VSI OpenVMS

PERFDAT V4.8

PERFDAT_MGR Reference Manual

February 2019

Revision/Update Information	New Manual
Software Version	VSI PERFDAT V4.8
Operating System Version	OpenVMS Alpha V7.3-2 & higher OpenVMS I64 V8.2 & higher

vms Software

January 2019

Contents

Preface

This manual includes:

- Introduction to PERFDAT_MGR including a brief description of the command set ordered by the main tasks of PERFDAT_MGR
- Detailed description of the PERFDAT_MGR command set alphabetically ordered.

Audience

This manual provides a detailed description of the PERFDAT_MGR command set. The reader should be familiar with:

- *VSI OpenVMS PERFDAT– Architecture and Technical Description*

Document Structure

- Chapter 1 Introduction to PERFDAT_MGR
- Chapter 2 PERFDAT_MGR reference section

Conventions Used in this Manual

Special	in examples indicates text that the system displays or user type input.
UPCASE	in a command represents text that you have to enter as shown.
Lowercase Italics	indicates variable information that a user supplies.
[ ]	in a command definition, enclose parts of the command that a user can omit.
Key	indicates a named key on the keyboard; for example, RETURN
CTRL/x	is the symbol used to represent the pressing of a control key. It indicates that the user holds down the key marked Ctrl and press the appropriate key.

Introduction to PERFDAT_MGR

This chapter provides an overview of the main tasks of PERFDAT_MGR and a brief description of the commands available.

The PERFDAT_MGR utility is the common management interface for the components of PERFDAT.

The PERFDAT_MGR image PERFDAT_MGR.EXE is located in the directory SYS$COMMON:[SYSEXE].

The main tasks of PERFDAT_MGR are:

- Startup/shutdown of the PERFDAT environment
- Controls and monitors the status of OpenVMS performance data collections
- Controls and monitors the status of performance data collections processed by the PERFDATEVA extension
- Controls and monitors the status of remote performance data collections processed by the PERFDAT SNMP extension
- Controls and monitors the status of application data collections processed by the VSI PERFDAT API
- Management/control of the performance data archiving
- Management/maintenance of the PERFDAT configuration database
- Online performance alert management
- Management/control of the performance database file name cache service DQL_NAME

To invoke PERFDAT_MGR type at the DCL prompt

$ MCR PERFDAT_MGR

For more information about the components of PERFDAT, their interaction and the overall architecture, please see the manual VSI OpenVMS PERFDAT– Architecture and Technical Description.

Startup / shutdown of the PERFDAT environment

Any privileged user can start and shutdown the whole PERFDAT environment or parts of it via PERFDAT_MGR. Table 1.2 lists the PERFDAT_MGR commands available for start-up and shutdown processing.

Table 1.2 Command reference table for start-up and shutdown processing

Command	**Description**

LAUNCH ALL	Starts up the whole PERFDAT environment on the local OpenVMS node. The components started are
	• OpenVMS data collector
	• EVA extension
	• SNMP extension
	• All components of the DQL interface
	  ○ DQL$SRV
	  ○ PDBC$SRV
	• Performance database filename cache service
	• Auto Archiving process

LAUNCH PERFDAT	All components but the SNMP extension and the EVA extension will be started as listed for the LAUNCH ALL command.
LAUNCH PERFDAT_EVA	EVA extension startup. All components but the SNMP extension and the OpenVMS data collector will be started as listed for the LAUNCH ALL command.
LAUNCH PERFDAT_SNMP	SNMP extension startup. All components but the EVA extension and the OpenVMS data collector will be started as listed for the LAUNCH ALL command
LAUNCH DQL$SRV	The DQL$SRV will be started only.
LAUNCH PDBC$SRV	The PDBC$SRV will be started only.
START ARCHIVE	Starts up the auto archiving process.
STOP ARCHIVE	Shutdown of the auto archiving process only.
START NAME_SERVER	Starts up the performance database filename cache service.
STOP NAME_SERVER	Shutdown of the performance database filename cache service
SHUTDOWN ALL	Actions performed • Shutdown of the archiving process. • Shutdown of the performance database filename cache service • Stops all active collections of the OpenVMS data collector • Stops all active collections of the SNMP extension • Stops all active collections of the EVA extension • Shuts down the OpenVMS data collector process • Shuts down the SNMP extension master process • Shuts down the EVA extension master process

SHUTDOWN PERFDAT	Actions performed • Stops all active collections of the OpenVMS data collector • Shuts down the OpenVMS data collector process
SHUTDOWN PERFDAT_SNMP	Actions performed • Stops all active collections of the SNMP extension • Shuts down the SNMP extension master process
SHUTDOWN PERFDAT_EVA	Actions performed • Stops all active collections of the EVA extension • Shuts down the EVA extension master process

In order to start the PERFDAT environment or parts of it users do not have to execute the LAUNCH / START commands of the PERFDAT_MGR utility. For all start-up commands, a sub-process is spawned and a command procedure is executed. Thus, the user can directly call these start-up command procedures from the DCL command prompt as shown in Table 1.3.

Table 1.3 Startup script reference table

Command	**Startup scripts executed**
LAUNCH ALL	SYS$STARTUP:PERFDAT$STARTUP.COM SYS$STARTUP:PERFDAT_SNMP$STARTUP.COM SYS$STARTUP:PERFDAT_EVA$STARTUP.COM
LAUNCH PERFDAT	SYS$STARTUP:PERFDAT$STARTUP.COM
LAUNCH PERFDAT_SNMP	SYS$STARTUP:PERFDAT_SNMP$STARTUP.COM
LAUNCH PERFDAT_EVA	SYS$STARTUP:PERFDAT_EVA$STARTUP.COM
LAUNCH DQL$SRV	SYS$STARTUP:DQL$STARTUP.COM
LAUNCH PDBC$SRV	SYS$STARTUP:PDBC$STARTUP.COM
LAUNCH NAME_SERVER	SYS$STARTUP:DQL_NAME$STARTUP.COM
START ARCHIVE	SYS$STARTUP:PERFDAT_ARCHIVE$STARTUP.COM

For additional information regarding the commands and their qualifiers, please see the PERFDAT_MGR reference section of this manual.

Almost all VSI PERFDAT jobs have to run under the DQL$SRV user name and UIC[1]. Thus, if a user starts any of the VSI PERFDAT components using the PERFDAT_MGR LAUNCH commands the following batch command script is executed:

SYS$STARTUP:PERFDAT$STARTUP_BATCH.COM

This batch command script submits the appropriate startup scripts listed in Table 1.3 into a batch queue on behalf of the DQL$SRV user. This startup batch queue can be user defined with the logical PERFDAT$STARTUP_QUEUE.

[1]The DQL$SRV user account and is automatically created when HP PERFDAT is installed on a system.

If the batch queue referred by the logical exists and its status is idle, busy or available the startup scripts are submitted into this batch queue.

Otherwise the PERFDAT$STARTUP_BATCH.COM creates and initializes a temporary batch queue to execute the startup scripts.

The logical PERFDAT$STARTUP_QUEUE has to be defined system wide.

```
$ DEFINE/SYSTEM PERFDAT$STARTUP_QUEUE queue-name
```

In order to define the logical permanently VSI strongly recommends that you define the logical in SYS$STARTUP:

SYS$STARTUP:PERFDAT$LOGICALS_CUSTOM.COM.

