NOKIA

5529 Statistics and Data Collector, Release 9.6.07

5529 SDC Northbound Interface Guide

3JL-06001-BRAA-PWZZA

Issue: 01

August 2018

Table of contents

List of figures

List of tables

List of procedures

1 Preface

The *5529 SDC Northbound Interface Guide* describes the NBI component of the 5529 SDC, including XML NBI operations, and structure of the SOAP XML messages.

1.1 Related documentation

Table 1 lists other related documentation sources that you may need to reference.

Table 1   **Related documentation**

Customer documentation	Description
5529 Statistics and Data Collector	
5529 Enhanced Applications Release Notice	Provides an overview of the contents of the 5529 SDC software load, including known restrictions
5529 SDC Installation and User Guide	Provides information about using the 5520 AMS client to perform statistics data collection tasks
5529 SDC Northbound Interface Guide	Provides information about the 5529 SDC NBI component
5529 SDC NE support plug-in parameters guides	Provides information about the full set of NE object parameters such as current and historical PM interval counters, operational and inventory counters, that are supported by the 5529 SDC
Schema documentation provided with the 5529 SDC	Provides schema files and description of schema structures supported by the NBI. See the *5529 Enhanced Applications Release Notice* for information about how to access these files.
5520 AMS and 5529 Enhanced Applications Alarm Search Tool	Provides information about the 5529 Enhanced Applications Alarms.
5520 AMS and 5529 Enhanced Applications Privacy Considerations	Provides information on the product features that impact privacy and the measures taken to protect such data.
5520 Access Management System	
5520 AMS Solution Planning Guide	Provides information about the system requirements for the installation of the 5520 AMS server and client
5520 AMS Installation and Migration Guide	Provides information about how to install, optimize, and uninstall the 5520 AMS server, client, and plug-in components, as well as how to migrate data to the 5520 AMS from other EMSs
5520 AMS Administrator Guide	Provides information about administrative functions, including management of server-client communication, users, NE communication, and schedules

(1 of 2)

Customer documentation	Description
NE support plug-in release notes	Provide NE-specific information about the 5529 IDM NE support plug-ins

(2 of 2)

1.2   Conventions used in this guide

Table 2 lists the conventions that are used in this guide.

Table 2   **Documentation conventions**

Convention	Description	Example
Italics	Identify a variable	*hostname*
Key+Key	Type the appropriate consecutive keystroke sequence.	CTRL+G
Key–Key	Type the appropriate simultaneous keystroke sequence.	CTRL–G
↵	Press the Return key.	↵
—	An em dash in a table cell indicates that there is no information.	—
→	A right arrow graphic following the menu label indicates that a cascading submenu results from selecting a menu item.	File→Save

1.2.1   Important information

The following conventions are used to indicate important information.

Warning — Warning indicates that the described activity or situation may, or will, cause equipment damage or serious performance problems.

Caution — Caution indicates that the described activity or situation may, or will, cause service interruption.

Note — A note provides information that is, or may be, of special interest.

1.2.2   Procedures with options or substeps

When there are options in a procedure, they are identified by letters. When there are substeps in a procedure, they are identified by roman numerals.

Procedure 1   Example of options in a procedure

At step 1, you can choose option a or b. At step 2, you must do what the step indicates.

1   This step offers two options. You must choose one of the following:

    a   This is one option.

    b   This is another option.

2   You must perform this step.

Procedure 2   Example of substeps in a procedure

At step 1, you must perform a series of substeps within a step. At step 2, you must do what the step indicates.

1   This step has a series of substeps that you must perform to complete the step. You must perform the following substeps:

    i   This is the first substep.

    ii   This is the second substep.

    iii   This is the third substep.

2   You must perform this step.

1.3   Multiple PDF file search

You can use Adobe Reader, Release 6.0 or later, to search multiple PDF files for a term. Adobe Reader displays the results in a display panel. The results are grouped by PDF file. You can expand the entry for each file.

Note — The PDF files in which you search must be in the same folder.

Procedure 3   To search multiple PDF files for a term

1   Open the Adobe Reader.

2   Choose Edit→Search from the Adobe Reader main menu. The Search panel opens.

3   Enter the term to search for.

4   Select the All PDF Documents in radio button.

5   Choose the folder in which to search using the drop-down menu.

6   Select the following search criteria, if required:

- Whole words only
- Case-Sensitive
- Include Bookmarks
- Include Comments

7   Click on the Search button.

Adobe Reader displays the search results. You can expand the entries for each file by clicking on the + symbol.

Note — After you click on a hyperlink, you can right-click and choose Previous View from the contextual menu to return to the location of the hyperlink that you clicked on.

1.4   Contact information

If you have questions or comments about this documentation, contact:
documentation.feedback@nokia.com

Getting started

2 What's new

2.1   What's new in Release 9.6.07

Table 3 lists the new 5529 SDC features and enhancements added to the *5529 SDC Northbound Interface Guide* for Release 9.6.07.

Table 3   **What's new in Release 9.6.07**

Feature/enhancement	Description	See
New features/enhancements		
nbi.log file updates	Added information about the nbi.log file, including the following new log file keywords: SourceIP and X-Forwarded-For	Section 4.6
getRelatedObjectsPerformanceMonitoringData operation	Added the support for new operation.	Table 10, Sections 4.3, 5.1, 5.3, 5.6, and 5.7
Privacy Considerations	Included the *5520 AMS and 5529 Enhanced Applications Privacy Considerations* document to the related documentation section.	Section 1.1
Documentation changes		
SOAP 1.2	Included SOAP 1.2 to the list of supported technologies and standards.	Section 3.6
Alarm event notifications	Added a note about how the 5529 SDC NBI user that is subscribed to JMS event notifications receives alarm notifications for all of the NEs that are managed by the 5520 AMS	Section 4.7

2.2   What's new in Release 9.6.05

Table lists the new 5529 SDC features and enhancements added to the *5529 SDC Northbound Interface Guide* for Release 9.6.05.

Table 4   **What's new in Release 9.6.05**

Feature/enhancement	Description	See
New features/enhancements		
getPerformanceMonitoringDataForObjects operation	Added the support for new operation.	Table 10, Sections 4.3, 5.1, 5.3, 5.5, 5.7, 5.8, and 6.1
Health check support	Added a overview section for 5529 Enhanced Applications health check and described getSystemHealthInfo operation.	Section 5.2

2.3   What's new in Release 9.6.03

Table lists the new 5529 SDC features and enhancements added to the *5529 SDC Northbound Interface Guide* for Release 9.6.03.

Table 5   **What's new in Release 9.6.03**

Feature/enhancement	Description	See
New features/enhancements		
Configure the 5520 AMS so that all defaults are secure	The URL for schema documentation is updated with HTTPS as default.	Sections 3.3 and 5.2
Documentation changes		
getPerformanceMonitoringData operation	Added a note for the supported port number and provided reference for the number of ports supported for an NE card.	Section 5.2
	Updated the description for pmParemeterValue element and EXCPT_ENTITY_NOT_FOUND exception.	Tables 20 and 29
	Updated information about PMIS_Unavailable value for the pmParameterStatus attribute in the getPerformanceMonitoringData response and the exception.	Table 20
getPerformanceMonitoringData operation on SHDSL span	Removed information about passive SHDSL span.	Section 5.2

2.4   What's new in Release 9.6

Table 6 lists the new 5529 SDC features and enhancements added to the *5529 SDC Northbound Interface Guide* for Release 9.6.

