeMeasurementJob	The resumeMeasurementJob returns success.
suspendMeasurementJob	The suspendMeasurementJob returns success.
startTimeReached	The start time for MeasurementJob has been reached.
suspendMeasurementJobBySystem	The MeasurementJob has been suspended by the system in case of overload

7.6.1.3.2 To-state

measurementJobStatusChanged

Assertion Name	Definition
measurementJobStatusChanged	The `MeasurementJob.jobStatus` changed to a new value.

7.6.2 notifyThresholdMonitorStatusChanged (O)

7.6.2.1 Definition

The `PMIRP` Agent notifies all subscribed `IRPManagers` about the status changes of a `ThresholdMonitor`. The status changes in that case include Suspended=>Active, Active=>Suspended.

NOTE: The notifyThresholdMonitorStatusChanged notification is mandatory if PMIRPOperations_2 is supported.

7.6.2.2 Input Parameters

Parameter Name	Qualifier	Matching Information	Comment
objectClass	M	PMIRP.objectClass--	This parameter and objectInstance together carry the same semantics of IRPAgent.systemDN. Notification header - see 3GPP TS 32.302 [12]
objectInstance	M	PMIRP.objectInstance--	This parameter and objectClass together carry the same semantics of IRPAgent.systemDN. Notification header - see 3GPP TS 32.302 [12]
notificationId	M	--	Notification header - see 3GPP TS 32.302 [12]
eventTime	M	--	Notification header - see 3GPP TS 32.302 [12]
notificationType	M	"notifyThresholdMonitorStatusChanged".	Notification header - see 3GPP TS 32.302 [12]
systemDN	C	`IRPAgent.systemDN`.	It carries the DN of the `IRPAgent` that emits this notification. Notification header - see 3GPP TS 32.302 [12]
monitorId	M	`ThresholdMonitor.monitorId`	
monitorStatus	M	`ThresholdMonitor.thresholdMonitorStatus`	The new status of the `ThresholdMonitor.`
reason	O	String	It carries one or several of the assertion names of the From-state of Triggering Event.

7.6.2.3 Triggering Event

7.6.2.3.1 From-state

`resumeThresholdMonitor` OR `suspendThresholdMonitor`

Assertion Name	Definition
resumeThresholdMonitor	The resumeThresholdMonitor returns success.
suspendThresholdMonitor	The suspendThresholdMonitor returns success.

7.6.2.3.2 To-state

thresholdMonitorStatusChanged

Assertion Name	Definition
thresholdMonitorStatusChanged	The `ThresholdMonitor.thresholdMonitorStatus` changed to a new value.

End of Change in Clause 7.6
End of document

3GPP TSG-SA5 (Telecom Management) **S5-047116**
Meeting #40, Sanya, CHINA, 15 - 19 November 2004

CR-Form-v7

CHANGE REQUEST

⌘ **32.413** **CR** **004** ⌘ **rev** **-** ⌘ Current version: **6.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☒ Core Network ☒

Title:	⌘	Add missing exception & filter to PM IRP CORBA SS, Align with IDL Style Guide in 32.150
Source:	⌘	**SA5** (edwin.tse@ericsson.com)
Work item code:	⌘	OAM-PM
Date:	⌘	19/11/2004
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Incorrect mapping of IS-defined non-filterable parameters to SS-defined non-filterable fields (instead of filterable fields). Missing mandatory exceptions for optional methods. Implicitly defined time slot specification (reuse from OMG definition). Non-explicit high workload exception reasons for create_measurement_job method. Mis-aligned IDL style with IDL Style Guide.
Summary of change:	⌘	Correct mapping of IS-defined non-filterable parameters to SS-defined non-filterable fields (instead of filterable fields). Add mandatory exceptions for optional methods. Explicitly define time slot specification (reuse from OMG definition). Explicitly define high workload exception reasons for create_measurement_job method. Align IDL style with IDL Style Guide.
Consequences if not approved:	⌘	Filtering capability is not aligned with IS. IRPAgent wastes CPU cycles on non-filterable parameters. PMIRP that does not support the optional methods have no standard way of responding. IRPManager, on reception of non-standard response from PMIRP, can be confused and cannot conclude if the method is supported or not. Definition of time slot structure will be in IDL comments rather than in IDL explicitly defined structure. IRPManager, who receives exception for create_measurement_job in a multi-vendor environment can receive different strings denoting the same standard-defined-exceptions.