If this file does not exist in SYS$STARTUP, copy the template file PERFDAT$CFG:PERFDAT$LOGICALS_CUSTOM.TEMPLATE into either of these areas:

- SYS$COMMON:[SYS$STARTUP], if you want to maintain a common logical definition file containing the node-specific logicals
- SYS$SPECIFIC:[SYS$STARTUP], if you want to maintain node-specific logical definition files

Note

Starting with VSI PERFDAT V3.3 it is strongly recommended to use the startup command scripts rather than the PERFDAT_MGR launch commands. The VSI PERFDAT startup command scripts check the actual OpenVMS version in use before starting VSI PERFDAT. If OpenVMS has been upgraded all VSI PERFDAT version specific images are replaced and loaded automatically. The PERFDAT_MGR image is such a version specific image.

Performance data collection management

One can manually start and stop data collections from the OpenVMS data collector, the SNMP extension and the EVA extension and monitor the status of the active collections on the local node. Table 1.4 summarizes the commands available for managing performance data collections.

Table 1.4 Command reference table for managing performance data collections

Command	Description
START COLLECTION *profile-name*	Start a new performance data collection according to the settings of the predefined collection profile *profile-name*. Qualifiers: /ADDRESS = *IP-address* This qualifier is mandatory if you start a remote data collection via the SNMP extension or the EVA extension (/OS_TYPE is not OpenVMS). You can enter the IP address of the remote system or the full qualified IP name. It is recommended to enter the IP address. /COMMUNITY This qualifier is optional if you start a remote data collection via the SNMP extension or the EVA extension (/OS_TYPE is not OpenVMS). Enter the community name to be used for SNMP *GET* requests. Since the SNMP community string check is case sensitive it is strongly recommended to use quotation marks when you specify the community string. /DEVICE = *EVA access device (1GGAxxx)* The /DEVICE qualifier is also mandatory if you start a performance data collection for a HP StorageWorks Virtual Array (EVA). This qualifier defines the access device to the EVA system. /NODE = *node name* This qualifier is mandatory if you start a remote data collection via the SNMP or EVA extension (/OS_TYPE is not OpenVMS). Enter the node name of the remote system. /FLUSH_TIME = *time of day* Each performance data collection started creates a new data file daily. With the /FLUSH_TIME qualifier you can define the time of day the new data file shall be created. If you omit the qualifier new data files are created at day change. /OS_TYPE = *keyword* Defines the system or applicationthe profile *profile-name* is valid for. If you want to start an OpenVMS data collection the qualifier can be omitted. If you want to start a data collection for a non-OpenVMS system (either via the SNMP, the EVA extension or by use of the VSI PERFDAT API) the qualifier is mandatory. Supported keywords are • OpenVMS

- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

/OPENVMS_STYLE

The optional /OPENVMS_STYLE qualifier is only valid for EVA (HP StorageWorks Enterprise Array) data collections. It defines whether the performance data of a virtual disk with an OS unit ID assigned that is greater than zero will be stored using the OpenVMS FC device format (1DGAxxx, where xxx = OS unit ID of the virtual disk) or with its friendly name assigned by CV/EVA (CommandView/EVA).

/SHARE

If you start a data collection with the /SHARE qualifier the data of the actual collection is online accessible via the DQL interface.

The penalty is that the overall system performance may suffer due to excessive locking activity.

/SOURCE_ADDRESS

This qualifier is only valid for SNMP performance data collections. It defines the source IP address of the UDP/IP socket to be created and used by the SNMP extension to request SNMP performance data from a particular non OpenVMS system.

STOP COLLECTION *profile-name*

Stops an active performance data collection started with the collection profile *profile-name*.

Qualifiers:

/NODE = *node name*

This optional qualifier can be applied if one wants to stop an active performance data collection for a specific node.

/OS_TYPE = keyword

Defines the system or applicationthe profile *profile-name* is valid for. If you want to stop an OpenVMS data collection the qualifier can be omitted. If you want to stop a data collection for a non-OpenVMS system the qualifier is mandatory. Supported keywords are

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT

	performance database
	If you enter no additional qualifier all active collections for all nodes that match the *profile-name* parameter and the operating system type defined by this qualifier (if the qualifier is omitted OpenVMS is assumed) are stopped.
SUBMIT COLLECTION *profile-name*	Schedules a new performance data collection to be started using the collection profile *profile-name*. In contrast to the START COLLECTION command the data collection is not started directly, but the command is forwarded to the scheduler of VSI PERFDAT. This command is only valid for submitting OpenVMS collections The OpenVMS data collector checks if any active collection is using the same profile name. In that case the start command is rejected. Depending on the qualifiers applied, the performance data collection is periodically retriggered or a single shot collection. Qualifiers: /AFTER = OpenVMS date & time format Defines the time the scheduler triggers the performance data collection the first time. If you omit the qualifier the collection is started immediately. /UNTIL = OpenVMS date & time format Defines the time the scheduler stops the collection. If you omit the qualifier the collection stays active until you stop it manually. /RESTART = OpenVMS delta date & time format With the /RESTART_INTERVAL qualifier you can define cyclic performance data collection activations. One the performance collection is started it will be periodically re-activated with a time delay defined by this qualifier. If you omit the qualifier the scheduled collection is a single shot collection. This qualifier requires the /AFTER and /UNTIL qualifier to be defined too. /SHARE If you start a performance data collection with the /SHARE qualifier the data of the actual collection is online accessible via the DQL interface. The penalty is that the overall system performance may suffer due to excessive locking activity.
SHOW COLLECTION *profile-name*	Shows the status of an active performance data collection started with the collection profile *profile-name*. VSI PERFDATV3.0 and higher versions provide full wildcard support for the *profile-name* parameter. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within the string. Qualifiers: /ADVANCED Displays advanced information of the active collections.

	/BRIEF Displays summary information of all collection active – OpenVMS data collector, the SNMP extension and the EVA extension. /NODE=*node_name* Applying the /NODE qualifier displays all the performance collections active on/for the node defined by its value. VSI PERFDATV3.0 and higher versions provide full wildcard support for the *node-name* string. /OS_TYPE = keyword The /OS_TYPE qualifiers can be applied to selectively display the status of active performance data collections for systems or applications specified by this qualifier. Supported keywords are • OpenVMS • Tru64 • Brocade • EVA • Solaris • Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

All commands listed in Table 1.4 except the SUBMIT COLLECTION command are valid for managing data collections of the OpenVMS data collector as well as of the SNMP and the EVA extension. The value assigned to the qualifier /OS_TYPE determines which component PERFDAT_MGR forwards the request to (OpenVMS data collector, SNMP, EVA extension, VSI PERFDAT API). If you omit any qualifier the request is sent to the OpenVMS data collector.

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

Managing data archiving

Table 1.5 summarizes the commands for managing the archiving process.

Table 1.5 Command reference table for managing the archiving process

Command	Description
START ARCHIVE	Starts up the auto archiving process.
STOP ARCHIVE	Shuts down of the auto archiving process.
DEFINE ARCHIVE	Changes the control parameters for the archiving process in the archive control table of the PERFDAT configuration database. This is done by applying different qualifiers /ENABLE Enables archive processing /DISABLE Disable archive processing. /KEEP_DAYS = *integer number* Number of days to keep performance data collection files in the directory PERFDAT$DB_TREND. /TIME_OF_DAY = *OpenVMS time format* Defines the time the archiving process will be daily triggered. Any change of the parameters applied with the DEFINE command does not effect the active archiving process but its default settings used when the archiving process (re)starts.
SET ARCHIVE	The SET command changes the parameter of the volatile archive control table. Thus, this command dynamically changes the behaviour of the archiving process. The settings will be lost when restarting the archiving process. The parameters are changed by applying different qualifiers /ENABLE Enables archive processing. /DISABLE Disable archive processing. /KEEP_DAYS = *integer number* Number of days to keep performance data collection files in the directory PERFDAT$DB_TREND. /TIME = *OpenVMS date & time format* Next archive time (date & time).
SHOW ARCHIVE	Displays the parameter settings of the permanent and volatile archive control table.

