

IPWorks Performance Measurements

DESCRIPTION

Copyright

© Ericsson AB 2017, 2018. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



Contents

1	Introduction	1
1.1	Prerequisites	1
1.2	Related Information	1
2	Measurement Parameters	3
3	Measurement Description	5
4	Performance Measurement Jobs	7
4.1	Defining Performance Management Jobs	7
4.2	List Performance Management Information	12
4.3	Delete Performance Management Jobs	12
4.4	Changing Performance Management Jobs	12
4.5	Upgrading Performance Management	13
5	Measurement Report File	15
5.1	File Location	15
5.2	Log Rotation of Report Files	15
5.3	File Name Format	16
5.4	Time Format	16
5.5	File Format	17
5.6	Supported XML Elements	19
5.7	Sample File	21
6	Display Performance Measurements	23
7	KPI of AAA	25
7.1	Accessibility	25
7.2	Capacity - Session	26
7.3	Capacity - TPS	26
	Reference List	27





1 Introduction

This document describes Performance Measurement principles and the configuration of Measurement Jobs for the IPWorks.

Scope

This document describes how the IPWorks measurements can be used for IPWorks performance monitoring, designing performance reports, and evaluating IPWorks performance data.

Target Groups

This document is written for Operation and Maintenance personnel working on IPWorks performance measurements.

1.1 Prerequisites

Users of this document must have knowledge and experience with the following:

- UNIX
- Ericsson Command Line Interface (ECLI), refer to *Ericsson Command-Line Interface User Guide*.
- Basic knowledge about the IPWorks

1.2 Related Information

Reference information on the terms and concepts employed, as well as the typographic conventions used throughout this document is available in different companion documents. These documents are:

- *Trademark Information*
- *Typographic Conventions*
- *Glossary of Terms and Acronyms*





2 Measurement Parameters

The measurement parameters are listed and described in the *MeasurementType* MO in *Managed Object Model (MOM)*.

The following list contains additional information on measurement parameters presented in the *MeasurementType* MO:

The following list contains additional information on measurement parameters presented in the *MeasurementType* MO:

Aggregation

Defines how data is aggregated for measurement instances:

- **SUM:** All measured values are added during a granularity period.
- **AVG:** An average value is calculated, using all values collected during a granularity period. If the Collection Method is **GAUGE**, the average can be calculated over several granularity periods.
- **MIN:** The minimum value during a granularity period is collected.
- **MAX:** The maximum value during a granularity period is collected.
- **LAST_UPDATE:** The last value during a granularity period is collected.

Collection Method

The form in which this measurement data is obtained:

- **CC:** Cumulative Counter, when numeric data being measured needs to be collected cumulatively. The value of such a counter can only increase unless it is reset.
- **GAUGE:** Dynamic variable, used when data being measured can vary up or down during the period of measurement.
- **DER:** Discrete Event Registration; when data related to a particular event are captured, every nth event is registered, where n can be 1 or greater.
- **SI:** Status Inspection, used for generic purposes. Can be numeric or string.



Description	Describes what the measurement monitors, and at what event it is triggered.
FM Alarm Type	Reference to an Alarm that will be triggered if a threshold is crossed. Only measurements having this attribute defined can be monitored in a Threshold Job.
Measurement Status	<p>The current status of the measurement. The status must follow the order: PRELIMINARY => USED => DEPRECATED => OBSOLETE. The statuses are used as follows:</p> <ul style="list-style-type: none">• PRELIMINARY: The measurement is available, storing data, but is not approved.• USED: The measurement is available, storing data, and approved.• DEPRECATED: The measurement is available, but is not storing data.• OBSOLETE: The measurement is not available, and is not storing data.
Measurement Type ID	A unique ID for the measurement type. It is the same as the Measurement Name, unless the name is longer than 32 characters. In that case, the ID is a short form of the name.
Threshold Direction	The direction of crossing a threshold that will trigger a notification alarm. This attribute is valid for measurements monitored in a Threshold Job.



3 Measurement Description

The measurements of the IPWorks are distributed over several groups (PmGroups). Grouping is based on the Managed Object Class (MOC) that a Measurement Type refers to. A MOC may be a Class from the *Managed Object Model (MOM)* of the IPWorks. The actual counting of values on a Measurement Type is implemented as one or more Measurement Instances, which are created during run-time. When a new instance of a MOC is configured by the operator, then a new Measurement Instance is created immediately for a Measurement Type, and the Measurement Instance is named using the Local Distinguished Name (LDN) of the MOC instance. Thus all Measurement Types that have Measurement Instances named after LDNs of the same MOC in the IPWorks MOM, are grouped under the same PmGroup.