Clauses affected:	⌘	5.2, 5.3, A.1, A.2, A.3

		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	

Other comments:	⌘	

Change in Clause 5.2

5.2 Operation parameter mapping

The PMIRP: IS 3GPP TS 32.412 [7] defines semantics of parameters carried in operations across the PMIRP. The following tables indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table 2: Mapping from IS `createMeasurementJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
iocName	PMIRPConstDefs::MOClassNameType moClass	M
iocInstanceList	PMIRPConstDefs::MOInstanceListType moInstanceList	M
measurementCategoryList	PMIRPConstDefs::MeasurementCategoryListType measurementCategoryList	M
granularityPeriod	PMIRPConstDefs::GranularityPeriodType granularityPeriod	M
reportingPeriod	PMIRPConstDefs::ReportingPeriodType reportingPeriod	M
startTime	PMIRPConstDefs::~~StartTimeTypeOpt~~IRPTimeTypeOpt startTime	O
stopTime	PMIRPConstDefs::~~StopTimeTypeOpt~~IRPTimeTypeOpt stopTime	O
schedule	PMIRPConstDefs::ScheduleTypeOpt schedule	O
jobId	Return value of type PMIRPConstDefs::JobIdType	M
unsupportedList	PMIRPConstDefs::~~JUnsu~~JobUnsupportedListType unsupportedList	M
priority	PMIRPConstDefs::JobPriorityTypeOpt priority	O
status	ManagedGenericIRPConstDefs::Signal Exception: CreateMeasurementJob, ManagedGenericIRPSystem::InvalidParameter, ManagedGenericIRPSystem::ParameterNotSupported, HighWorkLoad	M

Table 3: Mapping from IS `stopMeasurementJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
jobId	PMIRPConstDefs::JobIdType jobId	M
status	PMIRPConstDefs::ResultType Exception: StopMeasurementJob, UnknownJob, JobCannotBeStopped	M

Table 4: Mapping from IS `suspendMeasurementJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
jobId	PMIRPConstDefs::JobIdType jobId	M
status	PMIRPConstDefs::ResultType Exception: SuspendMeasurementJob, UnknownJob, JobAlreadySuspended, ManagedGenericIRPSystem::OperationNotSupported	M

Table 5: Mapping from IS `resumeMeasurementJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
jobId	PMIRPConstDefs::JobIdType jobId	M
status	PMIRPConstDefs::ResultType Exception: ResumeMeasurementJob, UnknownJob, JobIsNotSuspended, HighWorkLoad, ManagedGenericIRPSystem::OperationNotSupported	M

Table 6: Mapping from IS `listMeasurementJobs` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
jobIdList	PMIRPConstDefs::JobListIdType jobListId	M
jobInfoList	Return value of type PMIRPConstDefs::JobInfoListType	M
status	PMIRPConstDefs::ResultType Exception: ListMeasurementJobs, ManagedGenericIRPSystem::InvalidParameter	M

Table 7: Mapping from IS `createThresholdMonitor` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
iocName	PMIRPConstDefs::MOClassNameType moClass	M
iocInstanceList	PMIRPConstDefs::MOInstanceListType moInstanceList	M
thresholdInfoList	PMIRPConstDefs::ThresholdInfoListType thresholdInfoList	M
monitorGranularityPeriod	PMIRPConstDefs::MonitorGranularityPeriodType monitorGranularityPeriod	M
monitorId	Return value of type PMIRPConstDefs::MonitorIdType	M
unsupportedList	PMIRPConstDefs::~~MUnsupported~~MonitorUnsupportedListType unsupportedList	M
status	ManagedGenericIRPConstDefs::Signal Exception: CreateThresholdMonitor, ManagedGenericIRPSystem::InvalidParameter, ManagedGenericIRPSystem::OperationNotSupported	M

Table 8: Mapping from IS `deleteThresholdMonitor` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
monitorId	PMIRPConstDefs::MonitorIdType monitorId	M
status	PMIRPConstDefs::ResultType Exception: DeleteThresholdMonitor, UnknownThresholdMonitor, ManagedGenericIRPSystem::OperationNotSupported	M