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

Management / maintenance of the PERFDAT configuration database

Auto-start table

A user can add, modify, delete, and view the entries in the auto-start table of the PERFDAT configuration database. All nodes of a collection shall be automatically started upon launching the OpenVMS data collector, the SNMP extension, or the EVA extension, or when the VSI PERFDAT API is initialized to run an application data collection have to be inserted into this table.

Table 1.6 summarizes the commands for managing the auto-start table

Table 1.6 Command reference table for managing the auto-start table

Command	Description
ADD AUTOSTART *node-name*	Register a new node (*node-name*) in the auto-start table by invoking the auto-start configuration wizard. Optional qualifier: /OS_TYPE Defines the system or application to be added. If it is an OpenVMS node the qualifier can be omitted. If the system is non-OpenVMS the qualifier is mandatory. Supported keywords are: • OpenVMS • Tru64 • Brocade • EVA • Solaris • Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database Depending on the /OS_TYPE qualifier the auto-start configuration wizard asks for different inputs.
DELETE AUTOSTART *node-name*	Deletes a node (*node-name*) from the auto-start table. Optional qualifier: /OS_TYPE Defines the system or application to be deleted. If it is an OpenVMS node the qualifier can be omitted. If the system is non-OpenVMS the qualifier is mandatory. Supported keywords are: • OpenVMS • Tru64 • Brocade • EVA • Solaris

	• Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database
MODIFY AUTOSTART *node-name*	Modifies the parameter in the auto-start table for an existing node (*node-name*) entry by invoking the auto-start configuration wizard. Optional qualifier: /OS_TYPE Defines the system or application to be modified. If it is an OpenVMS node the qualifier can be omitted. If the system is non-OpenVMS the qualifier is mandatory. Supported keywords are: • OpenVMS • Tru64 • Brocade • EVA • Solaris • Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database Depending on the /OS_TYPE qualifier the auto-start configuration wizard asks for different inputs.
SHOW AUTOSTART *node-name*	Displays the entries of the auto-start database. The parameter *node-name* is optional. Optional qualifier: /OS_TYPE Defines the system or application to be displayed. Supported keywords are: • OpenVMS • Tru64 • Brocade • EVA • Solaris • Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

Archive control table

Table 1.7 summarizes the commands for managing the archive control table of the PERFDAT configuration database.

Table 1.7 Command reference table for managing the archive control table

Command	Description
DEFINE ARCHIVE	Changes the control parameters for the archiving process in the archive control table of the PERFDAT configuration database. This is done by applying different qualifiers /ENABLE Enables archive processing /DISABLE Disable archive processing. /KEEP_DAYS = *integer number* The value of this qualifier defines how long (how many days)performance data shall be kept in the directory PERFDAT$DB_ARCHIVE on the local node before the archiving process deletes these data files. /TIME_OF_DAY = *OpenVMS time format* Defines the time the archiving process will be daily triggered. Any change of the parameters applied with the DEFINE command does not effect the active archiving process but its default settings used when the archiving process (re)starts.

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

Collection profile table

Any performance data collection is profile controlled. These collection profiles are stored in the collection profile table of the PERFDAT configuration database. A user can add, copy, modify, delete, import, export and view the collection profiles defined in that table. Table 1.8 summarizes the commands for managing the collection profile table

Table 1.8 Command reference table for managing the collection profile table

Command	Description
ADD PROFILE *profile-name*	Adds a new collection profile named *profile-name* to the collection profile table by invoking the collection profile configuration wizard. Optional qualifier: /ADVANCED Depending on the /OS_TYPE qualifier the profile configuration wizard provides additional profile configuration options. For more detailed information about the /ADVANCED qualifier please refer to the PERFDAT_MGR Reference Section. /OS_TYPE Defines the system or applicationthe collection profile is valid for. If the collection profile is valid for OpenVMS the qualifier can be omitted. If the collection profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are: • OpenVMS • Tru64 • Brocade • EVA • Solaris • Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database Depending on the /OS_TYPE qualifier the collection profile wizard asks for different inputs.
COPY PROFILE *old-profile new-profile*	Copies the existing collection profile *old-profile* to the new collection profile *new-profile*. Optional qualifier: /OS_TYPE Defines the system or applicationthe collection profile is valid for. If the collection profile is valid for OpenVMS the qualifier can be omitted. If the collection profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are: • OpenVMS • Tru64 • Brocade

- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

DELETE PROFILE *profile-name*

Deletes the existing collection profile named *profile-name* from the collection profile table. Optional qualifier:

/OS_TYPE

Defines the system or applicationthe collection profile is valid for. If the collection profile is valid for OpenVMS the qualifier can be omitted. If the collection profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

EXPORT PROFILE *profile-name*

Exports an existing collection profile defined by the *profile_name* parameter from the collection profile table of the VSI PERFDAT configuration database to a transport file.

Mandatory Qualifier:

/FILENAME

File name of the transport file.

Optional qualifiers:

/OS_TYPE

Defines the system or applicationthe exported collection profile is valid for. If the collection profile is valid for OpenVMS the qualifier can be omitted. If the collection profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

IMPORT PROFILE *profile-name*

Imports a collection profile defined by the *profile_name*

parameter from a transport file to the collection profile table of the VSI PERFDAT configuration database.

Mandatory Qualifier:

/FILENAME

File name of the transport file.

Optional qualifiers:

/OS_TYPE

Defines the system or applicationthe imported collection profile is valid for. If the collection profile is valid for OpenVMS the qualifier can be omitted. If the collection profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

/UPDATE

If the collection profile already exists in the collection profile table of the VSI PERFDAT configuration database you can apply the /UPDATE qualifier to update that collection profile.

MODIFY PROFILE *profile-name*

Modifies the collection profile *profile-name* by invoking the collection profile configuration wizard. Optional qualifier:

/ADVANCED

Depending on the /OS_TYPE qualifier the profile configuration wizard provides additional profile configuration options. For more detailed information about the /ADVANCED qualifier please refer to the PERFDAT_MGR Reference Section.

/OS_TYPE

Defines the system or applicationthe collection profile is valid for. If the collection profile is valid for OpenVMS the qualifier can be omitted. If the collection profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI

	PERFDAT API to insert data into the VSI PERFDAT performance database
SHOW PROFILE *profile-name*	Displays the collection profiles configured in the collection profile table. If *profile-name* is omitted all collection profiles are displayed. VSI PERFDATV3.0 and higher versions provide full wildcard support for the *profile-name* parameter. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within the string. Optional qualifiers: /ADVANCED Depending on the /OS_TYPE qualifier advanced profile configuration options are displayed. For more detailed information about the /ADVANCED qualifier please refer to PERFDAT_MGR Reference Section. /BRIEF Displays summary information of all collection profiles configured in the collection profile table. /OS_TYPE Displays the collection profiles valid for the system or application specified by the /OS_TYPE qualifier. Supported keywords are: • OpenVMS • Tru64 • Brocade • EVA • Solaris • Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

License table

Table 1.9 summarizes the commands for managing the license table of the PERFDAT configuration database.

Table 1.9 Command reference table for managing the license table

Command	Description
CHECK LICENSE	Reads the license table and displays the status of each license key found (type of license, valid/expired).
LOAD LICENSE *key*	Checks if the license *key* entered is valid and loads the *key* into the license table.
UNLOAD LICENSE *key*	Deletes the license *key*.

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

Record descriptor table

Table 1.10 shows the command for managing the record descriptor table of the PERFDAT configuration database.