The following list describes the MOC of each PmGroup:

- The `DnsASDNSStatisticsGroup` measurement group includes the measurements related to ASDNS service. The MOC is the `DnsASDNSStatisticsGroup`.
- The `DnsDNSSECStatisticsGroup` measurement group includes the measurements related to the DNS security operations. The MOC is the `DnsDNSSECStatisticsGroup`.
- The `DnsForwardStatisticsGroup` measurement group includes the measurements related to the queries that DNS forwards to another DNS for recursive query. The MOC is the `DnsForwardStatisticsGroup`.
- The `DnsNotifyStatisticsGroup` measurement group includes the measurements related to the notify requests the DNS server receives. The MOC is the `DnsNotifyStatisticsGroup`.
- The `DnsQueryStatisticsGroup` measurement group includes the measurements related to query statistics for DNS. The MOC is the `DnsQueryStatisticsGroup`.
- The `DnsTSIGStatisticsGroup` measurement group includes the measurements related to TSIG Statistics of DNS. The MOC is the `DnsTSIGStatisticsGroup`.
- The `DnsUpdateStatisticsGroup` measurement group includes the measurements related to DNS update requests. The MOC is the `DnsUpdateStatisticsGroup`.

For more detailed information about each group and the specific Measurement Types, refer to *IPWorks Measurement List*.





4 Performance Measurement Jobs

A Performance Measurement Job is the process executed in the Network Element (NE) in order to either accumulate data or generate alarms.

Specifically, a Measurement Job is the process to accumulate measurement result data and assemble it for collection, inspection, or both. It can be defined either to collect the values for a Group of Measurement Types or to collect the value of a specific Measurement Type.

A Threshold Job is the process to monitor for surpassed thresholds. When a threshold is surpassed, a notification (alarm or alert) is issued. A Threshold Job does not generate any measurement data.

The following sections describe the way to define, delete or change a Performance Management Job.

4.1 Defining Performance Management Jobs

The definition of PM Jobs is performed by the operator through the ECLI.

A Performance Management Job can be defined either as a Measurement Job or as a Threshold Job. The details are provided in the next sections.

Note: Due to system wide constraints, the RDN identifying a MO must be less than 64 characters long. Also, the DN of the MO must be below the 256 characters limit. Do not use unnecessarily long MO names.

The default PM job will be created by IPWorks initialization. The jobGroup and compressionType of these jobs are unset. If customers want to create a PM job which uses compress mode, the jobGroup of the PM job must be specified. More information about jobGroup refers to "*Performance Management Report File Format*".

4.1.1 Measurement Jobs

When a Measurement Job is defined, the operator sets the measurement specification, the granularity period, the job priority and the requested job state. It is the operator's responsibility to ensure that all these attributes are defined properly in order to collect the result data. The measurement specification can either be a Group of Measurement Types or a specific Measurement Type. For more information about these attributes, see Section 4.1.4 on page 11.

Each Measurement Job collects result data at a particular frequency, controlled by the granularity period of the Measurement Job. At the end of the granularity period a scheduled result report is generated for each Measurement Job



that is actively collecting performance measurement result data, for all the Measurement Types covered by the job.

Measurement Jobs are used for the operator to monitor measurements by collecting measurement results. The collection method of each Measurement Type is either "Cumulative Counter" (CC) or "Gauge Counter" (GAUGE). The CC collection method is used for counting events.

Each time the event occurs a value is added to the measurement. The measurement is reset at the beginning of each granularity period and will only have a valid result at the end of the granularity period. The GAUGE collection method is used to show a current value. For example, during the granularity period, the maximum value of the `ipworksDnsServTransactionPerSecondGauge` GAUGE is kept and reported. The measurement is reset at the beginning of each granularity period.

For how to create Measurement Collection Jobs, refer to *Create Measurement Collection Job*. These Measurement Jobs collect measurement data for all the Measurement Types that are affected during a granularity period belonging to these `PmGroups`, thus, all measurements of IPWorks are monitored.

4.1.2 Measurement Report File

One Measurement Report File is generated by the NE immediately at the end of each granularity period. This file contains all measurement results produced by the NE within that granularity period. For example, if IPWorks runs 10 measurements with a granularity period of 15 minutes and 5 measurements with a granularity period of 5 minutes, then it generates one file containing 10 results every 15 minutes, and one file containing 5 results every 5 minutes. If two or more granularity periods come to an end at the same time, the NE generates one file per granularity period. Hence in the above example, the IPWorks generates 2 files – one containing 10 results (15 min granularity period) and the other containing 5 measurement results (5 min granularity period), only when the end time of the granularity periods do not coincide. The NE and the granularity period can be identified both in the file name and the file contents.

4.1.3 Threshold Job

Threshold Jobs are used by the operator to be able to setup alarm notifications. When a Threshold Job is defined, the operator has to set the measurement specification, the granularity period, the job priority, the requested job state, the reporting period and the thresholds used for generating Performance Management alarm notifications. It is the operator's responsibility to ensure that all the attributes and threshold values are defined properly in order to detect performance degradations before they start affecting service.

The measurement specification can only be a Measurement Type having values to the attributes FM Alarm Type and Threshold Direction. A list of Measurement Types with these attributes defined is shown in Table 1. The threshold is monitored based on a monitor granularity period, where the monitor



granularity period is a fraction of the measurement collection granularity period. Following the threshold creation, it is possible to query the threshold information defined for an object instance. The threshold definition allows the user to assign up to four different severity levels (critical, major, minor, warning) based on different threshold values. For more information about these attributes, see Section 4.1.4 on page 11.