Table 9: Mapping from IS `listThresholdMonitors` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
monitorIdList	PMIRPConstDefs::MonitorIdListType monitorIdList	M
monitorInfoList	Return value of PMIRPConstDefs::MonitorInfoListType	M
status	PMIRPConstDefs::ResultType Exception: ListThresholdMonitors, ManagedGenericIRPSystem::InvalidParameter, ManagedGenericIRPSystem::OperationNotSupported	M

Table 10: Mapping from IS `suspendThresholdMonitor` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
monitorId	PMIRPConstDefs::MonitorIdType monitorId	M
status	PMIRPConstDefs::ResultType Exception: SuspendThresholdMonitor, UnknownThresholdMonitor, ThresholdMonitorAlreadySuspended, ManagedGenericIRPSystem::OperationNotSupported	M

Table 11: Mapping from IS `resumeThresholdMonitor` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
monitorId	PMIRPConstDefs::MonitorIdType monitorId	M
status	PMIRPConstDefs::ResultType Exception: ResumeThresholdMonitor, UnknownThresholdMonitor, ThresholdMonitorIsNotSuspended, ManagedGenericIRPSystem::OperationNotSupported	M

Table 12: Mapping from IS `getIRPVersion` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
versionNumberSet	Return value of type ManagedGenericIRPConstDefs::VersionNumberSet	M
status	Exception: GetPMIRPVersions	M

…

End of change in Clause 5.2

Change in Clause 5.3

5.3 Notification parameter mapping

The PMIRP: IS 3GPP TS 32.412 [7] defines semantics of parameters carried in notifications. The following table indicates the mapping of these parameters to their OMG CORBA Structured Event (defined in OMG Notification Service [9]) equivalents. The composition of OMG Structured Event, as defined in the OMG Notification Service [9], 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 PMIRP: IS 3GPP TS 32.412 [7] defined notification parameters.

Table 5.3.1: Mapping for `notifyMeasurementJobStatusChanged`

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 subclause 3.1. It indicates the syntax and semantics of the Structured Event as defined by the present document.
notificationType	t~~T~~ype_name	M	This is the ~~ET_MEASUREMENT_JOB_STATUS_CHANGED of module of PMIRPNotifDefs.~~ "notifyMeasurementJobStatusChanged".
There is no corresponding IS attribute	event_name	M	It carries no information.
There is no corresponding IS 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 this NV pair is the MANAGED_OBJECT_INSTANCE of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
notificationId	One NV pair of remaining_body ~~filterable_ body_fields~~	M	Name of NV pair is the NOTIFICATION_ID of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a long. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
eventTime	One NV pair of filterable_ body_fields	M	Name of NV pair is the EVENT_TIME of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
systemDN	One NV pair of filterable_ body_fields	M	Name of NV pair is the SYSTEM_DN of interface AttributeNameValue of module NotificationIRPConstDefs.

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
			Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
jobId	One NV pair of filterable_ body_fields	M	Name of NV pair is the JOB_ID of ~~module~~ PMIRPNotifDefs::notifyMeasurementJobStatusChanged Value of NV pair is JobIdType of module PMIRPConstDefs.
jobStatus	One NV pair of remaining_body ~~filterable_ body_fields~~	M	Name of NV pair is the JOB_STATUS of ~~module~~ PMIRPNotifDefs::notifyMeasurementJobStatusChanged Value of NV pair is JobStatusType of module PMIRPConstDefs.
reason	One NV pair of remaining_body ~~filterable_ body_fields~~	O	Name of NV pair is the REASON of ~~module~~ PMIRPNotifDefs::notifyMeasurementJobStatusChanged Value of NV pair is a string.
~~There is no corresponding IS attribute.~~	~~remaining_body~~		

Table 5.3.2: Mapping for `notifyThresholdMonitorStatusChanged`

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 subclause 3.1. It indicates the syntax and semantics of the Structured Event as defined by the present document.
notificationType	t~~T~~ype_name	M	This is the ~~ET_THRESHOLD_MONITOR_STATUS_CHANGED of module of PMIRPNotifDefs.~~ "notifyThresholdMonitorStatusChanged".
There is no corresponding IS attribute	event_name	M	It carries no information.
There is no corresponding IS 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 this NV pair is the MANAGED_OBJECT_INSTANCE of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
notificationId	One NV pair of remaining_body ~~filterable_ body_fields~~	M	Name of NV pair is the NOTIFICATION_ID of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a long. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
eventTime	One NV pair of filterable_ body_fields	M	Name of NV pair is the EVENT_TIME of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
systemDN	One NV pair of filterable_ body_fields	M	Name of NV pair is the SYSTEM_DN of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [5]).
monitorId	One NV pair of filterable_ body_fields	M	Name of NV pair is the MONITOR_ID of ~~module~~ PMIRPNotifDefs::notifyThresholdMonitorStatusChanged Value of NV pair is MonitorIdType of module PMIRPConstDefs.