Table 1.10 Command reference table for managing the record descriptor table

Command	Description
LOAD METRIX *filename*	Loads metric and record descriptors from a valid *filename* into the record descriptor table of the PERFDAT configuration database.

This command is reserved for use of VSI support only.

Report profile table

Trend, capacity and baseline reports are extracted from performance data either via the auto-trend engine or manually via DQL$. In either case these reports are profile controlled. These report profiles are stored in the report profile table of the PERFDAT configuration database. The user can add, copy, modify, delete, import, export and view the report profiles defined in that table. Table 1.11 summarizes the commands for managing the collection profile table.

Table 1.11 Command reference table for managing the collection profile table

Command	Description
ADD REPORT *report-name*	Adds a new report profile named *report-name* to the report profile table by invoking the report profile configuration wizard. Optional qualifier: /OS_TYPE Defines the system or application the report profile is valid for. If the report profile is valid for OpenVMS the qualifier can be omitted. If the report profile is valid for non-OpenVMS, the qualifier is mandatory. Supported keywords are: • OpenVMS • Tru64 • Brocade • RDB • EVA • Solaris • Linux • CACHE • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database
COPY REPORT *old-report new-report*	Copies an existing report profile *old-report* to the new report profile *new-report*. Optional qualifier: /OS_TYPE Defines the system or application the report profile is valid for. If the report profile is valid for OpenVMS the qualifier can be omitted. If the report profile is valid for non-OpenVMS, the qualifier is mandatory. Supported keywords are: • OpenVMS • Tru64 • Brocade • RDB • EVA • Solaris

- Linux
- CACHE
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

DELETE REPORT *report-name*

Deletes the existing report profile *report-name* from the report profile table. Optional qualifier:

/OS_TYPE

Defines the system or application the report profile is valid for. If the report profile is valid for OpenVMS the qualifier can be omitted. If the report profile is valid for non-OpenVMS, the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- RDB
- EVA
- Solaris
- Linux
- CACHE
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

EXPORT REPORT *report-name*

Exports an existing report profile defined by the *report_name* parameter from the report profile table of the PERFDAT configuration database to a transport file.

Mandatory Qualifier:

/FILENAME

File name of the transport file.

Optional qualifiers:

/OS_TYPE

Defines the system or applicationthe exported report profile is valid for. If the report profile is valid for OpenVMS the qualifier can be omitted. If the report profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- RDB
- EVA
- Solaris
- Linux
- CACHE
- Name of any application that uses the VSI

PERFDAT API to insert data into the VSI PERFDAT performance database

IMPORT REPORT *report-name*

Imports a report profile defined by the *report_name* parameter from a transport file to the report profile table of the PERFDAT configuration database.

Mandatory Qualifier:

/FILENAME

File name of the transport file.

Optional qualifiers:

/OS_TYPE

Defines the system or application the imported report profile is valid for. If the report profile is valid for OpenVMS the qualifier can be omitted. If the report profile is valid for non-OpenVMS the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- RDB
- EVA
- Solaris
- Linux
- CACHE
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

/SOURCE

This qualifier can be applied to select a different source collection profile as that stored in the report header section if an existing report profile is updated with the imported report profile.

/UPDATE

If the report profile already exists in the report profile table of the PERFDAT configuration database you can apply the /UPDATE qualifier to update the layout and statistics section of that report profile.

MODIFY REPORT *report-name*

Modifies the report profile *report-name* by invoking the report profile configuration wizard. Optional qualifier:

/OS_TYPE

Defines the system or application the report profile is valid for. If the report profile is valid for OpenVMS the qualifier can be omitted. If the report profile is valid for non-OpenVMS, the qualifier is mandatory. Supported keywords are:

- OpenVMS
- Tru64
- Brocade

	• RDB • EVA • Solaris • Linux • CACHE • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database
SHOW REPORT *report-name*	Displays the collection profiles configured in the report profile table. If *report-name* is omitted all report profiles are displayed. PERFDAT V3.0 and higher versions provide full wildcard support for the *report-name* parameter. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within the string.Optional qualifiers: /BRIEF Displays summary information of all report profiles configured in the collection profile table. /OS_TYPE Displays only the report profiles valid for the system or application specified by the /OS_TYPE qualifier. Supported keywords are: • OpenVMS • Tru64 • Brocade • RDB • EVA • Solaris • Linux • CACHE • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

Online performance alert management

Online performance alerting feature first introduced with PERFDAT V3.0 provides real-time alerting capabilities.

Online performance alerting can be enabled for any active performance data collection, independently if the data collection is performed by the OpenVMS data collector, the SNMP extension, the EVA extension or the VSI PERFDAT API.

Once online alerting has been enabled for an active performance data collection, the alerting subsystem tracks the actual values of specific statistics collected by the OpenVMS data collector, the SNMP extension, the EVA extension or by use of the VSI PERFDAT API. Alerts are triggered if any alert condition becomes true. An alert clear message (informational) is triggered if an alert condition is no longer true.

The statistics to monitor, the alert conditions and the alert method are defined by alert blocks within an alert definition file. Thus, a prerequisite for enabling online alerting for an active performance data collection is that a valid alert definition file exists. An alert definition file is a text file for easy customization.

The maximum number of elements tracked by the online performance alerting subsystem per alert block is 4096. If the number of elements associated with an alert block exceeds the supported number of elements the online performance alerting subsystem sends an OPCOM warning message.

For detailed description of alert definition files and alert blocks please refer the ENABLE ALERT command description.

Independently of the alert method defined in the alert definition file all alerts are written to log files. The log files are stored in the directory PERFDAT$ALERT. The log file that contains the OpenVMS alerts uses the following naming convention:

PERFDAT_ALERT_*node*.LOG_*date*

The naming convention of the log file that contains SNMP extension alerts is:

PERFDAT_SNMP_x_ALERT_*node*.LOG_*date*

where:

- *node* local node name
- *date* date the log file was created
- x working process number of the SNMP extension

The naming convention of the log file that contains EVA extension alerts is:

PERFDAT_EVA_x_ALERT_*node*.LOG_*date*

where:

- *node* local node name
- *date* date the log file was created
- x working process number of the EVA extension

The naming convention of the log file that contains VSI PERFDAT API alerts is:

*application*_ALERT_*node*.LOG_*date*

where:

- *application* Application name
- *node* local node name

On day change (midnight) a new log file is created if online alerting has been enabled for any data collection managed by the caller of the online performance alerting sub-system (OpenVMS data collector, SNMP extension, EVA extension, VSI PERFDAT API). Thus, one log file contains all alerts triggered by the alert subsystem on that day.

Table 1.11 summarizes the commands available for managing online performance alerting.

Table 1.11 Command reference table for managing online performance alerting

Command	Description
DISABLE ALERT *collection-profile*	Disable online alerting. The *collection-profile* name parameter specifies the active performance data collection that has online alerting enabled. With the SHOW COLLECTION command you can check if online alerting is enabled for the performance data collection specified by the *collection-profile* parameter. Qualifiers: /NODE = *node name* This qualifier is mandatory if you want to disable online alerting for an active non-OpenVMS performance collection. It specifies the node name of the remote system to disable online alerting /OS_TYPE = *keyword* The /OS_TYPE qualifier defines the system or application that runs a data collection started with the collection profile defined by the *collection-profile* parameter.In order to disable online alerting for an active non-OpenVMS performance data collection the /OS_TYPE qualifier is mandatory. If you disable online alerting for an active OpenVMS performance data collection profile you can omit the qualifier since OpenVMS is the default. Supported keywords are • OpenVMS • Tru64 • Brocade • EVA • Solaris • Linux • Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database
ENABLE ALERT *collection-profile*	Enable online alerting. The *collection-profile* name

parameter specifies the active performance collection that has online alerting disabled. With the SHOW COLLECTION command you can check if online alerting is disabled for the performance collection specified by the *collection-profile* parameter.