Table 6   **What's new in Release 9.6**

Feature/enhancement	Description	See
Documentation changes		
getChildObjectsofType attribute	Added a sub-section about the function of *getChildObjectsofType* attribute.	Section 5.2
5529 SDC Release Notice reference	Changed all occurrences of *5529 SDC Release Notice* to *5529 Enhanced Applications Release Notice*.	Table 1
SDC NBI operation names	Removed "Request" from the following SDC NBI operation names: • getPerformanceMonitoringData • enablePerformanceMonitoringData • disablePerformanceMonitoringData	Chapters 3, 4, and 5
WSDL file name	Modified the WSDL file name	Table 10
getPerformanceMonitoringData	Added the attribute information of *vendorExtensions*.	Table 19
enablePerformanceMonitoringData	Added the attribute information of *pmParemeterMaxIntervals* and *vendorExtensions*. Modified the definition of *propNm* attribute.	Table 25
disablePerformanceMonitoringData	Added the attribute information of *pmParemeterMaxIntervals* and *vendorExtensions*. Added the attribute information of *pmParemeterMaxIntervals* and *vendorExtensions*.	Table 27
NBI user role and NBI operation	Removed the note about the need to configure the correct NBI role for the 5529 SDC NBI operation as this is not applicable in R9.5.90.	Section 5.2
getPerformanceMonitoringData	Removed PARAMETER_MISSING_INTERVAL from the description of pmAdditonalInfo attribute as it is not applicable for this operation.	Table 20
Using object type and name	Added a sub-section for Using object type and name for propNm element.	Section 5.2
Schema documentation	Added a note about opening two different schema documentation in the same web browser.	Section 3.3

2.5   What's new in Release 9.5

Table 7 lists the new 5529 SDC features and enhancements added to the *5529 SDC Northbound Interface Guide* for Release 9.5.

Table 7   **What's new in Release 9.5**

Feature/enhancement	Description	See
Documentation changes		
SDC NBI operations	Aligned the SDC NBI operation names with the schema documentation.	Sections 4.3 and 5.2 Table 10
Logs	Log directory name changed to $AMS_DEBUG_DIR	Section 4.6
Linked and child objects	Added information about linked and child objects.	Tables 18 and 19
NBI user role and NBI operation	Added information about the need to configure the correct NBI role for the 5529 SDC NBI operation to succeed.	Section 5.2
-passive value in ports	Added a note about how to recognize a passive SHDSL span in a port.	
mdNm, meNm, propNm parameters are mandatory for getPerformanceMonitoringDataRequest request	Added a note mentioning mdNm, meNm, propNm parameters are mandatory for getPerformanceMonitoringDataRequest request.	Table 19
pmParemeterMaxIntervals	Modified the maximum PM intervals to be retrieved value of *pmParemeterMaxIntervals* element.	Table 19
Order of parameters in the response table of operations	Added a note about the order of parameters in the response table of operations.	Section 5.2
EXCPT_UNABLE_TO_COMPLY	Added information about the exception returned when the NE is not supervised.	Table 29

2.6   What's new in Release 9.4

Table 8 lists the new 5529 SDC features and enhancements added to the *5529 SDC Northbound Interface Guide* for Release 9.4.

Table 8   **What's new in Release 9.4**

Feature/enhancement	Description	See
New features/enhancements		
Performance monitoring operation	Added information about retrieving PM data from multiple child objects when performing getPerformanceMonitoringDataRequest operation.	Tables 18 and 19

(1 of 2)

Feature/enhancement	Description	See
Enable or disable performance monitoring counters	Added information about enabling and disabling the explicit start of performance monitoring counters operations.	Tables 14 and 18 Section 5.2

(2 of 2)

2.7   What's new in Release 9.3.10

The 5529 SDC Northbound Interface Guide is a new document in 5529 SDC, Release 9.3.10. In future releases, this chapter will provide tables of the feature and document changes applicable to this guide.

3 Getting started

3.1 Overview of 5529 SDC

The 5529 Statistics and Data Collector is an advanced statistics data collection solution. It collects statistics from access NEs and transfers the data to CSV files. Collection can be controlled for each NE family. The 5529 SDC ensures that the overall load to the NEs is optimally balanced.

Table 9   **Key functions of 5529 SDC**

Function	Description
Collection configuration	Defining the statistics to collect for an NE family, and the frequency of collection. The collection can be configured: • From the 5520 AMS GUI: It is recommended that you use the GUI to create and modify the configuration file. Use the GUI option available in the SDC Configuration Files view to transfer the modified file from the client to the server. • From a configuration file: The administrator configures collection for an OSS by placing a configuration file in a designated directory of the 5520 AMS data server. The file is processed within 5 minutes.
Collection processing	Collecting statistics from NEs and converting it to CSV results files.
Northbound interface	Providing an interface to the OSS application to retrieve real-time statistics for a specific object and parameters. The NBI is independent of NE type and release. It requires a valid SDC NBI license. Only a user with the SDC NBI-related security functions and roles can perform SDC NBI operations. The SDC NBI provides online schema documentation.

Before you start using the 5529 SDC, ensure that:

- The 5529 SDC is properly installed and licensed on the 5520 AMS server.
- The NEs in your network are supervised by the 5520 AMS. See the Operations and Maintenance guides for the NEs present in your network.

3.2   Overview of the 5529 SDC NBI

The 5529 SDC NBI component facilitates the integration of 5529 SDC data reports into OSS client applications.

The 5529 SDC architecture is based on a web service interface over HTTP/S, which allows OSS client applications to send requests and receive responses with current system data. This interface facilitates the retrieval of specific real-time performance counter values to the OSS for troubleshooting or monitoring purposes.

3.3   Support files

The 5529 SDC NBI is delivered with a set of support files to help you develop client applications. The support files contain sample code for an HTTP/S client application. For information about downloading and installing the support files, see the *5520 AMS Installation and Migration Guide*.

The schema documentation page provides SOAP XML samples of the schema structure of performance monitoring operations.

You need the 5529 SDC WSDL and XSD schema source files to design the OSS client applications. The schema documentation for the activated 5529 SDC is available at:

https://*host*:8443/sdc/services

where *host* is the IP address or host name of the application server.

If you are using HTTP, enter http://*host*:8080/sdc/services

> **Note —** You cannot open two different NBI schema documentation pages in the same web browser. When attempting to do so, an error, "File not found," is displayed.

If you want to open two different NBI schema documentation at the same time, use one of the following options:

- Use different web browsers for different schema documentation.
- Use a user id which has the AMS NBI and all the 5529 Enhanced Applications NBI roles.
- Each time you refresh the web page or when you navigate to the WSDL link of a different schema that was opened before the current schema documentation in the same browser, you need to clear the browser private data, specifically website login information. Different browsers perform this function differently, and use different terminology to refer to private data, such as cache, cookies, and active logins. Consult information about the browser to determine how to perform this function.

Log in to the services page with the username and password of a 5529 SDC user with the SDC NBI role. You can download the schema files individually or as a zip file. Table 10 describes the locations of the WSDL source files.

Table 10   **WSDL location in schema files**

Web services operation	WSDL file	Location
getPerformanceMonitoringData or getPerformanceMonitoringDataForObjects or getRelatedObjectsPerformanceMonitoringData	PerformanceManagementRetrievalExtnsSOAP.wsdl	schema/alu/wsdl/binding/soap_http/PerformanceManagementRetrievalExtnsSOAP.wsdl
enablePerformanceMonitoringData or disablePerformanceMonitoringData	PerformanceManagementControlExtnsSOAP.wsdl	schema/alu/wsdl/binding/soap_http/PerformanceManagementControlExtnsSOAP.wsdl

3.4   Web services

This section provides general information about the web services supported by the 5529 SDC NBI architecture. The 5529 SDC architecture supports the performance monitoring operations performed by OSS client applications. A web services server is the interface between the OSS and the 5529 SDC. The supported operations on the 5529 SDC NBI are synchronous RPC-type transactions, in which the data types and the request/response messages are defined.

The OSS client applications and the 5529 SDC use SOAP as the web service messaging format for exchanging XML-based messages over HTTP/S. The web services operate as defined in the WSDL files, which are structured XML documents. Schema files describe the structure and elements of the SOAP XML messages. The messages are formatted according to the MTOSI 1.1 standard requirements.

To send requests to the 5529 SDC using web services, an OSS client application must specify a username and password as part of the HTTP/S header.

3.4.1   Information model

To perform web service operations on the 5529 SDC NBI, you need to create OSS client applications. An OSS client application incorporates an operation request wrapped in a SOAP XML envelope. The 5529 SDC receives the request, performs the operation, and sends a SOAP XML response message that contains the response of the operation to the OSS.

