Source: **SA5 (Telecom Management)**

Title: New Rel-6 TS 32.413-200 "Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)" - **For SA Approval**

Document for: **Decision**

Agenda Item: **7.5.3**

Although presented NOW for the 1st time to SA, SA5 asks SA#23 to approve this draft TS and place it under Change Control (CR-regime)

SP-040136	New Rel-6 TS 32.413-200 "Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)" - **For SA Approval**

3GPP TSG-SA5 (Telecom Management) **S5-046213**
Meeting #37, Málaga, España, 23 – 27 February 2004

Presentation of Technical Specification to TSG SA

Presentation to: **TSG SA Meeting #23**
Document for presentation: **TS 32.413, Version 2.0.0**
Presented for: **Approval**

Abstract of document:
This TS is intended for Release 6 and is part of the Performance Management IRP family consisting of:

Numbe	Title
32.411	Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Requirements
32.412	Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Information Service (IS)
32.413	**Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)**
32.414	Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Common Management Information Protocol (CMIP) Solution Set (SS)

The purpose of these specifications is to provide the essential Performance Management (PM) capabilities for measurement job administration, performance alarm handling and management of measurement file transfer over the Itf-N.

Work done against the WID contained in SP-020499 (Work Item ID: OAM-PM), approved in SA#17.

Changes since last presentation to TSG-SA:
New.
However, although presented NOW for the 1st time to SA, SA5 asks SA#23 to approve this draft TS and place it under Change Control (CR-regime)

Outstanding Issues:
None.

Contentious Issues:
None.

3GPP TS 32.413 V2.0.0 (2004-03)

Technical Specification

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

Keywords
Performance Management

3GPP

Postal address

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

Internet
http://www.3gpp.org

Contents

Foreword

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

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

The present document is part the 32.41x-series covering the 3rd Generation Partnership Project: Technical Specification Group Services and System Aspects; Telecommunication management; Performance Management (PM) Integration Reference Point (IRP), as identified below:

TS 32.411: "Requirements";

TS 32.412: "Information Service (IS)";

TS 32.413: "Common Object Request Broker Architecture (CORBA) Solution Set (SS)";

TS 32.414: "Common Management Information Protocol (CMIP) Solution Set (SS)".

The present document is part of a set of TSs which describe the requirements and information model necessary for the Telecommunication Management (TM) of 3G systems. The TM principles and TM architecture are specified in 3GPP TS 32.101 [1] and 3GPP TS 32.102 [2].

A 3G system is composed of a multitude of Network Elements (NE) of various types and, typically, different vendors, which inter-operate in a co-ordinated manner in order to satisfy the network users' communication requirements. Any evaluation of PLMN-system behaviour will require performance data collected and recorded by its NEs according to a schedule established by the EM.

This aspect of the management environment is termed Performance Management. The purpose of any Performance Management activity is to collect performance related data, which can be used to locate potential problems in the network.

1 Scope

The present document specifies the Common Object Request Broker Architecture (CORBA) Solution Set (SS) for the IRP whose semantics is specified in PM (Performance Management) IRP: Information Service [7].

This Solution Set specification is related to 3GPP TS 32.412 V6.1.X.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document.*

[1] 3GPP TS 32.101: "Telecommunication management; Principles and high level requirements".

[2] 3GPP TS 32.102: "Telecommunication management; Architecture".

[3] 3GPP TS 32.411: "Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Requirements".

[4] 3GPP TS 32.311: "Telecommunication management; Generic Integration Reference Point (IRP): Requirements".

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

[6] 3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects".

[7] 3GPP TS 32.412: "Telecommunication management; Performance Management (PM) Integration Reference Point (IRP): Information Service (IS)".

[8] 3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management: Information Service (IS)".

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

[10] 3GPP TS 32.401: "Telecommunication management; Performance Management (PM); Concept and requirements".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 32.101 [1], 3GPP TS 32.102 [2], 3GPP TS 32.401 [10], 3GPP TS 32.411 [3] and the following apply:

IRP document version number string (or "IRPVersion"): See 3GPP TS 32.311 [4].

3.2 Abbreviations

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

CM	Configuration Management
CORBA	Common Object Request Broker Architecture
DN	Distinguished Name
EM	Element Manager
IDL	Interface Definition Language
IS	Information Service
MOC	Managed Object Class
NE	Network Element
OMG	Object Management Group
PM	Performance Management
SS	Solution Set

4 Architectural features

The overall architectural feature of PMIRP is specified in 3GPP TS 32.411 [3].