Qualifiers:

/ALERT_FILENAME = *filename*

The /ALERT_FILENAME qualifier specifies the alert definition file to use.

If you omit the /ALERT_FILENAME qualifier default alert definition files are used depending on the value of the /OS_TYPE qualifier as listed below. The files are located in PERFDAT$CFG:

OS_TYPE	Default alert definition file
OpenVMS	PERFDAT_ALERT_OPENVMS.CFG
Tru64	PERFDAT_ALERT_TRU64.CFG
Brocade	PERFDAT_ALERT_BROCADE.CFG
EVA	PERFDAT_ALERT_EVA.CFG

No default alert definition files are available for Solaris and Linux systems and any application. Thus, if you omit the /ALERT_FILENAME qualifier for Solaris, Linux or any application data collections the command fails.

/NODE = *node name*

This qualifier is mandatory if you want to enable online alerting for an active non-OpenVMS performance collection. It specifies the node name of the remote system to disable online alerting.

/OS_TYPE = *keyword*

The /OS_TYPE qualifier defines the system or application that runs a data collection started with the collection profile defined by the *collection-profile* parameter.In order to enable online alerting for an active non-OpenVMS performance collection the /OS_TYPE qualifier is mandatory. If you enable online alerting for an active OpenVMS performance data collection you can omit the qualifier since OpenVMS is the default. Supported keywords are

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the VSI PERFDAT performance database

CHECK ALERT *alert-definition-file*

This command reads the alert definition file defined by the *alert-definition-file* parameter and checks if all alert blocks defined within are valid. If an invalid line item is detected the line item and the line number is displayed.

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

Managing the performance database file name cache service DQL_NAME

The distributed performance database is organized in way that there exists no persistent root file for any PERFDAT performance database on disk (see manual VSI PERFDAT– Architecture and Technical description). All meta-data (field and record descriptors, data link descriptors, index reference table descriptor …) necessary to access the data is stored in the header of each physical storage area. The advantage is that data files can be moved to any OpenVMS node and the data file stays read accessible without any additional actions such as data conversion, unload and load operations. On the other hand the meta-data have to be fetched again any time a user connects to the distributed performance database via the DQL$ utility or the GUI in order to create a virtual root file (data link cache) required to access the data.

Prior to VSI PERFDATV3.2 the meta-data were fetched by performing a full data file header scan on all members of the PERFDAT community. Thus, prior to VSI PERFDATV3.2 the initial connect request to the distributed performance database took few seconds up to minutes.

In order to solve that performance issue VSI PERFDAT provides the performance database file name cache service DQL_NAME first introduced with VSI PERFDAT V3.2.

The performance database file name cache service DQL_NAME provides a database file name cache to all PERFDAT components that contains full header information about all PERFDAT database files locally stored. As long as the performance database file name cache service DQL_NAME is available and the database file name cache is marked valid all PERFDAT components obtain database file header information from that cache rather than scanning the files on disk. Thus, the initial connect request speeds up dramatically (ten times and more) compared to PERFDAT V3.1 and lower versions of PERFDAT.

The file name cache is updated by the DQL_NAME process:

- periodically once per TTL (time to leave) duration.
- whenever a PERFDAT component creates, renames or deletes a database file.

The TTL (time to leave) parameter defines the time duration the entries in the performance database file name cache are valid. Every entry has to be updated once during the TTL duration by DQL_NAME process. If TTL duration time expires and the entries in the cache have not been updated for any reason the cache is marked invalid and from this time on all PERFDAT components will fetch the file header information directly from disk on connect requests until the performance database file name cache service DQL_NAME starts processing again. Thus, the TTL parameter defines the cache entry lifetime.

The default value of the TTL parameter is 30 minutes.

Note

If you delete any PERFDAT data file in the directory PERFDAT$DB manually the file name cache will not be updated automatically until TTL time expires. Thus, in this case you have to trigger a rebuild of the performance database file name cache manually to keep the cache up to date. To perform a cache rebuild you can either use the FLUSH NAME_SERVER command or you stop the file name cache service and restart it again (has the same effect as the FLUSH NAME_SERVER command). If you run PERFDAT in a cluster you have to rebuild the file name cache on all cluster members.

You do not have to start the performance database file name cache service DQL_NAME manually after re-starting the PERFDAT environment/DQL interface. This is automatically done by the PERFDAT and DQL start-up routines.

Table 1.6 summarizes the commands for managing the performance database file name cache service DQL_NAME.

Table 1.6 Command reference table for managing the performance database file name cache service DQL_NAME

Command	Description
START NAME_SERVER	Starts up the performance database file name cache service DQL_NAME.
STOPNAME_SERVER	Shuts down of the performance database file name cache service DQL_NAME.
FLUSH NAME_SERVER	Flushes the performance database file name cache on the local node and triggers the DQL_NAME service to rebuild it.
SET NAME_SERVER	Use this command to change the time to leave duration of the performance database file name cache entries. /TTL The /TTL qualifier defines the new cache time to leave duration in minutes. The /TTL qualifier is mandatory. The new TTL value takes effect immediately after the current TTL period has expired.
SHOW NAME_SERVER	Displays the current TTL setting of the performance database file name cache service DQL_NAME.

For additional information regarding the commands and the respective qualifiers please see the PERFDAT_MGR reference section of this manual.

PERFDAT_MGR reference section

This section describes the available commands of the PERFDAT_MGR command set. They are provided in alphabetical order.

ADD AUTOSTART

This command adds an auto-start entry for a node to the auto-start table of the PERFDAT configuration database by starting the appropriate auto-start configuration wizard.

Format

ADD AUTOSTART *node_name*

Parameter

node_name
Specifies the node name to be added to the auto-start table of the PERFDAT configuration database.

This parameter is mandatory. The use of wildcards is not permitted.

Description

This command adds an auto-start entry for a node to the auto-start table of the PERFDAT configuration database by starting the appropriate auto-start configuration wizard.

The auto-start table of the VSI PERFDAT configuration database contains all required start-up parameters to start performance data collections automatically when launching the OpenVMS data collector, the SNMP extension, the EVA extension or when an application starts that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT collection database. The OpenVMS data collector, the SNMP extension, the EVA extension and the VSI PERFDAT API access this table. These components of VSI PERFDAT check if any performance data collections are defined to be started on the local node and they start them automatically if appropriate auto-start entries exist. This is done by checking the content of every entry in this table.

The entries of that auto-start table are also read by the auto-trend engine to determine if any trend and capacity report shall be processed. It checks:

- if the local node itself is registered in the auto-start table
- if the local node is defined as the agent (=node that runs a SNMP or an EVA data collection) for any remote system (Tru64, Brocade, EVA, Solaris, Linux) registered therein.
- if the local node is configured to start application data collections automatically whenever applications are started on the local node that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database.

If this is the case the reports defined in the report profile table of the PERFDAT configuration database marked to be processed by the auto-trend engine and valid for these auto-start entries are processed (the auto-trend engine checks if the OS type of the auto-started performance data collection matches the OS type of the report profiles).

Depending on the /OS_TYPE qualifier the auto-start configuration wizard prompts the user for different inputs.