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
monitorStatus	One NV pair of remaining_body ~~filterable_body_fields~~	M	Name of NV pair is the MONITOR _STATUS of ~~module~~ PMIRPNotifDefs::notifyThresholdMonitorStatusChanged Value of NV pair is MonitorStatusType of module PMIRPConstDefs.
reason	One NV pair of remaining_body ~~filterable_body_fields~~	O	Name of NV pair is the REASON of ~~module~~ PMIRPNotifDefs::notifyThresholdMonitorStatusChanged Value of NV pair is a string.
~~There is no corresponding IS attribute.~~	~~remaining_body~~		

End of change in Clause 5.3

Change in Clause A.1, A.2, A.3

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

```
// File: PMIRPConstDefs.idl

#ifndef _PMIRPCONSTDEFS_IDL_PMIRPConstDefs_idl
#define _PMIRPCONSTDEFS_IDL_PMIRPConstDefs_idl

#include "TimeBase.idl"

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

/* ## Module: PMIRPConstDefs
This module contains commonly used definitions for PM IRP
================================================================
*/
module PMIRPConstDefs
{

   enum ResultType {OK, Failure};

   typedef   string   MOClassNameType;
   typedef   string   MOInstanceType;
   typedef   sequence<MOInstanceType>   MOInstanceListType;
   typedef   string   MeasurementCategoryType;
   typedef   sequence<MeasurementCategoryType>  MeasurementCategoryListType;
   typedef   unsigned long  GranularityPeriodType; //The unit is minute.
   typedef   unsigned long  ReportingPeriodType; //The unit is minute.
   typedef   TimeBase::UtcT UTCTimeType;

   union IRPStartTimeTypeOpt switch(boolean)
   {
     case TRUE: UTCTimeType value;
   };

   union StopTimeTypeOpt switch(boolean)
   {
     case TRUE: UTCTimeType value;
   };

   struct Time24
   {
     unsigned short hour; // 0-23
     unsigned short minute; // 0-59
   };


   typedef string IntervalTimeType;
   // the IntervalTimeType with format ab:cd (from 00:00 to 23:59 of one day)
   struct IntervalOfDayType
   {
     Time24IntervalTimeType intervalStartTime;
     Time24IntervalTimeType intervalStopTime;
   };
   typedef sequence<IntervalOfDayType> DailySchedulingType;

   const short Sunday = 1;
   const short Monday = 2;
   const short Tuesday = 4;
   const short Wednesday = 8;
   const short Thursday = 16;
   const short Friday = 32;
   const short Saturday = 64;

   typedef string DayOfWeekType;
```

```
    //The value of DayOfWeekType is a BIT STRING (SIZE(7))
    //{sunday(0),monday(1),tuesday(2),wednesday(3),
    //thursday(4), friday(5), saturday(6)}
    typedef shortsequence<DayOfWeekType> DaysOfWeekType;
    // Bit mask of week days,
    // e.g. “Sunday(1) and Wednesday(8)” is encoded as 9.

    struct WeeklySchedulingElement
    {
      DaysOfWeekType daysOfWeek;
      DailySchedulingType intervalsOfDay;
    };
    typedef sequence<WeeklySchedulingElement> WeeklySchedulingType;
    enum scheduleTypeChoice { Daily, Weekly };
    union ScheduleType switch (scheduleTypeChoice)
    {
      case Daily:  DailySchedulingType dailyScheduling;
      case Weekly: WeeklySchedulingType weeklyScheduling;
    };
    union ScheduleTypeOpt switch(boolean)
    {
      case TRUE: ScheduleType value;
    };


    typedef   unsigned long   JobIdType;
    typedef sequence<JobIdType> JobIdListType;
    struct JUnsupportedType
    {
      MOInstanceType  moInstance;
      MeasurementCategoryType measurementCategory;
      string reason;
    };
    typedef  sequence<JUnsupportedType>  JUnsupportedListType;


    /**
     * Defines the name of an attribute of a Managed Object