Table 1 Measurement Types with Attributes for Threshold Job

Measurement Type	FM Alarm Type
ipworksDnsServTransactionPerSecondGauge	ipworksQueryFailureError

Alarm notifications are generated as follows:

- New alarm notification

A new alarm notification is sent when an alarm criteria is fulfilled and no alarm is active for the measured instance.

- Changed alarm notification

A changed alarm notification is sent, at the end of the GP, when a new alarm criteria is fulfilled while an alarm is active.

- Cleared alarm notification

A cleared alarm notification is sent, at the end of the GP, when all alarms clear criteria is fulfilled.

The Threshold Monitoring Class defines two thresholds, low and high. This can be used to define a hysteresis for the corresponding alarm. If the Threshold Direction (as described in Section 2 on page 3) is INCREASING and the measurement value at the end of the Granularity Period is higher than the High threshold, a New or Changed alarm notification is issued. If the value is lower than the Low threshold a Cleared or Changed alarm notification is issued. If Threshold Direction is DECREASING the opposite applies.

In Figure 1, an example of alarm notifications using two Threshold Monitoring MOs A1 and A2 is shown. The Threshold Direction is INCREASING.

Threshold monitoring can be used with both Cumulative Counters (CC) and Gauges. The measurement value in the threshold parameters is defined in different ways:

- Gauge

An absolute value.

- CC

Rate of variation, meaning that the total value is reduced to counts per second at the end of the granularity period.

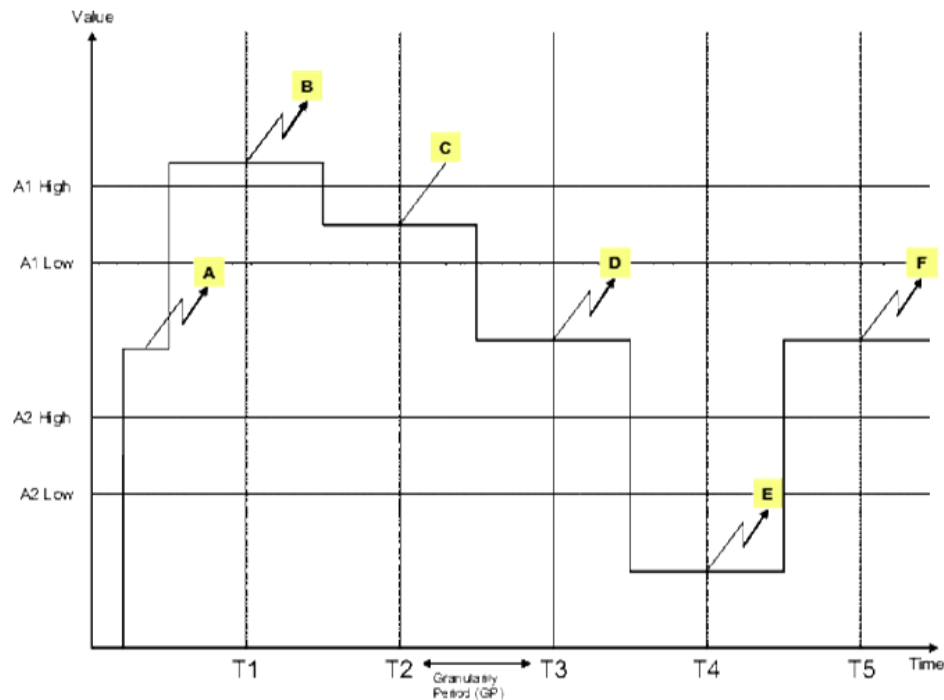


Figure 1 Notifications - Examples

- A** Nothing happens, since this is not at the end of a GP.
- B** The measurement value is above the **A1 High** threshold. No previous alarm is "active". A New alarm notification is issued.
- C** The measurement value is below **A1 High** but above **A1 Low**, indicating that the A1 alarm is still "active". No notification is issued.
- D** The measurement value is below **A1 Low**, indicating that the A1 alarm is no longer "active", but it is above **A2 High**, indicating that A2 alarm is now "active". A Changed alarm notification is issued.
- E** The measurement value is below both **A1 Low** and **A2 Low**, indicating that no alarm is "active". A Cleared alarm notification is issued.
- F** The measurement value is above **A2 High** but below **A1 High**, indicating that A2 alarm is "active". A New alarm notification is issued.



Information about the alarms that are raised or ceased can be found in *IPWorks Alarm List*.

For more information on how to create a Threshold Job, refer to *Create Threshold Monitoring Job*.

4.1.4 Supported Attributes for Defining a Job

The supported TimePeriods are shown in Table 2.

Table 2 Supported TimePeriods

TimePeriod
ONE_MIN
FIVE_MIN
FIFTEEN_MIN (Default value)
THIRTY_MIN
ONE_HOUR
TWELVE_HOUR
ONE_DAY

The supported JobPriorities are shown in Table 3.