/OS_TYPE = OpenVMS

- Collection profile to auto-start when launching the OpenVMS data collector. This field is also read by the auto-trend engine to determine the source collection database for capacity and trend report processing.
- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting
  First introduced with V3.0, PERFDAT provides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable period of time the system manager will be alerted via OPCOM messages and user definable command procedures.
- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc. The default alert definition file is
  PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG
- Deferred write
  If OpenVMS data collections are started on several nodes using the same sample interval, and the VSI PERFDAT data disks of all these nodes are configured on the same external storage (i.e. EVA array) this may cause I/O bursts at the end of each sample interval since all data collections try to write the collected data records to their data files concurrently.
  To overcome such I/O bursts VSI PERFDAT provides the deferred data write option for OpenVMS data collections. The deferred data write option is enabled if an OpenVMS data collection is started with a deferred data write timer value greater than zero. The deferred data write timer value defines the time in milliseconds that has to expire after the end of a sample interval before the collected data records are inserted into the data file. Thus, if OpenVMS data collections are started with different deferred data write timers I/O bursts can be inhibited even if the VSI PERFDAT data disk of all nodes within an environment access the same external storage.
- Data flush time
  Each performance data collection started creates a new data file daily. The time entered at the data flush time prompt defines the time of day a new data file shall be created for the auto-started performance data collection. Enter a valid time string only.

/OS_TYPE = (Tru64, Brocade, Solaris, Linux)

- Collection profile to auto-start when launching the SNMP extension. This field is also read by the auto-trend engine to determine the source collection database for capacity and trend report processing.
- IP address of the remote node
- SNMP community string
- Agent node
  The agent node defines where to run the SNMP data collection and the node to run the auto-trend engine for processing data collected for the remote system referred by this entry.
- Source IP address to use
  If the SNMP agent (OpenVMS node that run the SNMP data collection) has more than one IP address configured, you can define which source IP address will be used by the SNMP extension to request the SNMP performance data from the target system.
- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting
  First introduced with V3.0, PERFDATprovides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable period of time the system manager will be alerted via OPCOM messages and user definable command procedures.
- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc. Depending on the value of the /OS_TYPE qualifier the default alert definition file is:
  - TRU64
    PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG
  - BROCADE
    PERFDAT$CFG:PERFDAT_ALERT_BROCADE.CFG
  - No default alert definition files are available for SOLARIS and LINUX
- Deferred write
  If SNMP data collections are started on several nodes using the same sample interval, and the VSI PERFDAT data disks of all these nodes are configured on the same external storage (i.e. EVA array) this may cause I/O bursts at the end of each sample interval since all data collections try to write the collected data records to their data files concurrently.
  To overcome such I/O bursts VSI PERFDAT provides the deferred data write option for SNMP data collections. The deferred data write option is enabled if a SNMP data collection is started with a deferred data write timer value greater than zero. The deferred data write timer value defines the time in milliseconds that has to expire after

the end of a sample interval before the collected data records are inserted into the data file. Thus, if SNMP data collections are started with different deferred data write timers I/O bursts can be inhibited even if the VSI PERFDAT data disk of all nodes within an environment access the same external storage.

- Data flush time
  Each performance data collection started creates a new data file daily. The time entered at the data flush time prompt defines the time of day a new data file shall be created for the auto-started performance data collection. Enter a valid time string only.

/OS_TYPE = EVA

- Collection profile to auto-start when launching the EVA extension. This field is also read by the auto-trend engine to determine the source collection database for capacity and trend report processing.
- EVA access device
  Enter the console access device to the EVA (HP StorageWorks Virtual Array) system you want to monitor. You can access the console of an EVA system only if the 'Console LUN ID' parameter of the EVA system is greater than zero. If the 'Console LUN ID' parameter is greater than zero, and you have executed the MCR SYSMAN IO AUTOCONFIGURE command you will get a 1GGAxxx device, where xxx = 'Console LUN ID' parameter value of the EVA system. This is the device you have to enter.
- OpenVMS style
  The OpenVMS style parameter defines whether performance data of a virtual disk with an OS unit ID greater than zero will be stored using the OpenVMS FC device format (1DGAxxx, where xxx = OS unit ID of the virtual disk) or with its friendly name assigned by CV/EVA (CommandView/EVA).
- Instantaneous EVA configuration scan
  The EVA extension of VSI PERFDAT automatically detects EVA configuration changes (i.e. new virtual disks) without being connected to the SAN appliance. In order to collect performance data for newly configured EVA items (virtual disk, disk groups, physical disks, host connections) a full EVA configuration scan has to be performed in order to update the friendly name table of the EVA data collection. Due to the design of the EVA management interface (management commands are serialized) an EVA configuration scan of the EVA extension may slowdown any other utilities that access the EVA including Command View/EVA. This parameter defines whether or not an EVA configuration scan is instantaneously triggered when an EVA configuration change is detected. If the instantaneous configuration update option is disabled the EVA configuration scan is scheduled to be executed at midnight. Thus, the EVA extension does not influence active Command View/EVA sessions. The penalty incurred with this setting is that the performance data of any new configuration item is not collected until the EVA configuration scan has been performed (i.e. at midnight).
- Agent node

The agent node defines where to run the EVA data collection and the node to run the auto-trend engine for processing data collected for the EVA system referred by this entry.

- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting
  First introduced with V3.0, PERFDAT provides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable period of time the system manager will be alerted via OPCOM messages and user definable command procedures.
- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc. The default alert definition file is:
  - PERFDAT$CFG:PERFDAT_ALERT_EVA.CFG
- Data flush time
  Each performance data collection started creates a new data file daily. The time entered at the data flush time prompt defines the time of day a new data file shall be created for the auto-started performance data collection. Enter a valid time string only.

/OS_TYPE = *application-name*

- Collection profile to be used by the VSI PERFDAT API to auto-start an application data collection when a process of the application defined by the /OS_TYPE qualifier (*application-name* parameter) is started on the node defined by the auto-start *node_name* parameter of the ADD AUTOSTART command.
- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting
  First introduced with V3.0, PERFDAT provides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable period of time the system manager will be alerted via OPCOM messages and user definable command procedures.
- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc.

The predefined alert definition files

- PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG
- PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG
- PERFDAT$CFG:PERFDAT_ALERT_BROCADE.CFG
- PERFDAT$CFG:PERFDAT_ALERT_EVA.CFG

are part of the distribution kit. For detailed information about how to configure alert blocks within an alert definition file please refer to ENABLE ALERTcommand description.

Qualifier

/OS_TYPE=*system*|*application-name*

/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application that is referenced by the newly added auto-start entry.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to add a non-OpenVMS node to the auto-start table of the VSI PERFDAT configuration database and to start the appropriate auto-start configuration wizard the /OS_TYPE qualifier is mandatory.

If you are adding a new OpenVMS node to the auto-start database you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS -> The node is an OpenVMS node.
- Tru64 -> The node is a Tru64 node.
- Brocade -> The node is a Brocade switch.
- EVA -> HP StorageWorks Virtual Array
- Solaris -> The node is a Solaris node
- Linux -> The Node is a Linux node
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Examples

The ADD AUTOSTART commands invokes the auto-start wizard that prompts you for all start-up parameters for the node defined by the *node_name* parameter that are user definable.

Example 1

In this example an auto-start entry for an OpenVMS node named VMSTM4 is added to the auto-start table.

The collection profile 2MIN will be used for auto-starting a performance data collection on VMSTM4. Online alerting is enabled and the alert definition file in use is the default alert definition file. The start date for trend and capacity data processing is 10-MAR-2005. Thus, performance data created since 10-MAR-200500:00:00 are processed by the auto-trend engine the next (first) time it is triggered on VMSTM4. Performance data created before are ignored. New data files shall be created daily at 08:00. The database alias of the source collection database accessed by the auto-trend engine is VMSTM4_2MIN.

```
PerfDat_MGR> add autostart vmstm4
     Autostart Profile [DEFAULT] 2MIN↵
Allow online access [No]↵
     Enter Auto Report Start date [ 1-MAR-2005] 10-MAR-2005↵
     Enable online alerting  [No]: Yes↵
     Enter alert definition file [PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG]: ↵
   Data flush (data file close) time [00:00:00]08:00:00↵
```

Example 2

In this example an auto-start entry for a TRU64 node named MIANIX is added to the auto-start table.