The SOAP envelope for the XML request/response includes information about the schema that is used to describe the information in the message, and the envelope elements described in Table 11.

Table 11   **SOAP envelope elements**

Element	Description
Header	The SOAP envelope header defines the XML Solution Set (TMF 854) supported by MTOSI Release 1.1, as well as the version of the supported interface.
Body	The SOAP envelope body carries all the information related to the associated operation and filter.

See the operations chapter for more information about the SOAP XML request and response messages supported by the 5529 SDC NBI operations.

3.5   SOAP envelope components

Figure 1 shows the types of SOAP envelopes and their content.

Figure 1   **SOAP envelope types**

XML request

| SOAP envelope |
| Schema info |
| SOAP header |
| User security info |
| Sequence info |
| SOAP body |
| XML request |
| XML filter (optional) |

XML response

| SOAP envelope |
| Schema info |
| SOAP header |
| Response info |
| SOAP body |
| XML response |
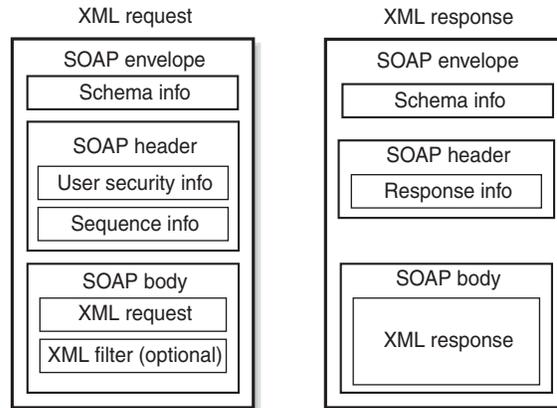
3.5.1   SOAP XML header

Table 12 describes the elements used in the SOAP envelope header.

The elements used in the SOAP envelope body vary by operation. See the operations chapter for more information.

Table 12   **SOAP envelope header**

Element	Description	Values
activityName	Operation name	*Operation name*, for example, getPerformanceMonitoringData
msgName	Message name in the WSDL file	*Operation name* *Operation name*Response
msgType	Message type	REQUEST RESPONSE ERROR
senderURI	Application sending the message	String
destinationURI	Destination for the message	String
activityStatus	Status of the response operation This element appears in response messages only.	SUCCESS FAILURE WARNING
communicationPattern	Message communication pattern	SimpleResponse
communicationStyle	Message communication style	RPC

(1 of 2)

Element	Description	Values
timestamp	Time when the message was created This element appears in response messages only.	Time in MTOSI format: yyyyMMddhhmmss.s[Z|{+|-}HHMm]

(2 of 2)

3.6   Technologies and standards

The 5529 SDC software architecture is built on open interoperable technologies such as SOAP and XML, the Java and J2EE framework, multi-tier layering, and web service interfaces. The use of standard interfaces allows the 5529 SDC to integrate with OSS client applications.

To create OSS client applications that interface with the 5529 SDC, you need to be familiar with the following technologies and standards:

- XML 1.0
- XML Schema (XSD) 1.0
- SOAP 1.1/1.2
- WSDL 1.1

- HTTP/S 1.1
- MTOSI 1.1
- TMF OSS interfaces (TMF 854)

See the following websites for more information:

- http://www.w3.org/TR/2004/REC-xml-20040204 (W3C standards about XML 1.0, third edition)
- http://www.w3.org/TR/2001/REC-xmlschema-1-20010502 (XSD)
- http://www.w3.org/TR/2000/NOTE-SOAP-20000508 (SOAP 1.1)
- https://www.w3.org/TR/soap12/ (SOAP 1.2)

- http://www.w3.org/TR/2001/NOTE-wsdl-20010315 (WSDL 1.1)
- http://www.ietf.org/rfc/rfc2616.txt (HTTP/S 1.1)
- http://www.tmforum.org (TMF and MTOSI)

5529 SDC NBI Overview

4 5529 SDC Northbound Interface

4.1 Overview

This chapter describes the MTOSI object names, licensing requirements, 5529 SDC NBI roles and functions, and the number of concurrent operations allowed.

4.2 MTOSI object names

The SOAP XML body defines each MTOSI and Nokia-proprietary name as a sequence of name components, where each component is in a position relative to the parent element. Each XML element name corresponds to one of the components of the hierarchical name, and the element value contains the value of the specified hierarchical name component.

Table 13   **Northbound object names**

Object	Naming convention
Management domain	<mdNm>MD name</mdNm>
Network element	<mdNm>MD name</mdNm> <meNm>NodeName</meNm>

(1 of 2)

Object	Naming convention
Managed object	<mdNm>MD name</mdNm> <meNm>NodeName</meNm> <propNm>/type=friendly object type/friendly object name</propNm> For example: <name> <mdNm>5520 AMS</mdNm> <meNm>gpon17</meNm> <propNm>/type=POTS Port/R1.S1.LT7.PON1.ONT15.C1.P1</propNm> </name>

(2 of 2)

4.3   Operation constraints

The default configuration of the maximum number of concurrent getPerformanceMonitoringData, getPerformanceMonitoringDataForObjects or getRelatedObjectsPerformanceMonitoringData operations allowed is 10. The default configuration of the maximum number of parameters allowed in one operation is 20.

Contact your local Nokia representative to change the configuration of the maximum number of concurrent operations allowed or maximum number of parameters allowed in one operation.

4.4   Licensing

The 5529 SDC NBI has its own license key, which is generated by the Nokia Software Licensing tool. You need to add the 5529 SDC NBI license key to the 5520 AMS licensing management system in the Administration perspective. See the 5520 AMS Administrator Guide for information about creating licenses.

4.5   5529 SDC NBI roles and functions

The 5529 SDC relies on the 5520 AMS security and user management mechanism to authenticate users.

The 5520 AMS user authentication/authorization mechanism is based on roles and functions. Users can execute certain tasks or access specific functionality depending on the roles and functions granted to them.

Table 14 describes the 5529 SDC NBI function and role.

Table 14   **5529 SDC NBI function and role**

Function	Description	Role
SDC NBI - View	This function enables NBI OSS client applications to retrieve SDC NBI performance monitoring operations.	Administrator SDC NBI
SDC NBI - Edit	This function enables or disables the explicit start of performance monitoring counters operations.	Administrator SDC NBI

See the *5520 AMS Administrator Guide* for information about managing user roles and functions.

4.6   Logs

The 5529 SDC NBI log information is copied to the following files in the $AMS_DEBUG_DIR directory:

- nbi.log
- server.log

Note — In a cluster deployment, log information is recorded on the server where the authentication is done.

4.6.1   nbi.log

The nbi.log file contains 5529 SDC NBI-specific messages and JMS client connection, disconnection, and authentication failure messages.

The 5529 SDC NBI logs information about the result of operation requests sent by OSS client applications to the nbi.log file. Each new operation request adds a new entry to the nbi.log. Each log entry starts with a timestamp that indicates when the operation was completed or failed, and contains keywords that identify information blocks about the operation result. Two adjacent information blocks are delimited by a comma.

Table 15   **nbi.log file keywords**

Keyword	Description
INFO	JBOSS log level information

Keyword	Description
Operation:	Name of the supported operation that was sent in the request. Examples: exportNetwork, getInventory, getSystemInfo
SourceIP:	Source IP address of the OSS client (or load balancer) that sent the operation request. When the SourceIP information is present in the log, the X-Forwarded-For information is not present in the log.
X-Forwarded-For:	Original source IP address of the OSS client that is sent the operation request. The X-Forwarded-For is an HTTP header field that is used to identify the originating IP address of an OSS client that originated the request through an HTTP proxy or load balancer. When the X-Forwarded-For information is present in the log, the SourceIP information is not present in the log.
Username:	Name of the OSS client application that sent the operation request
Result:	Result of the operation. Example: Success or Failed
ErrorCode:	Error code if the operation result is Failed. The error code in the nbi.log is the same error code that is provided in the operation response.
ErrorMessage:	User-friendly description of the error if the operation result is Failed

The following table provides log example for a NBI operation:

Table 16   **Log example**

Operation name	Log example
getPerformanceMonitoringData	2017-11-17 14:22:19,466 INFO [default task-3] [SDC-NBI] Operation:getPerformanceMonitoringData,SourceIP:135.249.43.12,Username:nbiClient,Result:Success

For more information about the nbi.log file, see the *5520 AMS Administrator Guide.*

4.6.2   server.log

The server.log file contains communication messages exchanged with the 5520 AMS, internal error messages, and details about processing events from the 5520 AMS.