Table 3 Supported JobPriorities

JobPriority
LOW
MEDIUM (Default value)
HIGH

The supported JobTypes are shown in Table 4.

Table 4 Supported JobTypes

JobType
MEASUREMENTJOB (Default value)
THRESHOLDJOB

The supported JobStates are shown in Table 5.

*Table 5 Supported JobStates*

JobState
ACTIVE (Default value)
STOPPED

The supported ThresholdSeverities are shown in Table 6.

Table 6 Supported ThresholdSeverities

ThresholdSeverity
CLEARED
CRITICAL
MAJOR
MINOR (Default value)
WARNING

4.2 List Performance Management Information

For information on how to list Performance Management information, refer to the *List Performance Management Groups and Measurement Types* and *List Performance Management Jobs*.

4.3 Delete Performance Management Jobs

A Performance Management Job can be deleted in order to stop the action defined in it.

If the operator deletes a Measurement Job, then the measurement report files stop storing measurement values of the monitored object, which is either a Measurement Type or a Group of Measurement Types. The existing measurement report files are not removed when a Measurement Job is deleted.

If the operator deletes a Threshold Job, then the relevant threshold is no longer monitored.

The way to delete a Job is the same for all Job types: first the Job must be stopped and then the Job can be deleted. For more information, refer to *Delete Threshold Monitoring Job* or *Delete Measurement Collection Job*.

4.4 Changing Performance Management Jobs

A Performance Management Job cannot be changed after it has been defined. It can only be stopped or activated.



If there is a need to change attributes of a job, then a new job must be created with the new attribute values. When the new job is in place, the old one with the old attribute values can be deleted, refer to *Delete Threshold Monitoring Job* or *Delete Measurement Collection Job*.

4.5 Upgrading Performance Management

When a new IPWorks software is released, there is a chance that new Measurement Types or new Groups are defined or existing Measurement Types or Groups are removed. For this reason, any defined Performance Management Jobs are stopped and removed. Some default Jobs may be started by the deployment of the new software release. It is the operator's responsibility to redefine new Performance Management Jobs, which will restart the production of Measurement Report files and trigger the desired alarms.

If the new IPWorks software does not modify the Performance Management structure, the existing Measurement Jobs are not affected.





5 Measurement Report File

A report file contains results of measurements collected from a single managed element during a single granularity period by all active Measurement Jobs.

Report files are created at the end of the reporting periods, one file for each granularity period.

Note: As a limitation, IPWorks considers the reporting period to be equal to the value of the granularity period, as specified in the `granularityPeriod` attribute value of the *PmJob* Managed Object (MO); the value of the `reportingPeriod` attribute is effectively ignored.

Additionally, the minimum recommended value for is five minutes, thus the Measurement Report Files can be produced every five minutes. If the value of `granularityPeriod` is set to be less than the recommended value, then the report files will be still produced every five minutes. The contents of the report files will contain the measurement values of the last granularity interval period.

5.1 File Location

The measurement report files are generated in the directory, as specified in the `fileGroup` attribute of the *PmMeasurementCapabilities* MO. The `fileGroup` attribute is read-only and can be seen by navigating to the *PmMeasurementCapabilities* MO of the Pm MOM through the CLI.

The directory with the measurement report files, as seen through SFTP to the IPWorks O&M IP, is in `/storage/no-backup/nbi_root/PerformanceManagementReportFiles`.

5.2 Log Rotation of Report Files

The IPWorks supports log rotation of published reports to prevent the number of reports growing indefinitely. Log rotation is performed based on the number of reports, not the size or time since creation.

The number of allowed reports is stored in the `maxNoOfPmFiles` attribute found in the *PmMeasurementCapabilities* MO. If the number of reports exceeds this limit, reports are removed in the order they were created until the total number of reports is less than `maxNoOfPmFiles`. This attribute is read-only in IPWorks and its value is set to 1000.

5.3 File Name Format

The report XML files are compliant with type “A” of [3GPP TS 32.432 v10.0.0; Performance measurement: File format definition](#). This means that the file contains results of measurements collected from a single managed element during a single granularity period.

The file name has the following format:

`A<StartDate>.<StartTime>-<EndTime>_<UniqueId>.xml`

Where:

- `StartDate` – Specifies the Coordinated Universal Time (UTC) date when the granularity period began and has the format YYYYMMDD.
- `StartTime` – Specifies the UTC time when the granularity period began and has the format HHMM+0000.
- `EndTime` – Specifies the UTC time when the granularity period ended and has the format HHMM+0000.
- `UniqueId` – Specifies the unique name of the managed element, as defined in the key value of the `ManagedElement` MO.

A sample file name is shown in Example 1.