The agent node is BCSXTC (= node that runs the SNMP data collection and hosts the data), and the collection profile 2MIN will be used when auto-starting the collection for node MIANIX on BCSXTC. Online alerting is enabled and the alert definition file in use is the user defined alert definition file PERFDAT$CFG:MIANIX_ALERT.CFG. The start date for trend and capacity data processing is 10-MAR-2005. Thus, performance data created since 10-MAR-2005 00:00:00 are processed by the auto-trend engine the next (first) time it is triggered on BCSXTC to create trend and capacity reports for node MIANIX. New data files shall be created daily at 03:00. The database alias of the source collection database accessed by the auto-trend engine is MIANIX_2MIN.

```
PerfDat_MGR> add autostart mianix/os_type=tru64
     Autostart Profile [DEFAULT] 2MIN↵

     You are configuring a remote host -> enter the IP address of the remote host.
          You can enter the IP address or the full qualified IP host name.
          It is recommended to enter the IP Address.

     IP address of remote host  [] 16.55.40.10↵
     Enter community string [public] ↵
Agent node (= node that runs the collection) [VMSTM1] BCSXTC ↵
     Allow online access  [No] ↵
     Enter Auto Report Start date [ 1-MAR-2005] 10-MAR-2005↵
     Enable online alerting  [No]: Yes↵
     Enter alert definition file [PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG]:
                                        PERFDAT$CFG:MIANIX_ALERT.CFG↵
```

```
   Data flush (data file close) time [00:00:00]03:00↵
```

Example 3

In this example an auto-start entry for an EVA (HP StorageWorks Virtual Array) system named EVA1 is added to the auto-start table.

The agent node is BCSXTC (= node that runs the EVA data collection and hosts the data), and the collection profile 2MIN will be used when auto-starting the collection for the HP StorageWorks Virtual Array EVA1 on BCSXTC. The value assigned to the 'Console LUN ID' parameter of the EVA system EVA1 is 90. Thus, the EVA access device is 1GGA90. Online access is enabled and online alerting is disabled. The start date for trend and capacity data processing is 10-JAN-2008. Thus, performance data created since 10-JAN2008 00:00:00 are processed by the auto-trend engine the next (first) time it is triggered on BCSXTC to create trend and capacity reports for HP StorageWorks Virtual Array EVA1. New data files shall be created daily at 03:00. The database alias of the source collection database accessed by the auto-trend engine is EVA1_2MIN.

```
PerfDat_MGR> add autostart eva1/os_type=eva
Autostart Profile [DEFAULT] 2MIN↵
EVA access device [] $1$GGA90:↵
Agent node (= node that runs the collection) [VMSTM1] BCSXTC ↵
     Allow online access [No] Yes↵
     Enter Auto Report Start date [ 10-JAN-2008] ↵
     Enable online alerting  [No]: ↵
   Data flush (data file close) time [00:00:00]03:00↵
```

ADD PROFILE

This command creates a new entry (profile) in the collection profile table of the PERFDAT configuration database.

Format

ADD PROFILE *profile_name*

Parameter

profile_name
Defines the name of the newly created collection profile. The profile name references the collection profile in the collection profile table of the PERFDAT configuration database.

If you add a new profile the *profile_name* has to be unique for the system or application defined by the /OS_TYPE qualifier, but you can reuse the same *profile_name* if you define profiles for different systems or applications.

This parameter is mandatory. Wildcard characters within the *profile_name* string are not permitted.

Description

Any performance data collection is profile controlled. These collection profiles are stored in the collection profile table of the PERFDAT configuration database. This command creates a new entry (profile) in that table, assigns the name specified by the *profile_name* parameter to that entry and starts the profile wizard for defining the profile.

The collection profile wizard prompts the user for the sample interval and the metrics to be enabled. Since the metrics available for the supported systems (OpenVMS, Tru64, Brocade, EVA, Solaris, Linux, applications that use VSI PERFDAT API) differ, the profile collection wizard prompts for different inputs, depending on the /OS_TYPE qualifier.

Collection profile wizard for OpenVMS

This section provides detailed description of the OpenVMS wizard inquiries. When adding or modifying a collection profile for OpenVMS the collection profile wizard asks for:

WELCOME to OpenVMS performance data profile wizard

Collection sample interval [600 sec] 120

Enter the collection interval in seconds.

Enable SYSTEM metric [Yes]

Enter Yes/No to enable/disable the SYSTEM metric for this collection profile. The default is Yes.

```
Enable CPU metric [Yes]
    ... CPU load threshold [0.00 %]          (with /ADVANCED qualifier only)
```

Enter Yes/No to enable/disable the CPU metric for this collection profile. The default is Yes.

If the command is entered with the /ADVANCED qualifier the user is prompted to define a CPU load threshold. If the average CPU load within a sample interval of a CPU is less than the threshold, the data sample for that CPU is not written. The default value for the threshold is 0.

```
Enable PROCESS metric [Yes]
    On Process  [ALL]
    ... CPU load threshold [0.00 %]                 (with /ADVANCED qualifier only)
    ... MEM load threshold [0.00 MByte]    (with /ADVANCED qualifier only)
    ... IO request threshold [0.00 Req/sec]  (with /ADVANCED qualifier only)
```

Enter Yes/No to enable/disable the PROCESS metric for this collection profile. The default is Yes.

The user can restrict the data collection to dedicated process ("On Process" inquiry). Enter all processes to monitor as a comma separated list. VSI PERFDAT V4.1 and higher versions provide the feature to define exclude lists. Processes that should be excluded from being monitored have to be preceded with the '<>' or '!=' tag. VSI PERFDATV3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list. The default is ALL (one can also enter an asterisk (*) instead).

If the command is entered with the /ADVANCED qualifier the user is prompted to define CPU load, Memory and I/O request thresholds. If all three statistic values of a data sample - the average CPU load, memory allocated and I/O request rate - for a process are less than these thresholds, the data sample for that process is not written. The default value for these thresholds is 0.

```
Enable USER metric [Yes]
    On User(s)  [ALL] *SYS*,*RDB*
    ... CPU load threshold [0.00 %]                 (with /ADVANCED qualifier only)
    ... MEM load threshold [0.00 MByte]    (with /ADVANCED qualifier only)
    ... IO request threshold [0.00 Req/sec]  (with /ADVANCED qualifier only)
```

Prerequisite to is that the PROCESS metric has been enabled before. Enter Yes/No to enable/disable the USER metric for this collection profile. The default is Yes.

The user can restrict the data collection to dedicated users ("On User(s)" inquiry). Enter all the users to monitor as a comma separated list. The user filter list is independent of the process filter list. VSI PERFDAT V4.1 and higher versions provide the feature to define

exclude lists. Users that should be excluded from being monitored have to be preceded with the '<>' or '!=' tag. VSI PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list. The default is ALL (one can also enter an asterisk (*) instead).

If the command is entered with the /ADVANCED qualifier the user is prompted to define CPU load, Memory and I/O request thresholds. If all three statistic values of a data sample - the average CPU load, memory allocated and I/O request rate - for a user are less than these thresholds, the data sample for that user is not written. The default value for these thresholds is 0.