For more information about the server.log file, see the *5520 AMS Administrator Guide.*

4.6.3   Viewing logs

You can view the logs in the $AMS_DEBUG_DIR directory from the 5520 AMS GUI or on the server. For information about viewing logs from the GUI, see the *5520 AMS Administrator Guide.*

4.6.4   Setting tracing level

Contact your local Nokia representative to set the tracing level for logs.

4.7   Event notification

Note — The 5529 SDC NBI user that is subscribed to JMS event notifications receives alarm notifications for all of the NEs that are managed by the 5520 AMS. The user does not receive alarm notifications only for NEs that have the same PAP group as the user.

5   Operations

5.1   Overview

The 5529 SDC NBI component provides a performance monitoring interface accessible to OSSs.

Table 17 provides information about the supported performance monitoring interfaces for getPerformanceMonitoringData, getPerformanceMonitoringDataForObjects and getRelatedObjectsPerformanceMonitoringData operations.

Table 17   **Supported performance monitoring interfaces**

Operation	The interface supports up to:
getPerformanceMonitoringData	10,000 operations per hour
getPerformanceMonitoringDataForObjects	10,000 objects per hour
getRelatedObjectsPerformanceMonitoringData	10,000 related objects per hour

5.2   5529 Enhanced Applications health check overview

The getSystemHealthInfo operation allows the OSS client application (and load balancers in a cluster deployment) to determine the availability of the 5529 Enhanced Application NBI server. If the NBI server is available, it is able to process NBI operation requests and successfully provide operation responses. The following information is applicable for getSystemHealthInfo operation:

- The operation does not have any parameters for request and response
- An OSS or a load balancer can check the HTTP error code to determine if the NBI is available

The getSystemHealthInfo operation will check if the operation can access the database. If the database connection cannot be established, it is not possible for the getSystemHealthInfo operation to authenticate the user. In this case, an exception is returned with HTTP code 401.

5.3   Performance monitoring operation

Table 18 describes the performance monitoring operation supported by the 5529 SDC NBI component. All operation requests and responses are structured as SOAP XML envelopes. See the schema documentation for more information.

Note — The order of parameters in the tables where request SOAP XML envelope bodies are described may not appear in the same order as in the actual SOAP XML response received by an OSS. See the schema documentation for the correct order of parameters.

Table 18 **Supported performance monitoring operation**

Operation	Description	Request	Response
getPerformanceMonitoringData	The operation returns real-time PM data and latest intervals on a given object for given parameters. Also, the operation collects data for the child object[1] and the linked object[2] of a given object in a single request. See getPerformanceMonitoringData SOAP XML envelope body.	The request contains the target object identifier, and list of parameter names (and maximum intervals) to be retrieved.	If the operation succeeds, it returns the list of requested parameters, and for each parameter, its retrieval status, and interval values or additional information of reason for failure, where applicable. If the operation fails, it returns a ProcessingFailureException_T with an exception type and a message that indicates the reason of the failure. For more information, see Table 29.
getPerformanceMonitoringDataForObjects	The operation returns real-time PM data and latest intervals on the given multiple objects for given parameters. Also, the operation collects data for the child object[1] and the linked object[2] of the multiple objects in a single request. See getPerformanceMonitoringDataForObjects SOAP XML envelope body.	The request contains the target object identifiers, and list of parameter names (and maximum intervals) per object identifier to be retrieved.	If the operation succeeds, it returns the list of requested parameters per object identifier, and for each parameter, its retrieval status, and interval values or additional information of reason for failure, where applicable. If the operation fails, it returns a ProcessingFailureException_T with an exception type and a message that indicates the reason of the failure. For more information, see Table 29.
getRelatedObjectsPerformanceMonitoringData [3] [4]	The operation returns real-time PM data and latest intervals on related objects for given parameters. When retrieving related objects of a given object, a filter can be specified to restrict the number of related objects to be returned. See getRelatedObjectsPerformanceMonitoringData SOAP XML envelope body.	The request contains the object identifier of a parent object of the related object type, the related object type of interest and list of parameter names (and maximum intervals) to be retrieved for each related object found.	If the operation succeeds, it returns the list of requested parameters per related object identifier, and for each parameter, its retrieval status, and interval values. If the operation fails, it returns a ProcessingFailureException_T with an exception type and a message that indicates the reason of the failure. For more information, see Table 29.

(1 of 2)

Operation	Description	Request	Response
enablePerformanceMonitoringData	The operation enables explicit start of the PM counters on a given object.	The request contains the target object identifier and list of parameter names to enable or disable PM counters.	If the operation fails, it returns a ProcessingFailureException_T with an exception type and a message that indicates the reason of the failure. For more information, see Table 29.
disablePerformanceMonitoringData	The operation disables explicit start of the PM counters on a given object.		

(2 of 2)

Notes

(1) An object is a child of another object, if they are on the same object identifier path. For example, a slot is a child of rack.

(2) An object (b) is linked to an object (a) if the object identifier of (b) is an attribute of object (a). For example, profile *x* is linked to a port, the object identifier of profile *x* is an attribute of that port.

(3) getRelatedObjectsPerformanceMonitoringData operation is applicable only for VLAN Associations on non-ATM ports. For the list of supported parent objects, refer to Section 5.6.

This operation is not supported for ports supporting vpi or vci.

(4) getRelatedObjectsPerformanceMonitoringData operation is supported only on ISAM R4.4 and above, and GPON R4.7 and above.

5.4   getPerformanceMonitoringData SOAP XML envelope body

Table 19 describes the elements used in the getPerformanceMonitoringData SOAP envelope body of a request message. See section 3.5 for information about the elements that are present in the header of request and response messages.

Note 1 — The maximum port number supported is 256. If you use a value greater than 256, the value will be converted to "value mod 256" internally by 5520 AMS. For example, 257 becomes 1, 258 become 2, and so on. Do not use any value greater than 256 for a port number.

Note 2 — For information about the number of ports supported for an NE card, see the relevant Unit Data Sheets (UDS) of the NE.

Table 19   **getPerformanceMonitoringData request SOAP XML envelope body**

Element	Description	Required	Value
pmObjectSelect	Object identifier of the target object for PM data to be retrieved.	Mandatory	NamingAttributes_T
NamingAttributes_T elements			
mdNm[1]	Management domain name	Mandatory	String. For example, <mdNm>AMS</mdNm>

(1 of 2)

Element	Description	Required	Value
meNm[1]	Network element name	Mandatory	String. For example, <meNm>gpon45</meNm>
propNm[1]	Friendly object type and name (For more information on the format of the 5520 AMS friendly object type and object identifier, see the 5529 SDC Parameters Guide for the NE in the supported object parameter's section of the target object.)	Mandatory	String. For example, <propNm>/type=POTS Port/R1.S1.LT7.PON1.ON T15.C1.P1</propNm>
pmParameterList	List of PM parameters to be retrieved	Mandatory	PerformanceMonitoringPar ameterList_T
PerformanceMonitoringParameterList_T elements			
pmParameter	One or more instances of a member PM parameter	Mandatory	PerformanceMonitoringPar ameter_T
PerformanceMonitoringParameter_T			
pmParameterName	PM parameter name to be retrieved (For more information on the PM parameter name, see the 5529 SDC Parameters Guide for the NE in the supported object parameter's section of the target object.). The attribute name of a linked or child object (with the association of one to one or many to one) contains its object name and attribute name separated by a delimiter. Only one level of linked object is supported. If the child object is not within the hierarchy of the parent object (for example, a Bridgeport within a physical port), the object must be retrieved in a separate operation.	Mandatory	string. For example, <sdc:pmParameterName>/ Type=ONT Card/C1::bponOntActualC ardType</sdc:pmParamete rName>
pmParemeterMaxIntervals [2]	Maximum PM intervals to be retrieved (default=1, maximum=288)	Optional	integer
vendorExtensions[3]	Container element for additional user parameters.	No	PackageList_T

(2 of 2)

Note

[1]   Although parameters *mdNm*, *meNm*, and *propNm* are optional in the schema, they are mandatory for this operation.