`A20120901.1150+0000-1155+0000_Stockholm.xml`

Example 1 Sample File Name

5.4 Time Format

Times in the measurement result files are expressed in local time and have the following format:

`YYYYMMDDhhmmss.sTZ`

Where:

- `YYYY` – Four-digit year.
- `MM` – Two-digit month (01–12).
- `DD` – Two-digit day of month (01–31).
- `hh` – Two-digit hour (00–23).
- `mm` – Two-digit minute (00–59).
- `ss.s` – Two-digit of second (00–59) and one digit representing the decimal fraction of a second.



- TZ – Time zone designator: + hhmm or - hhmm.

An example time is shown in Example 2.

20120901221510.1+0200

Example 2 Example Time

The `alignedReportingPeriod` attribute in `PmMeasurementCapabilities` MO defines if the granularity and reporting periods are aligned to whole minutes. It is a read-only attribute and its default value is set to `true`. For example, measurement results are reported at 10:00, 10:10, 10:20, and so on, if the granularity period is 10 minutes.

5.5 File Format

This section describes the measurement report file format. The XML document type declaration contains the markup declarations that provide a grammar for the measurement file format. This grammar is known as a Document Type Definition (DTD). This solution set defines the DTD to be used.

The measurement report file is compliant with the 3GPP TS 32.435 v10.0.0; PM XML File Format Definition in the following URL:

- [3GPP TS 32.435 v10.0.0; Performance measurement; eXtensible Markup Language \(XML\) file format definition](#)

By the `measCollec.xsd` schema, the standard defines the structure, name, and number of XML elements allowed in the XML file, as summarized in Figure 2.

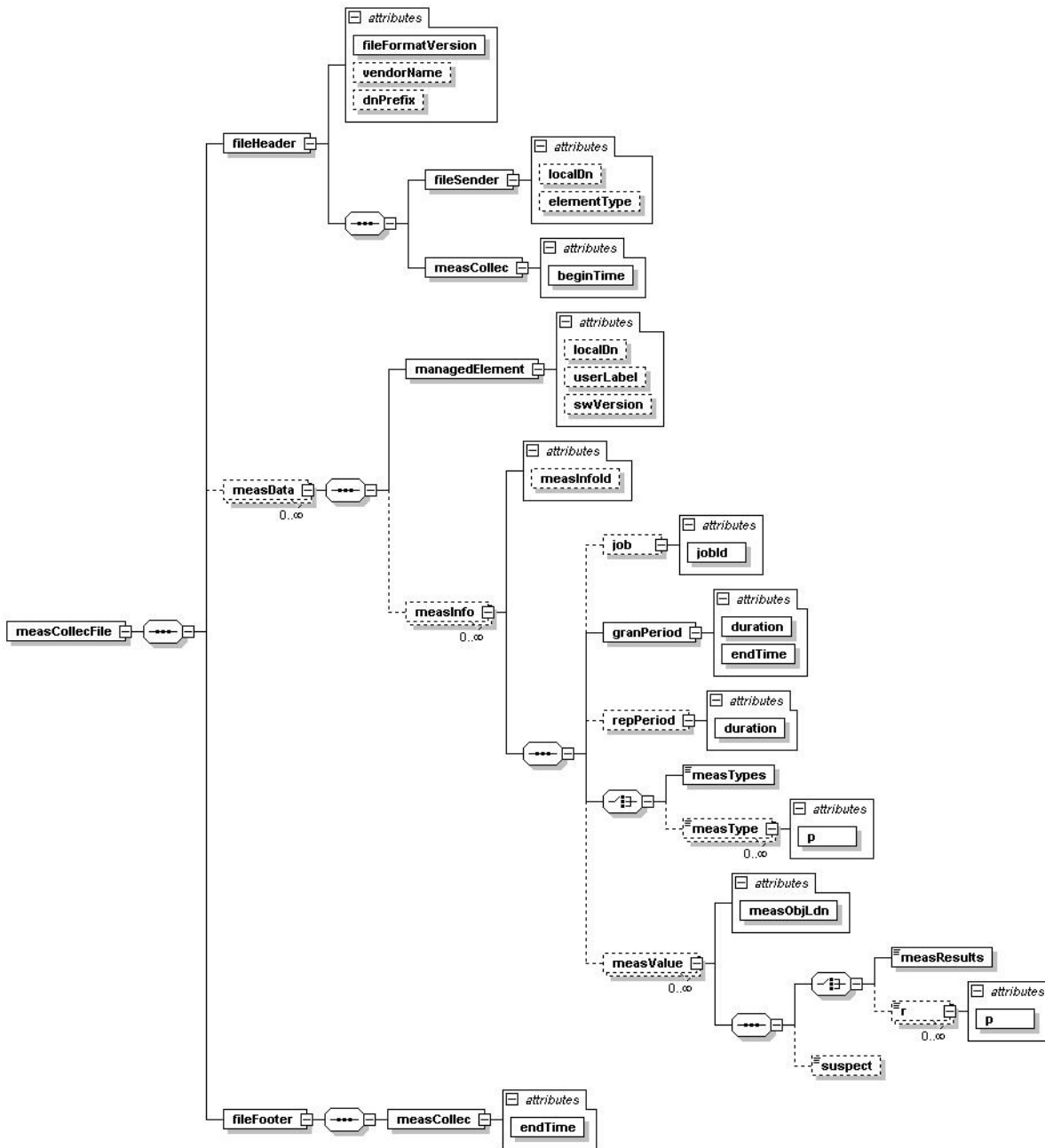


Figure 2 XML Schema Diagram

Details on support for the elements are specified in Section 5.6 Supported XML Elements.