     */
    typedef string MOAttributeName;

    enum  JobStatusType {Scheduled, Active, Suspended, Stopped};
    enum  JobPriorityType {Low, Medium, High};
    union JobPriorityTypeOpt switch (boolean)
    {
       case TRUE: JobPriorityType value;
    };

    struct JobInfoType
    {
     JobIdType  jobId;
     MOClassNameType  moClass;
     MOInstanceListType  moInstanceList;
     MeasurementCategoryListType  measurementCategoryList;
     GranularityPeriodType  granularityPeriod;
     ReportingPeriodType  reportingPeriod;
     StartTimeTypeOptIRPTimeTypeOpt  startTime;
     StopTimeTypeOptIRPTimeTypeOpt  stopTime;
     ScheduleTypeOpt  schedule;
     JobStatusType jobStatus;
     JobPriorityTypeOpt jobPriority;
    };
    typedef sequence<JobInfoType> JobInfoListType;

    typedef string MeasurementTypeNameType;
    typedef string SubCounterNameType;
    typedef shortstring ProbableCauseType; //THRESHOLD_CROSSED = 351;

    typedef string SpecificProblemType;
    typedef any ThresholdValueType;
    enum SeverityType {Warning, Minor, Major, Critical};
    union HysteresisType switch(boolean)
    {
      case TRUE:  long longValue;
      case FALSE: float floatValue;
    };
    enum DirectionType { Increasing, Decreasing};
```

```
    struct ThresholdPackElementType
    {
      ThresholdValueType thresholdValue;
      SeverityType severity; // the value shall be
      // one of Warning, Minor, Major or Critical.
      HysteresisType hysteresis;
    };
    typedef sequence<ThresholdPackElementType> ThresholdPackType;
    struct ThresholdInfoType
    {
      MeasurementTypeNameType measurementTypeName;
      SubCounterNameType subCounterName;
      ProbableCauseType probableCause;
      SpecificProblemType specificProblem;
      DirectionType direction;
      ThresholdPackType thresholdPack;
    };
    typedef sequence<ThresholdInfoType> ThresholdInfoListType;
    typedef GranularityPeriodType MonitorGranularityPeriodType;// time period is based on 5 minutes.
    typedef unsigned long MonitorIdType;
    struct MUnsupportedType
    {
      MOInstanceType  moInstance;
      MeasurementTypeNameType measurementTypeName;
      SubCounterNameType subCounterName;
      string reason;
    };
    typedef  sequence<MUnsupportedType>  MUnsupportedListType;
    enum  MonitorStatusType {MSuspended, MActive};

    typedef sequence<MonitorIdType> MonitorIdListType;
    typedef string EventTypeType; // The value is "Quality of Service Alarm"
    struct MonitorInfoType
    {
     MonitorIdType  monitorId;
     MOClassNameType  moClass;
     MOInstanceListType  moInstanceList;
     MonitorGranularityPeriodType monitorGranularityPeriod;
     ThresholdInfoListType thresholdInfoList;
     MonitorStatusType thresholdMonitorStatus;
     EventTypeType eventType;
    };
    typedef sequence<MonitorInfoType> MonitorInfoListType;


    /**
    * This block identifies attributes which are included as part of the
    * PMIRP. These attribute values should not
    * clash with those defined for the attributes of notification
    * header (see IDL of Notification IRP).
    */
    interface AttributeNameValue
    {
     const string JOB_ID = "JOB_ID";
     const string JOB_STATUS = "JOB_STATUS";
     const string REASON = "REASON";
     const string MONITOR_ID = "MONITOR_ID";
     const string MONITOR_STATUS = "MONITOR_STATUS";

     const string MONITOR_GRANULARITYPERIOD = "MONITOR_GRANULARITYPERIOD";
     const string THRESHOLD_INFO_LIST = "THRESHOLD_INFO_LIST";
    };

};

#endif _PMIRPCONSTDEFS_IDL_
```