[2]   The pmParemeterMaxIntervals attribute is applicable only for PM Interval Counters.

[3]   The attribute *vendorExtensions* is not used in this operation.

Table 20 describes the elements used in the getPerformanceMonitoringData SOAP envelope body of a response message.

Table 20   **getPerformanceMonitoringData response SOAP XML envelope body**

Element	Description	Required	Value
pmDataList	Performance monitoring returned data list	Mandatory	PerformanceMonitoringD ataList_T
PerformanceMonitoringDataList_T elements			

(1 of 2)

Element	Description	Required	Value
pmData	One or more instances of a member PM data	Mandatory	PerformanceMonitoringData_T
PerformanceMonitoringData_T elements			
pmParameterName	PM parameter name of the data point	Mandatory	string
pmParemeterValue	• PM values (Comma-separated if more than one value is returned); may not be present for invalid/unavailable parameters • Instance Unavailable - It will be returned when no value is available for the parameter. Note: 5529 SDC supports up to 32 intervals (although some tables support 192 intervals).	Optional	string
pmParameterStatus	Measurement parameter status. Allowed values: • PMIS_Valid - Valid data • PMIS_Invalid - Data available but marked as invalid for the interval • PMIS_Unavailable - No data available PMIS_Unavailable value is returned when the port value is greater than the supported value but smaller than 256.	Mandatory	string (enum)
pmAdditonalInfo	Additional parameter information, used for "PMIS_Unavailable" or "PMIS_Invalid" parameter values. Valid values: • PARAMETER_NOT_FOUND • PARAMETER_NOT_SUPPORTED • PARAMETER_INVALID_INTERVAL • PARAMETER_NULL_VALUE This field is not returned for valid parameters.	Optional	string

(2 of 2)

If the attribute provided in the pmParameterName of getPerformanceMonitoringData request does not exist (example, if the parameter name is incorrect, the parameter is not enabled for collection in the NE), in the response pmParameterStatus is set to PMIS_Unavailable, and pmAdditionalInfo is set to PARAMETER_NULL_VALUE.

Example of getPerformanceMonitoringData response:

```
<soapenv:Body>

 <sdc:GetPerformanceMonitoringDataResponse xmlns:sdc="sdcNbi">

   <sdc:pmDataList>

     <sdc:pmData>

       <sdc:pmParameterName>eqptBoardAdminStatus1</sdc:pmParameterName>

       <sdc:pmParameterStatus>PMIS_Invalid</sdc:pmParameterStatus>
```

```
        <sdc:pmAdditionalInfo>PARAMETER_NOT_FOUND</sdc:pmAdditionalInfo>

      </sdc:pmData>

    </sdc:pmDataList>

  </sdc:GetPerformanceMonitoringDataResponse>

</soapenv:Body>
```

5.5 getPerformanceMonitoringDataForObjects SOAP XML envelope body

Table 21 describes the elements used in the getPerformanceMonitoringDataForObjects SOAP envelope body of a request message. See section 3.5 for information about the elements that are present in the header of request and response messages.

Table 21   **getPerformanceMonitoringDataForObjects request SOAP XML envelope body**

Element	Description	Required	Value
pmObjectSelect	Object identifiers of the target object for PM data to be retrieved.	Mandatory	NamingAttributes_T
NamingAttributes_T elements and PerformanceMonitoringParameterList_T elements [1]			

Notes
[1]   For "NamingAttributes_T elements" and "PerformanceMonitoringParameterList_T elements", refer to Table 19.

Example of getPerformanceMonitoringDataForObjects request:

```
<soapenv:Body>

<sdc:GetPerformanceMonitoringDataForObjectsRequest>

<sdc:pmInputList>

<sdc:pmObjectSelect extVersion="?" extAuthor="?" tmf854Version="?">

<tmf:mdNm>AMS</tmf:mdNm>

<tmf:meNm>ISAM100</tmf:meNm>

<tmf:propNm>/Type=SHDSL Span/R1.S1.LT8.P1</tmf:propNm>

</sdc:pmObjectSelect>

<sdc:pmParameterList>

<sdc:pmParameter>
```

```
<sdc:pmParameterName>atmInterfaceMaxActiveVciBits</sdc:pmParameterName>

<sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

</sdc:pmParameter>

<sdc:pmParameter>

<sdc:pmParameterName>atmInterfaceMaxVccs</sdc:pmParameterName>

<sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

</sdc:pmParameter>

</sdc:pmParameterList>

</sdc:pmInputList>

</sdc:GetPerformanceMonitoringDataForObjectsRequest>

</soapenv:Body>
```

Table 22 describes the elements used in the getPerformanceMonitoringDataForObjects SOAP envelope body of a response message.

Table 22 **getPerformanceMonitoringDataForObjects response SOAP XML envelope body**

Element	Description	Required	Value
pmDataList	Performance monitoring returned data list	Mandatory	PerformanceMonitoringDataList_T
PerformanceMonitoringDataList_T elements [1]			
PerformanceMonitoringData_T elements [1]			
pmAdditonalInfo	Additional parameter information, used for "PMIS_Unavailable" or "PMIS_Invalid" parameter values. Valid values: • PARAMETER_NE_NOT_FOUND For other valid values, refer to pmAdditonalInfo element in Table 20.	Optional	String

Notes
[1]   For "PerformanceMonitoringDataList_T elements" and "PerformanceMonitoringData_T elements" refer to Table 20.

Example of getPerformanceMonitoringDataForObjects response:

```
<soapenv:Body>

<sdc:GetPerformanceMonitoringDataForObjectsResponse xmlns:sdc="sdcNbi">

<sdc:pmDataListForObjects>

<sdc:pmObjectSelect>
```

```
<tmf:mdNm>AMS</tmf:mdNm>

<tmf:meNm>ISAM100</tmf:meNm>

<tmf:propNm>/Type=SHDSL Span/R1.S1.LT8.P1</tmf:propNm>

</sdc:pmObjectSelect>

<sdc:pmDataList>

<sdc:pmData>

<sdc:pmParameterName>atmInterfaceMaxVccs</sdc:pmParameterName>

<sdc:pmParameterValue>8</sdc:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

<sdc:pmData>

<sdc:pmParameterName>atmInterfaceMaxActiveVciBits</sdc:pmParameterName>

<sdc:pmParameterValue>16</sdc:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

</sdc:pmDataList>

</sdc:pmDataListForObjects>

</sdc:GetPerformanceMonitoringDataForObjectsResponse>

</soapenv:Body>
```

5.6   getRelatedObjectsPerformanceMonitoringData SOAP XML envelope body

Table 23 describes the elements used in the getRelatedObjectsPerformanceMonitoringData SOAP envelope body of a request message. See section 3.5 for information about the elements that are present in the header of request and response messages.

Table 23   **getRelatedObjectsPerformanceMonitoringData request SOAP XML envelope body**

Element	Description	Required	Value
pmObjectSelect	Object identifier of a relative (for example, parent or an ancestor) of the related object for which the PM data to be retrieved.	Mandatory	NamingAttributes_T

(1 of 2)

Element	Description	Required	Value
NamingAttributes_T elements [1]			
relatedObjectType	The child or linked object type.	Mandatory	String. For example, <sdc:relatedObjectType>VLAN Association</sdc:relatedObjectType>
pmParameterList	List of PM parameters to be retrieved	Mandatory	PerformanceMonitoringParameterList_T
PerformanceMonitoringParameterList_T elements [1]			
vendorExtensions [2]	Container element for additional user parameters.	No	PackageList_T

(2 of 2)

Note

[1]   For "NamingAttributes_T elements" and "PerformanceMonitoringParameterList_T elements", refer to Table 19.