An example of a measurement report file is provided in Section 5.7 Sample File.



5.6 Supported XML Elements

This section describes the elements of the measurement report file.

The supported XML elements and attributes are shown in Table 7, Table 8, and Table 9.

The root element is `measCollecFile`.

Table 7 Supported XML Elements in Header of Report File

XML Element or Attribute	Description
<code>beginTime</code>	Start time of the measurement data collection. Example: 2010-05-17T20:25:00-02:00.
<code>dnPrefix</code>	Distinguished Name (DN) prefix, as specified in the <code>dnPrefix</code> attribute of <code>ManagedElement</code> . The <code>dnPrefix</code> element is present in the report file if the <code>dnPrefix</code> attribute is specified, that is, not an empty string. Example: DC=ericsson.se,g3SubNetwork=Sweden.
<code>elementType</code>	The element type, as specified in the key value and the <code>productDesignation</code> structure member in the <code>productIdentity</code> attribute of <code>ManagedElement</code> . The <code>elementType</code> element is present in the report file if <code>productDesignation</code> is specified, that is, not an empty string. Example: IPWorks.
<code>fileFormatVersion</code>	The format version identifier, that is, 32.435 V10.0.
<code>fileHeader</code>	Root element of the file header.
<code>localDn</code>	The Local Distinguished Name (LDN) of <code>ManagedElement</code> . Example: <code>ManagedElement=Stockholm</code> .
<code>vendorName</code>	Vendor name, that is, Ericsson.

**Table 8** Supported XML Elements in Measured Data Section of Report File

XML Element or Attribute	Description
duration in granPeriod	<p>The granularity period in the format PT <n> S, where n denotes seconds and can be 10, 30, 60, 300, 900, 1800, 3600, 43200, or 86400.</p> <p>The duration attribute can have the Unknown Time value, that refers to an internal error. This value is specified by the granularityPeriod attribute value of the PmJob MO.</p> <p>Example: PT300S.</p>
duration in repPeriod	<p>The reporting period in the format PT <n> S, where n denotes seconds and can be 10, 30, 60, 300, 900, 1800, 3600, 43200, or 86400.</p> <p>The duration attribute can have the Unknown Time value, that refers to an internal error. This value is specified by the reportingPeriod attribute value of the PmJob MO.</p> <p>Example: PT60S.</p>
endTime in granPeriod	<p>End time of the actual granularity period.</p> <p>Example: 2010-05-17T20:30:00-02:00.</p>
granPeriod	<p>Container for the duration and endTime elements.</p> <p>Example: PT300S.</p>
jobId	<p>The identifier of the measurement job, as specified in the pmJobId attribute of the PmJob MO.</p> <p>Example: DnsASDNSStatisticsGroupJob.</p>
localDn	<p>The LDN of ManagedElement.</p> <p>Example: ManagedElement=Stockholm.</p>
managedElement	<p>Container for the optional elements localDn, userLabel, and swVersion.</p>
measData	<p>Root element of the measurement data.</p>
measInfo	<p>Root element of the measurement information.</p>
measObjLdn	<p>The LDN of the measured object.</p> <p>Example: IPWPMInstance_SC_1.</p>
measType	<p>The name of the measurement type, as specified in the measurementName attribute of MeasurementType.</p> <p>Example: CC_IN_REQ_NUM_OF_MSGS_RECEIVED.</p>



Table 8 Supported XML Elements in Measured Data Section of Report File

measValue	Container element for the measured values. It is an empty element (<measValue/>) if no measurement value is available.
p	Positioning element that connects the measurement type and value.
r	<p>The measurement result as a decimal number in one of the following formats:</p> <ul style="list-style-type: none"> • 64-bit signed integer with range -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. • 64-bit floating point form with at least five digits after the decimal point. <p>The <code>r</code> element is present only if measurement data is available for the measured object in the actual granularity period.</p>
repPeriod	The optional container for the duration element.
suspect	Optional element. The value <code>true</code> indicates that the measurement result is suspected to be incomplete due to an internal error.
swVersion	<p>The element type, as specified in the key value and the <code>productRevision</code> structure member in the <code>productIdentity</code> attribute value of <code>ManagedElement</code>.</p> <p>The <code>elementType</code> element is present in the report file if <code>productRevision</code> is specified, that is, not an empty string.</p> <p>Example: R7A.</p>
userLabel	<p>User-defined label, as specified in the <code>userLabel</code> attribute of <code>ManagedElement</code>.</p> <p>The <code>userLabel</code> element is present in the report file if the <code>userLabel</code> attribute is specified, that is, not an empty string.</p>

Table 9 Supported XML Elements in Footer of Report File

XML Element or Attribute	Description
endTime	<p>End time of the measurement data collection.</p> <p>Example: 2012-09-28T20:30:00-02:00.</p>
fileFooter	Root element of the file footer.