A.2 IDL specification (file name "PMIRPSystem.idl")

```
//File: PMIRPSystem.idl

#ifndef _PMIRPSYSTEM_IDL_PMIRPSystem_idl
#define _PMIRPSYSTEM_IDL_PMIRPSystem_idl
```

```
#include "ManagedGenericIRPSystem.idl"
#include "ManagedGenericIRPConstDefs.idl"
#include "PMIRPConstDefs.idl"

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

/* ## Module: PMIRPSystem
This module contains the specification of all operations of PM IRP Agent.
================================================================
*/
module PMIRPSystem
{

   /**
   * Tthe reason specifies whether EM or NE hais high workload. The value shall be one
   * of following: emCpuBusy; emHDShortage, emLowMemory, {neCpuBusy, neObjectInstList},
   * {neHDShortage neObjectInstList}, {neLowMemory, neObjectInstList}, maxJobReached,
   * otherReason.
   * In the case the reason is a tuple, the first element is the string such as
   * "NE_CPU_BUSY" followed by a comma, then followed by a sequence of DN where
   * each DN is separated by its adjacent DN, if any, by a colon.  The DN is formatted
   * as described in 32.300.
   */

   exception HighWorkLoad { string reason; };
   interface HighWorkLoadExceptionReason
   {
      const string EmCpuBusy = "EM_CPU_BUSY";
      const string EmHDShortage = "EM_HD_SHORTAGE";
      const string EmLowMemory = "EM_LOW_MEMORY";
      const string NeCpuBusy = "NE_CPU_BUSY";
      const string NeHDShortage = "NE_HD_SHORTAGE";
      const string NeLowMemory = "NE_LOW_MEMORY";
      const string MaxJobReached = "MAX_JOB_REACHED";
      const string OtherReason = "OTHER_REASON";
   };


   exception UnknownJob { string reason; };
   exception JobCannotBeStopped { string reason; };
   exception JobAlreadySuspended { string reason; };
   exception JobIsNotSuspended { string reason; };
   exception UnknownThresholdMonitor { string reason; };
   exception ThresholdMonitorAlreadySuspended { string reason; };
   exception ThresholdMonitorIsNotSuspended { string reason; };

   /**
   * System fails to complete the operation.  System can provide reason
   * to qualify the exception.  The semantics carried in reason
   * is outside the scope of this IRP.
   */
   exception GetPMIRPVersions { string reason; };
   exception GetPMIRPOperationsProfile { string reason; };
   exception GetPMIRPNotificationProfile { string reason; };
   exception CreateMeasurementJob { string reason; };
   exception StopMeasurementJob { string reason; };
   exception SuspendMeasurementJob { string reason; };
   exception ResumeMeasurementJob { string reason; };
   exception ListMeasurementJobs { string reason; };

   exception CreateThresholdMonitor { string reason; };
   exception DeleteThresholdMonitor { string reason; };
   exception ListThresholdMonitors { string reason; };
   exception SuspendThresholdMonitor { string reason; };
   exception ResumeThresholdMonitor { string reason; };

   interface PMIRP
   {

      readonly attribute string iRPId;

      /**
      * Return the list of all supported PM IRP versions.
      */
      ManagedGenericIRPConstDefs::VersionNumberSet get_PM_IRP_versions (
      )
```

```
raises (GetPMIRPVersions);


/**
* Return the list of all supported operations and their supported
* parameters for a specific PM IRP version.
*/
ManagedGenericIRPConstDefs::MethodList get_PM_IRP_operations_profile (
   in ManagedGenericIRPConstDefs::VersionNumber pm_irp_version
)
raises (GetPMIRPOperationsProfile,
        ManagedGenericIRPSystem::OperationNotSupported,
        ManagedGenericIRPSystem::InvalidParameter);

/**
* Return the list of all supported notifications and their supported
* parameters for a specific PM IRP version.
*/
ManagedGenericIRPConstDefs::MethodList get_PM_IRP_notification_profile
(
   in ManagedGenericIRPConstDefs::VersionNumber pm_irp_version
)
raises (GetPMIRPNotificationProfile,
        ManagedGenericIRPSystem::OperationNotSupported,
        ManagedGenericIRPSystem::InvalidParameter);

/**
* Request to create a MeasurementJob through Itf-N.
*/
ManagedGenericIRPConstDefs::Signal create_measurement_job (
  in PMIRPConstDefs::MOClassNameType  moClass,
  in PMIRPConstDefs::MOInstanceListType  moInstanceList,
  in PMIRPConstDefs::MeasurementCategoryListType  measurementCategoryList,
  in PMIRPConstDefs::GranularityPeriodType  granularityPeriod,