[2]   A filter is specified as a package named "pmRelatedObjectFilter" in the vendorExtensions of the getRelatedObjectsPerformanceMonitoringData request. The only supported filter criteria is the "relativeRelatedObjectId" and for a VLAN Association it consists of either:

- "S<sVlanId>"
- "C<cVlanId>" or
- "S<sVlanId>.C<cVlanId>" pairs

Also, only one filter criteria is supported in the "pmRelatedObjectFilter".

Example of getRelatedObjectsPerformanceMonitoringData request:

```
<soapenv:Body>

<sdc:GetRelatedObjectsPerformanceMonitoringDataRequest>

<sdc:pmObjectSelect>

<tmf:mdNm>AMS</tmf:mdNm>

<tmf:meNm>ISAM</tmf:meNm>

<tmf:propNm>/type=NGPON2 UNI/CG1.SCG1.ONT1.C1.P1</tmf:propNm>

</sdc:pmObjectSelect>

<sdc:relatedObjectType>VLAN Association</sdc:relatedObjectType>

<sdc:pmParameterList>

<!--Instant counters-->

<sdc:pmParameter>

<sdc:pmParameterName>VLANPortAssociationIfAdminPseudo</sdc:pmParameterName>

<sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

</sdc:pmParameter>
```

```
<sdc:pmParameter>

<sdc:pmParameterName>VLANTranslationPseudoAttribute</sdc:pmParameterName>

<sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

</sdc:pmParameter>

</sdc:pmParameterList>

<sdc:vendorExtensions>

<sdc:package name="pmRelatedObjectFilter">

<sdc:NameAndStringValue>

<tmf:name>relativeRelatedObjectId</tmf:name>

<tmf:value>S656</tmf:value>

</sdc:NameAndStringValue>

</sdc:package>

</sdc:vendorExtensions>

</sdc:GetRelatedObjectsPerformanceMonitoringDataRequest>

</soapenv:Body>
```

Table 24 describes the elements used in the getRelatedObjectsPerformanceMonitoringData SOAP envelope body of a response message.

Table 24   **getRelatedObjectsPerformanceMonitoringData response SOAP XML envelope body**

Element	Description	Required	Value
relatedObjectPmDataList	Performance monitoring returned data list for related object identifiers	Mandatory	RelatedObjectPerformanceMonitoringDataList_T
RelatedObjectPerformanceMonitoringDataList_T elements			
relatedObjectPmData	One or more related object identifier instances of a member PM data	Mandatory	RelatedObjectPerformanceMonitoringData_T
RelatedObjectPerformanceMonitoringData_T elements			
relatedObjectName	Name of the related object identifier	Mandatory	String
pmDataList	Performance monitoring returned data list	Mandatory	PerformanceMonitoringDataList_T
PerformanceMonitoringDataList_T elements [1] [2]			
vendorExtensions	Container element for additional user parameters	No	PackageList_T

Notes
[1]   For "PerformanceMonitoringDataList_T elements", refer to Table 20.
[2]   PARAMETER_NOT_FOUND value is returned when the related object type, or the attribute of the related object type requested does not exist, refer to Table 20.

Example of getRelatedObjectsPerformanceMonitoringData response:

```
<soapenv:Body>

<sdc:GetRelatedObjectsPerformanceMonitoringDataResponse xmlns:sdc="sdcNbi"
xmlns:tmf="tmf854.v1">

<sdc:relatedObjectPmDataList>

<sdc:relatedObjectPmData>

<sdc:relatedObjectName>

<tmf:mdNm>AMS</tmf:mdNm>

<tmf:meNm>ISAM</tmf:meNm>

<tmf:propNm>/Type=VLAN Association/P1811939328.S656</tmf:propNm>

</sdc:relatedObjectName>

<sdc:pmDataList>

<sdc:pmData>

<sdc:pmParameterName>extendPortVlanMaxNumMacAddresses</sdc:pmParameterName>

<sdc:pmParameterValue>65535</sdc:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

<sdc:pmParameterName>extendPortVlanNetworkVlan</sdc:pmParameterName>

<sdc:pmParameterValue>2686976</sdc:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

</sdc:pmDataList>

</sdc:relatedObjectPmData>

</sdc:relatedObjectPmDataList>

</sdc:GetRelatedObjectsPerformanceMonitoringDataResponse>

</soapenv:Body>
```

The getRelatedObjectsPerformanceMonitoringData operation supports the following parent objects where the related object type is VLAN Association:

- /type=Ethernet Port/R[n].S[n].LT[n].PON[n].ONT[n].C[n].P[n]
- /type=HPNA Port/R[n].S[n].LT[n].PON[n].ONT[n].C[n].P[n]
- /type=MoCA Port/R[n].S[n].LT[n].PON[n].ONT[n].C[n].P[n]
- /type=VDSL2 Port/R[n].S[n].LT[n].PON[n].ONT[n].C[n].P[n]
- /type=VEIP Port/R[n].S[n].LT[n].PON[n].ONT[n].C[n].P[n]
- /type=Ethernet LT Port/R[n].S[n].LT[n].P[n]
- /type=Ethernet NT Port/R[n].S[n].ACU/NTIO|NT[A|B]|LT.P[n]
- /type=UNI/R[n].S[n].LT[n].PON[n].ONT[n].C[n].P[n]
- /type=NGPON2 UNI/CG[n].SCG[n].ONT[n].C[n].P[n]
- /type=NT Bridge Port/R[n].S[n].NT[A|B].P[n]
- /type=EFM Bridge Port/R[n].S[n].LT[n].P[n]
- /type=LAG Bridge Port/R[n].S[n].LT[n].P[n]
- /type=FE Bridge Port/R[n].S[n].LT[n].P[n]
- /type=FE LT Port/R[n].S[n].LT[n].P[n]
- /type=EPON Ethernet Port/R[n].S[n].LT[n].P[n]
- /type=NGPON2 Ethernet Port/CG[n].SCG[n].ONT[n].C[n].P[n]
- /type=NGPON2 VEIP Port/CG[n].SCG[n].ONT[n].C[n].P[n]

The following parent objects are supported for the non-ATM based ports:

- /type=XDSL Port/R[n].S[n].LT[n].P[n]
- /type=Bonding Group/R[n].S[n].LT[n].P[n]

5.7   Request endpoints

The OSS client applications need to send request messages to specific 5529 SDC service endpoints, depending on the operation.

The endpoint for getPerformanceMonitoringData, getPerformanceMonitoringDataForObjects, and getRelatedObjectsPerformanceMonitoringData is https://host:8443/sdc/services/PerformanceManagementRetrievalExtns.

The endpoint for enablePerformanceMonitoringData and disablePerformanceMonitoringData is https://host:8443/sdc/services/PerformanceManagementControlExtns.

5.8   getPerformanceMonitoringData or getPerformanceMonitoringDataForObjects operation on SHDSL span

This section provides information about using the SDC NBI for SHDSL in a multi-wire configuration.

SHDSL span can be in one of the following state:

- SHDSL spans that are available for configuration are indicated by customerId "available"
- SHDSL spans that are configured displays the Port State as "configured"

The state (admin state and operator status) and bit rate of the EFM bonding group are obtained from the configured SHDSL span. You can access the EFM bonding group through the SHDSL span. The Port State of the configured SHDSL spans of the EFM bonding group is shown as "configured." When configuring a SHDSL span with EFM bonding you must not use the subsequent spans id. The number of span ids that cannot be used is calculated based on the number of pairs in the bonding group minus one. For example, if SHDSL span has 6-wire configuration at P1, then P2 and P3 cannot be used.

To retrieve the state of an SHDSL span in the SDC NBI, from a configured SHDSL span execute the getPerformanceMonitoringData or getPerformanceMonitoringDataForObjects on /type=SHDSLSpan/R[n].S[n].LT[n].P[n] using the following attributes:

shdslSpanAdminStatePseudo

shdslSpanState

> **Note —** The admin state can be only retrieved for the configured SHDSL Span.