Note: The XML elements `measInfoId` and `measResults` are not supported.

5.7 Sample File

A PM XML report file valid for the XML schema is shown in Example 3.



```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="MeasDataCollection.xsl"?>
<measCollecFile xmlns="http://www.3gpp.org/ftp/specs/archive/32_series/32.435#measCollec">
  <fileHeader fileFormatVersion="32.435 V10.0" vendorName="Ericsson AB">
    <fileSender localDn="ManagedElement=ipworks_cba" elementType="ERIC-COREMW_RUNTIME"/>
    <measCollec beginTime="2016-07-21T05:16:00+02:00"/>
  </fileHeader>
  <measData>
    <managedElement localDn="ManagedElement=ipworks_cba" swVersion="CXP9020355_2 R1C01"/>
    <measInfo measInfoId="AAADiaOtherGroup">
      <job jobId="eyotang"/>
      <granPeriod duration="PT60S" endTime="2016-07-21T05:17:00+02:00"/>
      <repPeriod duration="PT60S"/>
      <measType p="1">ipworksDiameterAAAHealthStatus</measType>
      <measType p="2">ipworksDiameterAAAMAPATIRequests</measType>
      <measType p="3">ipworksDiameterAAAMAPATIResponses</measType>
      <measType p="4">ipworksDiameterAAAOCSPRequests</measType>
      <measType p="5">ipworksDiameterAAAOCSPResponses</measType>
      <measType p="6">ipworksDiameterAAAProcessUpTime</measType>
      <measValue measObjLdn="PL-3_67E705FF-9800-4AA4-88A0-5B219136E678">
        <r p="1">1</r>
        <r p="2">0</r>
        <r p="3">0</r>
        <r p="4">0</r>
        <r p="5">0</r>
        <r p="6">951</r>
      </measValue>
    </measInfo>
  </measData>
</fileFooter>
  <measCollec endTime="2016-07-21T05:17:00+02:00"/>
</fileFooter>
</measCollecFile>
```

Example 3 Sample Measurement Report File



6 Display Performance Measurements

The command `cmw-pm-show-counters` provides the method to display measurement values for an active measurement instance. By using this command, the operators can quickly fetch the measurement values without waiting until a GP finishes. The measurement values are related to aggregation method of the measurement type. All measurement values refresh every 10 seconds.

The measurement values displayed by the command are related to the aggregation methods described as below:

- **SUM:** All measured values are added from the beginning of the latest GP to present.
- **AVG:** An average value is calculated, using all values collected from the beginning of the latest GP to present.
- **MIN:** The minimum value collected from the beginning of the latest GP to present.
- **MAX:** The maximum value collected from the beginning of the latest GP to present.
- **LAST_UPDATE:** The last value collected from the beginning of the latest GP to present.

Command Usage:

```
cmw-pm-show-counters [-v] <instance> [-t <timeout>]
    [-j <job_name> [<job_name> ...] ] [-m <meas_type> [<meas_type>]
Display active PM values for a specified instance.
-v|--verbose           verbose
-t|--timeout           command timeout in seconds
-j                     job name
-m                     measurement type
```

If -j is specified only instance values from the specified job_name otherwise instance values from all the jobs are shown.

If -m is specified only instance values for the meas_type list

Example 1: Displaying Measurement Type

```
# cmw-pm-show-counters -v PL-3_67E705FF-9800-4AA4-88A0-5B2
19136E678 -m ipworksDiameterAAAProcessUpTime

ipworksDiameterAAAProcessUpTime 62645      AAADiaDefaultPmJob 15_MIN
ipworksDiameterAAAProcessUpTime 62645      TestPmJob 1_MIN
```



In this example, the measurement instance name is `PL-3_67E705FF-9800-4AA4-88A0-5B219136E678`, which is stored as the measurement value of `measObjLdn` in PM report.

The command output shows that two PM jobs are created for the measurement type `ipworksDiameterAAAProcessUpTime`. 62645 is the `uptime(seconds)` of the measurement type on PL-3.

Example 2: Displaying Measurement Type with Specified Job Name

```
# cmw-pm-show-counters -v PL-3_67E705FF-9800-4AA4-88A0-5B219136E678 -j AAADiaDefaultPmJob -m ipworksDiameterAAAProcessUpTime
```

```
PL-3_67E705FF-9800-4AA4-88A0-5B219136E678  
ipworksDiameterAAAProcessUpTime 62775      AAADiaDefaultPmJob 15_MIN
```



7 KPI of AAA

7.1 Accessibility

- RADIUS Authentication Success Ratio

$$= \frac{\text{ipworksRadiusAuthServTotalAccessAccepts}}{\text{ipworksRadiusAuthServTotalAccessRequests}} \times 100\%$$