  in PMIRPConstDefs::ReportingPeriodType  reportingPeriod,
  in PMIRPConstDefs::StartTimeTypeOptIRPTimeTypeOpt  startTime,
  in PMIRPConstDefs::StopTimeTypeOptIRPTimeTypeOpt  stopTime,
  in PMIRPConstDefs::ScheduleTypeOpt  schedule,
  in PMIRPConstDefs::JobPriorityTypeOpt  priority,
  out PMIRPConstDefs::JobIdType jobId,
  out PMIRPConstDefs::JUnsupportedListType unsupportedList
)
raises (CreateMeasurementJob,
        ManagedGenericIRPSystem::InvalidParameter,
        ManagedGenericIRPSystem::ParameterNotSupported,
        HighWorkLoad);

/**
* Request to stop a MeasurementJob through Itf-N, after which,
* the MeasurementJob will still be visible via Itf-N. Whether
* the MeasurementJob is thoroughly removed immediately from
* the managed system is vendor specific.
*/
PMIRPConstDefs::ResultType stop_measurement_job (
  in PMIRPConstDefs::JobIdType jobId)
raises (StopMeasurementJob,
        UnknownJob,
        JobCannotBeStopped);

/**
* Request to suspend a MeasurementJob
*/
PMIRPConstDefs::ResultType suspend_measurement_job (
  in PMIRPConstDefs::JobIdType jobId)
raises (SuspendMeasurementJob,
        UnknownJob,
        JobAlreadySuspended,
        ManagedGenericIRPSystem::OperationNotSupported);

/**
* Request to resume a MeasurementJob
*/
PMIRPConstDefs::ResultType resume_measurement_job (
  in PMIRPConstDefs::JobIdType jobId)
raises (ResumeMeasurementJob,
        UnknownJob,
        JobIsNotSuspended,
```

```
                HighWorkLoad,
                ManagedGenericIRPSystem::OperationNotSupported);

      /**
       * Request to list the information of all or of specified
       * MeasurementJobs
       */
       PMIRPConstDefs::ResultType list_measurement_jobs (
         in PMIRPConstDefs::JobIdListType jobListId,
         out PMIRPConstDefs::JobInfoListType jobInfoList)
       raises (ListMeasurementJobs,
               ManagedGenericIRPSystem::InvalidParameter);

       /**
       * Request to create a ThresholdMonitor to define the threshold
       * for some specific measurementTypes or subCounters
       */
       ManagedGenericIRPConstDefs::Signal create_threshold_monitor (
         in PMIRPConstDefs::MOClassNameType  moClass,
         in PMIRPConstDefs::MOInstanceListType  moInstanceList,
         in PMIRPConstDefs::ThresholdInfoListType thresholdInfoList,
         in PMIRPConstDefs::MonitorGranularityPeriodType monitorGranularityPeriod,
         out PMIRPConstDefs::MonitorIdType monitorId,
         out PMIRPConstDefs::MUnsupportedListType unsupportedList)
       raises (CreateThresholdMonitor,
               ManagedGenericIRPSystem::InvalidParameter,
               ManagedGenericIRPSystem::OperationNotSupported);

       /**
       * Request to delete a specified ThresholdMonitor
       */
       PMIRPConstDefs::ResultType delete_threshold_monitor (
         in PMIRPConstDefs::MonitorIdType monitorId)
       raises (DeleteThresholdMonitor,
               UnknownThresholdMonitor,
               ManagedGenericIRPSystem::OperationNotSupported);

       /**
       * Request to list detailed information about all or
       * specified ThresholdMonitors
       */
       PMIRPConstDefs::ResultType list_threshold_monitors (
         in PMIRPConstDefs::MonitorIdListType monitorIdList,
         out PMIRPConstDefs::MonitorInfoListType monitorInfoList)
       raises (ListThresholdMonitors,
               ManagedGenericIRPSystem::InvalidParameter,
               ManagedGenericIRPSystem::OperationNotSupported);

       /**
       * Request to suspend a ThresholdMonitor
       */
       PMIRPConstDefs::ResultType suspend_threshold_monitor (
         in PMIRPConstDefs::MonitorIdType monitorId)
       raises (SuspendThresholdMonitor,
               UnknownThresholdMonitor,
               ThresholdMonitorAlreadySuspended,
               ManagedGenericIRPSystem::OperationNotSupported);

       /**
       * Request to resume a ThresholdMonitor
       */
       PMIRPConstDefs::ResultType resume_threshold_monitor (
         in PMIRPConstDefs::MonitorIdType monitorId)
       raises (ResumeThresholdMonitor,
               UnknownThresholdMonitor,
               ThresholdMonitorIsNotSuspended,
               ManagedGenericIRPSystem::OperationNotSupported);

   };

};

#endif _PMIRPSYSTEM_IDL_
```