Whereas, the port state must be retrieved from the SHDSL span object. For example, if the Line1, Line2, Line3, Line4 are grouped together in a span, then the SHDSL span definition is on R1.S1.LTx.P1.

The following information is retrieved by getPerformanceMonitoringData or getPerformanceMonitoringDataForObjects operation on a SHDSL span with R1.S1.LTx.P1 and if there are four lines with 8-wire configuration:

- Overall SPAN state: shdslSpanState
- Individual Link states at the configured SHDSL Span:
    * shdslSpanOperStateLink1Pseudo
    * shdslSpanOperStateLink2Pseudo
    * shdslSpanOperStateLink3Pseudo
    * shdslSpanOperStateLink4Pseudo

5.9   Function of getChildObjectsofType attribute

The attribute getChildObjectsofType returns all object IDs of the child objects (with the association of one to one or many to one). For getChildObjectsofType you should use a parameter name and must be followed by the name of child object type.

The values returned for function getChildObjectsofType attribute are always enclosed within brackets. For example: [].

Note — It is also possible to use the getChildObjectsofType attribute to retrieve all object IDs under an agent, the agent identifier must be provided in pmObjectSelect.

Following are the list of agent identifiers:

/type=IACM/

/type=IHUB/IHUB

/type=SHUB/SHUB

Following are the request and response examples of getChildObjectsofType attribute for VLANAssociation of ONTEthernetPort.

Request example

```
Request example

<sdc:getPerformanceMonitoringData>

<sdc:pmObjectSelect>

 <tmf:mdNm>ams</tmf:mdNm>

 <tmf:meNm>172.21.176.247 </tmf:meNm>

 <tmf:propNm>/type=Ethernet Port/R1.S1.LT2.PON1.ONT64.C1.P2</tmf:propNm>

</sdc:pmObjectSelect>

<sdc:pmParameterList>
```

```
<sdc:pmParameter>

<sdc:pmParameterName>getChildObjectsofType=VLAN
Association</sdc:pmParameterName>

 <sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

 </sdc:pmParameter>

 </sdc:pmParameterList>

 </sdc:getPerformanceMonitoringData>
```

Response example

```
<soapenv:Body>

<sdc:GetPerformanceMonitoringDataResponse xmlns:sdc="sdcNbi">

<sdc:pmDataList>

<sdc:pmData>

<sdc:pmParameterName>getChildObjectsofType=VLAN
Association</sdc:pmParameterName>

<sdc:pmParameterValue>[P113310785.C333,P113310785.C444,P113310785.C222]</sd
c:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

</sdc:pmDataList>

</sdc:GetPerformanceMonitoringDataResponse>
```

Following are the request and response examples of getChildObjectsofType attribute for XDSL Line Spectrum Profile of IACM Agent.

Request example

```
<sdc:getPerformanceMonitoringData>

<sdc:pmObjectSelect>

<tmf:mdNm>AMS</tmf:mdNm>

<tmf:meNm>172.21.176.247</tmf:meNm>

<tmf:propNm>/type=IACM/</tmf:propNm>

</sdc:pmObjectSelect>

<sdc:pmParameterList>

<sdc:pmParameter>
```

```
<sdc:pmParameterName>getChildObjectsofType:/type=XDSL Line
SpectrumProfile</sdc:pmParameterName>

<sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

</sdc:pmParameter>

</sdc:pmParameterList>

</sdc:getPerformanceMonitoringData>
```

Response example

```
<sdc:GetPerformanceMonitoringDataResponse xmlns:sdc="sdcNbi">

<sdc:pmDataList>

<sdc:pmData>

<sdc:pmParameterName>getChildObjectsofType:/type=XDSL Line
SpectrumProfile</sdc:pmParameterName>

<sdc:pmParameterValue>[4,3,2,1,6,5]</sdc:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

</sdc:pmDataList>

</sdc:GetPerformanceMonitoringDataResponse>
```

Following are the request and response examples of getChildObjectsofType
attribute for MAC Filter of IHUB Agent.

Request example

```
<sdc:getPerformanceMonitoringData>

<sdc:pmObjectSelect>

<tmf:mdNm>AMS</tmf:mdNm>

<tmf:meNm>172.21.176.247</tmf:meNm>

<tmf:propNm>/type=IHUB/IHUB</tmf:propNm>

</sdc:pmObjectSelect>

<sdc:pmParameterList>

<sdc:pmParameter>

<sdc:pmParameterName>getChildObjectsofType=MACFilter</sdc:pmParameterName>

<sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

</sdc:pmParameter>
```

```
</sdc:pmParameterList>

</sdc:getPerformanceMonitoringData>
```

Response example

```
<sdc:GetPerformanceMonitoringDataResponse xmlns:sdc="sdcNbi">

<sdc:pmDataList>

<sdc:pmData>

<sdc:pmParameterName>getChildObjectsofType:/type=MAC

Filter</sdc:pmParameterName>

<sdc:pmParameterValue>[IHUB:2,IHUB:1]</sdc:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

</sdc:pmDataList>

</sdc:GetPerformanceMonitoringDataResponse>
```

Following are the request and response examples of getChildObjectsofType
attribute for SHUB Bridge Port of SHUB Agent.

Request example

```
<sdc:getPerformanceMonitoringData>

<sdc:pmObjectSelect extVersion="?" extAuthor="?" tmf854Version="?">

<tmf:mdNm>ams</tmf:mdNm>

<tmf:meNm>172.21.176.241</tmf:meNm>

<tmf:propNm>/type=SHUB/SHUB</tmf:propNm>

</sdc:pmObjectSelect>

<sdc:pmParameterList>

<sdc:pmParameter>

<sdc:pmParameterName>getChildObjectsofType=BridgePort</sdc:pmParameterName>

<sdc:pmParemeterMaxIntervals>1</sdc:pmParemeterMaxIntervals>

</sdc:pmParameter>

</sdc:pmParameterList>

</sdc:getPerformanceMonitoringData>
```

Response example

```
<sdc:GetPerformanceMonitoringDataResponse xmlns:sdc="sdcNbi">

<sdc:pmDataList>

<sdc:pmData>

<sdc:pmParameterName>getChildObjectsofType=BridgePort</sdc:pmParameterName>

<sdc:pmParameterValue>[SHUB:6,SHUB:23,SHUB:13,SHUB:27,SHUB:11,SHUB:21,SHUB:
31,SHUB:17,SHUB:15,SHUB:1,SHUB:25,SHUB:2,SHUB:5,SHUB:37,SHUB:39,SHUB:9,SHUB
:41,SHUB:43,SHUB:33,SHUB:29,SHUB:35,SHUB:19]</sdc:pmParameterValue>

<sdc:pmParameterStatus>PMIS_Valid</sdc:pmParameterStatus>

</sdc:pmData>

</sdc:pmDataList>

</sdc:GetPerformanceMonitoringDataResponse>
```

5.10   enablePerformanceMonitoringData SOAP XML envelope body

Table 25 describes the elements used in the enablePerformanceMonitoringData SOAP envelope body of a request message. See section 3.5 for information about the elements that are present in the header of request and response messages.

Table 25   **enablePerformanceMonitoringData request SOAP XML envelope body**

Element	Description	Required	Value
pmObjectSelect	Object identifier of the target object for enabling PM counters. For example, <name> <mdNm>AMS</mdNm> <meNm>gpon17</meNm> <propNm>/type=POTS Port/R1.S1.LT7.PON1.ONT15.C1.P1</propNm> </name>	Mandatory	NamingAttributes_T
NamingAttributes_T elements			
mdNm	Management domain name	Mandatory	String. For example, <tmf:mdNm>AMS</tmf:mdNm>
meNm	Network element name	Mandatory	String. For example, <tmf:meNm>ISAM17</tmf:meNm>

(1 of 2)

Element	Description	Required	Value
propNm	Friendly object type and name (For more information on the format of the 5520 AMS friendly object type and object identifier, see the 5529 SDC Parameters Guide for the NE in the supported object parameter's section of the target object.)	Mandatory	String. For example, <tmf:propNm>/type=XDSL Port/R1.S1.LT7.P1</tmf:propNm>
pmParameterList	List of explicit start PM counters.	Mandatory	PerformanceMonitoringParameterList_T
PerformanceMonitoringParameterList_T elements			
pmParameter	One or more instances of a member PM parameter.	Mandatory	PerformanceMonitoringParameter_T
PerformanceMonitoringParameter_T elements			
pmParameterName	Name of the PM parameter for which the explicit start counter is to be enabled.	Mandatory	string
pmParemeterMaxIntervals[1]	Maximum PM intervals to be retrieved (default=1, maximum=288)	No	integer
vendorExtensions[1]	Container element for additional user parameters.	No	PackageList_T

(2 of 2)

Notes

[1]   The attributes *pmParemeterMaxIntervals* and *vendorExtensions* are not used in this operation.