This clause specifies features that are specific to the CORBA SS.

4.1 Notifications

Notifications are sent according to the Notification IRP: CORBA SS (see 3GPP TS 32.303 [5]).

The contents of the PMIRP notifications are defined in the present document.

4.2 Syntax for distinguished names and versions

The format of a Distinguished Name is defined in 3GPP TS 32.300 [6].

The version of this IRP is represented as a string (see also clause 3 for versions).

5 Mapping

5.1 Operation and Notification mapping

PMIRP: IS 3GPP TS 32.412 [7] defines semantics of operation and notification visible across the PMIRP. Table 1 indicates mapping of these operations and notifications to their equivalents defined in this SS.

Table 1: Mapping from IS Operations and Notification to SS equivalents

IS Operations/ notification 3GPP TS 32.412 [7]	SS Method	Qualifier
createMeasurementJob	create_measurement_job	M
stopMeasurementJob	stop_measurement_job	M
suspendMeasurementJob	suspend_measurement_job	O
resumeMeasurementJob	resume_measurement_job	O
listMeasurementJobs	list_measurement_jobs	M
createThresholdMonitor	create_threshold_monitor	O
deleteThresholdMonitor	delete_threshold_monitor	O
listThresholdMonitors	list_threshold_monitors	O
suspendThresholdMonitor	suspend_threshold_monitor	O
resumeThresholdMonitor	resume_threshold_monitor	O
getIRPVersion	get_PM_IRP_versions	M
getOperationProfile (see note)	get_PM_IRP_operations_profile	O
getNotificationProfile (see note)	get_PM_IRP_notification_profile	O
notifyMeasurementJobStatusChanged	push_structured_events(See subclause 6.1)	M
notifyThresholdMonitorStatusChanged	push_structured_events(See subclause 6.1)	O
NOTE: This operation is of ManagedGenericIRP IOC specified in 3GPP TS 32.312 [8]. The PMIRP IOC of [7] inherits from it.		

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 startTime	O
stopTime	PMIRPConstDefs::StopTimeTypeOpt stopTime	O
schedule	PMIRPConstDefs::ScheduleTypeOpt schedule	O
jobId	Return value of type PMIRPConstDefs::JobIdType	M
unsupportedList	PMIRPConstDefs::JUnsupportedListType unsupportedList	M
status	ManagedGenericIRPConstDefs::Signal Exception: CreateMeasurementJob, ManagedGenericIRPSystem::InvalidParameter, 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	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	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::MUnsupportedListType unsupportedList	M
status	ManagedGenericIRPConstDefs::Signal Exception: CreateThresholdMonitor, ManagedGenericIRPSystem::InvalidParameter	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	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	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	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	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

Table 13: Mapping from IS `getOperationProfile` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
iRPVersion	ManagedGenericIRPConstDefs::VersionNumber pm_irp_version	M
operationNameProfile,operationParameterProfile	Return of type ManagedGenericIRPConstDefs::MethodList	M
status	Exception: GetPMIRPOperationsProfile, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

Table 14: Mapping from IS `getNotificationProfile` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
iRPVersion	ManagedGenericIRPConstDefs::VersionNumber pm_irp_version	M
notificationNameProfile,notificationParameterProfile	Return value of type ManagedGenericIRPConstDefs::MethodList	M
status	Exception: GetPMIRPNotificationProfile, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

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	Type_name	M	This is the ET_MEASUREMENT_JOB_STATUS_CHANGED of module of PMIRPNotifDefs.
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 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]).
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 filterable_ body_fields	M	Name of NV pair is the JOB_STATUS of module PMIRPNotifDefs::notifyMeasurementJobStatusChanged

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
			Value of NV pair is JobStatusType of module PMIRPConstDefs.
reason	One NV pair of 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	Type_name	M	This is the ET_THRESHOLD_MONITOR_STATUS_CHANGED of module of PMIRPNotifDefs.
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 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.
monitorStatus	One NV pair of 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 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		

6 PMIRPNotification Interface

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

6.1 Method `push` (M)

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

NOTE 1: The `push_structured_events` method takes an input parameter of type `EventBatch` as defined in the OMG `CosNotification` module (OMG Notification Service [9]). This data type is the same as a sequence of Structured Events. Upon invocation, this parameter will contain a sequence of Structured Events being delivered to IRPManager by IRPAgent to which it is connected.