A.3 IDL specification (file name "PMIRPNotifications~~Defs~~.idl")

```
// File: PMIRPNotifications.idl

#ifndef _PMIRPNOTIFICATIONS_IDL_PMIRPNotifDefs_idl
#define _PMIRPNOTIFICATIONS_IDL_PMIRPNotifDefs_idl

#include "PMIRPConstDefs.idl"
#include "NotificationIRPConstDefs.idl"
#include "NotificationIRPNotifications.idl"

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

/* ## Module: PMIRPNotifDefs
This module contains the specification of all notifications of PM IRP Agent.
===============================================================
*/
module PMIRPNotificationsifDefs
{

   const string ET_MEASUREMENT_JOB_STATUS_CHANGED = "notifyMeasurementJobStatusChanged";
   const string ET_THRESHOLD_MONITOR_STATUS_CHANGED = "notifyThresholdMonitorStatusChanged";


   /**
   * Constant definitions for the notifyMeasurementJobStatusChanged notification
   */
   interface notifyMeasurementJobStatusChanged: NotificationIRPNotifications::Notify
ConstDefs::AttributeNameValue
   {
     const string EVENT_TYPE = "notifyMeasurementJobStatusChanged";
ET_MEASUREMENT_JOB_STATUS_CHANGED;


     /**
     * This constant defines the name of the jobId property.
     * which is transported in the filterable_body fields.
     * The data type for the value of this property
     * is PMIRPConstDefs::JobIdType.
     */
     const string JOB_ID = PMIRPConstDefs::AttributeNameValue::JOB_ID;

     /**
     * This constant defines the name of the jobStatus property.
     * which is transported in the filterable_body fields.
     * The data type for the value of this property
     * is PMIRPConstDefs::JobStatusType.
     */
     const string JOB_STATUS = PMIRPConstDefs::AttributeNameValue::JOB_STATUS;

     /**
     * This constant defines the name of the reason property.
     * which is transported in the filterable_body fields.
     * The data type for the value of this property is string.
     */
     const string REASON = PMIRPConstDefs::AttributeNameValue::REASON;
   };

   /**
   * Constant definitions for the notifyThresholdMonitorStatusChanged notification
   */
   interface notifyThresholdMonitorStatusChanged: NotificationIRPNotifications::Notify
ConstDefs::AttributeNameValue
   {
     const string EVENT_TYPE = "notifyThresholdMonitorStatusChanged";
ET_THRESHOLD_MONITOR_STATUS_CHANGED;

     /**
     * This constant defines the name of the monitorId property.
     * which is transported in the filterable_body fields.
     * The data type for the value of this property
     * is PMIRPConstDefs::MonitorIdType.
```

```
      */
      const string MONITOR_ID = PMIRPConstDefs::AttributeNameValue::MONITOR_ID;

      /**
      * This constant defines the name of the monitorStatus property.
      * which is transported in the filterable_body fields.
      * The data type for the value of this property
      * is PMIRPConstDefs::MonitorStatusType.
      */
      const string MONITOR_STATUS = PMIRPConstDefs::AttributeNameValue::MONITOR_STATUS;

      /**
      * This constant defines the name of the reason property.
      * which is transported in the filterable_body fields.
      * The data type for the value of this property is string.
      */
      const string REASON = PMIRPConstDefs::AttributeNameValue::REASON;
    };

};

#endif _PMIRPNOTIFICATIONS_IDL_
```

End of change in Clause A.1, A.2, A.3
End of document