Table 26 describes the elements used in the enablePerformanceMonitoringData SOAP envelope body of a response message.

Table 26   **enablePerformanceMonitoringData response SOAP XML envelope body**

Element	Description	Required	Value
pmDataList	Performance monitoring returned data list	Mandatory	PerformanceMonitoringDataList_T
PerformanceMonitoringDataList_T elements			
pmData	One or more instances of a member PM data	Mandatory	PerformanceMonitoringData_T
PerformanceMonitoringData_T elements			
pmParameterName	PM parameter name of the explicit start PM counter	Mandatory	string
pmParameterStatus	Measurement parameter status. Allowed values: • PMIS_Valid - Valid explicit start of PM counter and it is enabled sucessfully. • PMIS_Invalid - Valid explicit start of PM counter, but it is not enabled sucessfully. • PMIS_Unavailable - No PM counter available	Mandatory	string (enum)

(1 of 2)

Element	Description	Required	Value
pmAdditonalInfo	Additional parameter information, used for "PMIS_Unavailable" or "PMIS_Invalid" parameter values. Valid values: • PARAMETER_NOT_FOUND • PARAMETER_NOT_SUPPORTED This field is not returned for valid parameters.	Optional	string

(2 of 2)

5.11   disablePerformanceMonitoringData SOAP XML envelope body

Table 27 describes the elements used in the disablePerformanceMonitoringData SOAP envelope body of a request message. See section 3.5 for information about the elements that are present in the header of request and response messages.

Table 27   **disablePerformanceMonitoringData request SOAP XML envelope body**

Element	Description	Required	Value
pmObjectSelect	Object identifier of the target object for disabling PM counters. For example, <name> <mdNm>AMS</mdNm> <meNm>gpon17</meNm> <propNm>/type=POTS Port/R1.S1.LT7.PON1.ONT15.C1.P1</propNm> </name>	Mandatory	NamingAttributes_T
NamingAttributes_T elements			
mdNm	Management domain name	Mandatory	String. For example, <tmf:mdNm>AMS</tmf:mdNm>
meNm	Network element name	Mandatory	String. For example, <tmf:meNm>ISAM17</tmf:meNm>
propNm	Friendly object type and name (For more information on the format of the 5520 AMS friendly object type and object identifier, see the 5529 SDC Parameters Guide for the NE in the supported object parameter's section of the target object.)	Mandatory	String. For example, <tmf:propNm>/type=XDSL Port/R1.S1.LT7.P1</tmf:propNm>
pmParameterList	List of explicit start PM counters that is to be disabled.	Mandatory	PerformanceMonitoringParameterList_T
PerformanceMonitoringParameterList_T elements			

(1 of 2)

Element	Description	Required	Value
pmParameter	One or more instances of a member PM parameter.	Mandatory	PerformanceMonitoringParameter_T
PerformanceMonitoringParameter_T elements			
pmParameterName	Name of the PM parameter for which the explicit start counter is to be disabled.	Mandatory	string.
pmParemeterMaxIntervals[1]	Maximum PM intervals to be retrieved (default=1, maximum=288)	No	integer
vendorExtensions[1]	Container element for additional user parameters.	No	PackageList_T

(2 of 2)

Notes

[1]   The attributes *pmParemeterMaxIntervals* and *vendorExtensions* are not used in this operation.

Table 28 describes the elements used in the disablePerformanceMonitoringData SOAP envelope body of a response message.

Table 28   **disablePerformanceMonitoringData response SOAP XML envelope body**

Element	Description	Required	Value
pmDataList	Performance monitoring returned data list	Mandatory	PerformanceMonitoringDataList_T
PerformanceMonitoringDataList_T elements			
pmData	One or more instances of a member PM data	Mandatory	PerformanceMonitoringData_T
PerformanceMonitoringData_T elements			
pmParameterName	PM parameter name of the explicit start PM counter	Mandatory	string
pmParameterStatus	Measurement parameter status. Allowed values: • PMIS_Valid - Valid explicit start of PM counter and it is disabled sucessfully. • PMIS_Invalid - Valid explicit start of PM counter, but it is not disabled sucessfully. • PMIS_Unavailable - No PM counter available	Mandatory	string (enum)
pmAdditonalInfo	Additional parameter information, used for "PMIS_Unavailable" or "PMIS_Invalid" parameter values. Valid values: • PARAMETER_NOT_FOUND • PARAMETER_NOT_SUPPORTED This field is not returned for valid parameters.	Optional	string

5.12   Using object type and name for propNm element in enablePerformanceMonitoringData and disablePerformanceMonitoringData operations

These operations are used to enable/disable collection of performance monitoring data where the counters are required to be explicitly started by a user.

Enabling the performance monitoring (PM Interval counters or Rolling Counters) needs to be started and stopped by providing the parent object. The parent object will be used in "propNm" element of the request. To determine if a historical counter needs to be explicitly started, check if the corresponding current counter needs to be started explicitly. Starting explicitly the PM collection for a current counter will also start PM collection for the corresponding historical counter.

For information about the format of object identifiers used in propNm, see the 5529 SDC NE Support Plug-In Parameters Guide for the NE.

6 Error Handling

6.1 5529 SDC NBI exceptions

6.1   5529 SDC NBI exceptions

Table 29 describes the exceptions that the 5529 SDC NBI component generates for performance monitoring operations. The 5529 SDC interface supports the MTOSI exception "ProcessingFailureException_T" with the following exception types:

Table 29   **Performance monitoring exceptions**

Exception	Description
EXCPT_INVALID_INPUT (1)	The given input is not valid (this exception is seen due to validation or formatting errors).
EXCPT_ENTITY_NOT_FOUND (1)(2)(3)	The operation failed because an object is not found.
EXCPT_UNABLE_TO_COMPLY	The operation failed due to an internal error (this exception is due to an NE error, or internal 5520 AMS error). In addition to the above, this exception is returned when the NE is not in supervised state.
EXCPT_ACCESS_DENIED	The OSS user is not permitted to invoke this operation (this exception is due to license failure or user authorization failure).
EXCPT_CAPACITY_EXCEEDED	The maximum number of concurrent operations allowed has been reached (this exception is seen due to capacity or resources failure issues).

Notes

(1)   When an NE does not exist in the 5520 AMS, SDC NBI will return either EXCPT_INVALID_INPUT or EXCPT_ENTITY_NOT_FOUND. It is recommended that OSS check both exception codes for invalid NE name.

(2)   The following error scenarios should be considered when designing the work flow for an OSS:

- When an object has incorrect format in the request, the operation will return EXCPT_ENTITY_NOT_FOUND. For example in the format /Type=Slot/IACM:RR1.S1.LT4, RR1 is an incorrect format.

- When an object does not exist, the response will have pmParameterStatus attribute set to PMIS_Unavailable for every attribute in the request. For example, in the format /Type=Slot/IACM:R1.S1.LT10.P100, P100 does not exist.

(3)   This exception is not applicable for getPerformanceMonitoringDataForObjects operation as the request contains multiple target object identifiers.

Customer document and product support

Customer documentation

Customer Documentation Welcome Page

Technical Support

Customer Documentation Technical Support

Documentation feedback

Customer Documentation Feedback