- LDAP Query Success Ratio (AAA to CUDB)

$$= \frac{\text{ipworksRadiusServTotalSuccessCUDBQueryRequests}}{\text{ipworksRadiusServTotalCUDBQueryRequests}} \times 100\%$$

- STa+ EAP-AKA Authentication Success Ratio**

$$= \frac{(\text{ipworksDiameterAAASTaPlusEAPAKAAuthSuccessNumber} + \text{ipworksDiameterAAASTaPlusFastEAPAKAReAuthSuccessNumber})}{(\text{ipworksDiameterAAASTaPlusEAPAKAAuthSuccessNumber} + \text{ipworksDiameterAAASTaPlusFastEAPAKAReAuthSuccessNumber} + \text{ipworksDiameterAAASTaPlusEAPAKAAuthFailedNumber} + \text{ipworksDiameterAAASTaPlusFastEAPAKAReAuthFailedNumber})} \times 100\%$$

- STa+ EAP-AKA' Authentication Success Ratio**

$$= \frac{(\text{ipworksDiameterAAASTaPlusEAPAKAPrimeAuthSuccessNumber} + \text{ipworksDiameterAAASTaPlusFastEAPAKAPrimeReAuthSuccessNumber})}{(\text{ipworksDiameterAAASTaPlusEAPAKAPrimeAuthSuccessNumber} + \text{ipworksDiameterAAASTaPlusFastEAPAKAPrimeReAuthSuccessNumber} + \text{ipworksDiameterAAASTaPlusEAPAKAPrimeAuthFailedNumber} + \text{ipworksDiameterAAASTaPlusFastEAPAKAPrimeReAuthFailedNumber})} \times 100\%$$

- STa EAP-AKA' Authentication Success Ratio**

$$= \frac{\text{ipworksDiameterAAASTaEAPAKAPrimeAuthSuccessNumber}}{(\text{ipworksDiameterAAASTaEAPAKAPrimeAuthSuccessNumber} + \text{ipworksDiameterAAASTaEAPAKAPrimeAuthFailedNumber})} \times 100\%$$

- SWm EAP-AKA Authentication Success Ratio**

$$= \frac{(\text{ipworksDiameterAAASWmEAPAKAAuthSuccessNumber} + \text{ipworksDiameterAAASWmFastEAPAKAReAuthSuccessNumber})}{(\text{ipworksDiameterAAASWmEAPAKAAuthSuccessNumber} + \text{ipworksDiameterAAASWmFastEAPAKAReAuthSuccessNumber} + \text{ipworksDiameterAAASWmEAPAKAAuthFailedNumber} + \text{ipworksDiameterAAASWmFastEAPAKAReAuthFailedNumber})} \times 100\%$$

- SWm EAP-TLS Authentication Success Ratio (%)**



$$= \frac{ipworksDiameterAAASWmPlusEAPTLSAuthSuccessNumber}{(ipworksDiameterAAASWmPlusEAPTLSAuthSuccessNumber + ipworksDiameterAAASWmPlusEAPTLSAuthFailedNumber)} \times 100\%$$

- **SWx SAR Request per Second**

$$= \frac{ipworksDiameterAAASWxSAR}{sampling\ period(s)}$$

- **SWx MAR Response Ratio**

$$= \frac{ipworksDiameterAAASWxMAA}{ipworksDiameterAAASWxMAR} \times 100\%$$

- **SWx SAR Response Ratio**

$$= \frac{ipworksDiameterAAASWxSAA}{ipworksDiameterAAASWxSAR} \times 100\%$$

7.2 Capacity - Session

- **Active Session Number for AAA RADIUS (Gi/SGi)**

$$= ipworksRadiusServActiveSessionNumber$$

- **Active Session Number for Trusted CDMA Access (STa)**

$$= ipworksDiameterAAAActiveSTaSessionNumber + ipworksDiameterAAAActiveS6bSessionNumber$$

- **Active Session Number for Trusted WLAN Access (STa+)**

$$= ipworksDiameterAAAActiveSTaPlusSessionNumber + ipworksDiameterAAAActiveS6bSessionNumber$$

- **Active Session Number for Trusted WLAN Access (SWm/SWm+)**

$$= ipworksDiameterAAAActiveSWmSessionNumber + ipworksDiameterAAAActiveSWmPlusSessionNumber + ipworksDiameterAAAActiveS6bSessionNumber$$

7.3 Capacity - TPS

For the detail of TPS, refer to *IPWorks Measurement List*.



Reference List

- [1] *Trademark Information*
- [2] *Typographic Conventions*
- [3] *Glossary of Terms and Acronyms*
- [4] *IPWorks Performance Measurements*
- [5] *Ericsson Command-Line Interface User Guide*
- [6] *IPWorks Alarm List*
- [7] *Create Measurement Collection Job*
- [8] *Create Threshold Monitoring Job*
- [9] *List Performance Management Groups and Measurement Types*
- [10] *List Performance Management Jobs*
- [11] *Delete Threshold Monitoring Job*
- [12] *Delete Measurement Collection Job*