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

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

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

Annex A (normative): IDL specifications

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

```
#ifndef PMIRPConstDefs_idl
#define 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;
   typedef   unsigned long  ReportingPeriodType;
   typedef   TimeBase::UtcT UTCTimeType;

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

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

   typedef string IntervalTimeType;// the IntervalTimeType from 00:00 to 23:59 of one day
   struct IntervalOfDayType
   {
     IntervalTimeType intervalStartTime;
     IntervalTimeType intervalStopTime;
   };
   typedef sequence<IntervalOfDayType> DailySchedulingType;
   typedef unsigned short DayOfWeekType;
   //The value of DayOfWeekType is from 0 to 6.
   //sunday(0),monday(1),tuesday(2),wednesday(3),
   //thursday(4), friday(5), saturday(6)
   typedef sequence<DayOfWeekType> DaysOfWeekType;
   struct WeeklySchedulingElement
   {
     DaysOfWeekType dayOfWeek;
     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, Stoppped};
struct JobInfoType
{
 JobIdType  jobId;
 MOClassNameType  moClass;
 MOInstanceListType  moInstanceList;
 MeasurementCategoryListType  measurementCategoryList;
 GranularityPeriodType  granularityPeriod;
 ReportingPeriodType  reportingPeriod;
 StartTimeTypeOpt  startTime;
 StopTimeTypeOpt  stopTime;
 ScheduleTypeOpt  schedule;
 JobStatusType jobStatus;
};
typedef sequence<JobInfoType> JobInfoListType;

typedef string MeasurementTypeNameType;
typedef string SubCounterNameType;
typedef string ProbableCauseType;
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;
struct MonitorInfoType
```

```
   {
    MonitorIdType  monitorId;
    MOClassNameType  moClass;
    MOInstanceListType  moInstanceList;
    MonitorGranularityPeriodType monitorGranularityPeriod;
    ThresholdInfoListType thresholdInfoList;
   };
   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
```

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

```
#ifndef PMIRPSystem_idl
#define 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
{

   /**
   * the reason specifies whether EM or NE is high workload. The value shall be one
   * of following: emCpuBusy; emHDShortage, emLowMemory, {neCpuBusy, neObjectInstList},
   * {neHDShortage neObjectInstList}, {neLowMemory, neObjectInstList}, maxJobReached,
   * otherReason.
   */
   exception HighWorkLoad { string 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::StartTimeTypeOpt  startTime,
     in PMIRPConstDefs::StopTimeTypeOpt  stopTime,
     in PMIRPConstDefs::ScheduleTypeOpt  schedule,
     out PMIRPConstDefs::JobIdType jobId,
     out PMIRPConstDefs::JUnsupportedListType unsupportedList
   )
   raises (CreateMeasurementJob,
           ManagedGenericIRPSystem::InvalidParameter,
           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);

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

   /**
    * 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);

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

    /**
    * 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);

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

    /**
    * Request to resume a ThresholdMonitor
    */
    PMIRPConstDefs::ResultType resume_threshold_monitor (
      in PMIRPConstDefs::MonitorIdType monitorId)
    raises (ResumeThresholdMonitor,
```

```
              UnknownThresholdMonitor,
              ThresholdMonitorIsNotSuspended);

   };

};

#endif
```

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

```
#ifndef PMIRPNotifDefs_idl
#define PMIRPNotifDefs_idl

#include "PMIRPConstDefs.idl"
#include "NotificationIRPConstDefs.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 PMIRPNotifDefs
{

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


   /**
   * Constant definitions for the notifyMeasurementJobStatusChanged notification
   */
   interface notifyMeasurementJobStatusChanged: NotificationIRPConstDefs::AttributeNameValue
   {
     const string EVENT_TYPE = 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: NotificationIRPConstDefs::AttributeNameValue
   {
     const string EVENT_TYPE = ET_MEASUREMENT_JOB_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
```

Annex B (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
Mar 2004	S_23	SP-040136	--	--	Submitted to TSG SA#23 for Approval	2.0.0	