Enable IMAGE metric [Yes]
On Image(s) [ALL]
... CPU load threshold [0.00 %] (with /ADVANCED qualifier only)
... MEM load threshold [0.00 MByte] (with /ADVANCED qualifier only)
... IO request threshold [0.00 Req/sec] (with /ADVANCED qualifier only)

Prerequisite is that the PROCESS metric has been enabled before. Enter Yes/No to enable/disable the IMAGE metric for this collection profile. The default is Yes.

The user can restrict the data collection to dedicated images ("On Image(s)" inquiry). Enter all images to monitor as a comma separated list. The image filter list is independent of the process and user filter list. VSI PERFDAT V4.1 and higher versions provide the feature to define exclude lists. Images that should be excluded from being monitored have to be preceded with the '<>' or '!=' tag. VSI PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list. The default is ALL (one can also enter an asterisk (*) instead).

If the command is entered with the /ADVANCED qualifier the user is prompted to define CPU load, Memory and I/O request thresholds. If all three statistic values of a data sample - the average CPU load, memory allocated and I/O request rate - for an image are less than these thresholds, the data sample for that image is not written. The default value for these thresholds is 0.

Enable ACCOUNT metric [Yes]
On Account(s) [ALL]
... CPU load threshold [0.00 %] (with /ADVANCED qualifier only)
... MEM load threshold [0.00 MByte] (with /ADVANCED qualifier only)
... IO request threshold [0.00 Req/sec] (with /ADVANCED qualifier only)

Prerequisite is that the PROCESS metric has been enabled before. Enter Yes/No to enable/disable the ACCOUNT metric for this collection profile. The default is Yes.

The user can restrict the data collection to dedicated accounts ("On Account(s)" inquiry). Enter all accounts to monitor as a comma separated list. The account filter list is independent of the process, user and image filter list. VSI PERFDAT V4.1 and higher versions provide the feature to define exclude lists. Accounts that should be excluded from being monitored have to be preceded with the '<>' or '!=' tag. VSI PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list. The default is ALL (one can also enter an asterisk (*) instead).

If the command is entered with the /ADVANCED qualifier the user is prompted to define CPU load, Memory and I/O request thresholds. If all three statistic values of a data sample - the average CPU load, memory allocated and I/O request rate - for an account are less than these thresholds, the data sample for that account is not written. The default value for these thresholds is 0.

```
Enable VOLUME metric (based on XFC stats) [Yes]
        On Volumes  [ALL] *DATA*
        ... IO request threshold [0.00 Req/sec]  (with /ADVANCED qualifier only)
```

Enter Yes/No to enable/disable the XFCVOLUME metric for this collection profile. The default is Yes.

When enabling the XFCVOLUME metric volume statistics are collected from XFC for each volume known to the XFC.

The user can restrict the data collection to dedicated volumes ("On Volumes" inquiry). Enter all volumes to monitor as a comma separated list. VSI PERFDAT V4.1 and higher versions provide the feature to define exclude lists. Volumes that should be excluded from being monitored have to be preceded with the '<>' or '!=' tag. VSI PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list. The default is ALL (one can also enter an asterisk (*) instead).

If the command is entered with the /ADVANCED qualifier the user is prompted to define an I/O request threshold. If the average I/O request rate within a sample interval to a volume is less than the threshold, the whole data sample for that volume is not written. The default value for this threshold is 0.

```
Enable Volume IO size stats [No] y
```

Prerequisite is that the XFCVOLUME metric has been enabled before. Enter Yes/No to enable/disable the XFCVOLUME.IOSIZE metric for this collection profile. The default is No.

This XFCVOLUME.IOSIZE metric provides statistics per I/O size range and volume

- Single block
- 1-4 block
- 5-8 block
- 9-16 block
- 16-32 block
- 33-64 block
- 65-127 block

Enable FILE metric based on XFC stats [No] y

Prerequisite is that the XFCVOLUME metric has been enabled before. Enter Yes/No to enable/disable the XFCVOLUME.FILE metric for this collection profile. The default is No.

Once the XFCVOLUME.FILE metric is enabled the wizard prompts the user for filter options. Two filter options are available:

- Top I/O rate filter
- Selective file filter

It is recommended to enable one of these filters. If no filter is enabled probably thousands of records are written to the data file that may cause the data file to grow exponentially. This is cumbersome to analyse the data as any statistical method like sorting and correlation applied to this data could take hours.

Top $QIO rate FILE statistics only [Yes]

Enter Yes/No to enable or disable the top I/O rate filter for the XFCVOLUME.FILE metric enter The default is Yes.

With the top $QIO rate filter enabled only the 15 top I/O rate files per volume within a sample interval are stored each sample interval.

The top I/O rate filter and the selective file filter are mutual exclusive. Thus, if you disable the top I/O rate filter the wizard prompts you to enable the selective file filter. Otherwise the selective file filter inquiry (next inquiry) will be skipped.

You have enabled XFCVOLUME.FILE statistics.
Now you are prompted to enter the file that contains the file filter list for selective XFCVOLUME.FILE monitoring. All files (file specification syntax = DIRECTORY command) that are listed within that filter file are subject for XFCVOLUME.FILE monitoring.
Enter the keyword ALL or the asterisk (*) to disable selective XFCVOLUME.FILE monitoring

Filter List File name [ ]: 1DKB100:[PERFDAT.CFG]SEARCHFILE.DAT

The wizard prompts you to enable the selective file filter only if you have disabled the top I/O rate filter (previous inquiry) since these two file filters are mutual exclusive.

If you want to enable selective file filtering for this metric enter the file that contains the file filter list. This filter file is a text file that contains all files (filenames) to monitor. You can use wildcard

characters for the file names in the filter file as supported by the DIRECTORY command. Up to 2048 files are supported for selective file filtering. If the files defined by the filter file exceeds the supported number of files the first 2048 files specified are monitored all other files are ignored. In that case the user is informed via OPCOM messages.

The file PERFDAT$CFG:PERFDAT_FILEFILTER.TEMPLATE provided by the installation procedure of PERFDAT is an example of such a filter file.

When selective file filtering is enabled the OpenVMS data collector reads the filter file whenever a new performance data collection file is created (re-starting the performance data collection or at day change) and caches the file ID's of the files that matches the entries in the filter file. I/O's are only counted for files that matches the cached file ID's. Due to that implementation statistics are not collected for files created between adjacent performance data file creations until the next performance data file is created (day change, (re) start of the performance collection) even if these files match the entries in the filter file since the file ID's of these files are not stored in the file ID cache.

Thus, it is recommended to enable selective file filtering for persistent files (e.g. database files) only. Files listed in the filter file not stored on volumes listed in the volume filter list of the XFCVOLUME metric ("On Volumes" inquiry) are not monitored.

If you enter the keyword ALL or an asterisk (*) at the "Filter List File name" prompt selective file filtering will be disabled (= all files will be monitored).

Open File Collection only [Yes] (with /ADVANCED qualifier only)
... IO request threshold [0.00 Req/sec] (with /ADVANCED qualifier only)

If the command is entered with the /ADVANCED qualifier the user is prompted to define the collection mode and an I/O request threshold.

Two possible collection modes ("Open File Collection only" inquiry) exist:

- Open file mode
  If one enters Yes to the prompt the 'open file' mode is enabled. The 'open file' mode just collects statistics for open files cached in XFC. The advantage of this mode is that the data collector consumes less system resources. The disadvantage is that a file has to be open at least two times the sample interval to get valid data.
- All file mode
  If you enter No to the prompt the 'all file' mode is enabled. The 'all file' mode collects statistics for all files cached in XFC regardless if the files are open or closed. The disadvantage of

this mode is that the data collector consumes more system resources. The advantage is that there is no delay in collecting performance data for any file whenever XFC counters change.

Regardless of the enabled file mode, it is a general rule that statistics of files known to the XFC cache are collected only if data is actually accessed during the sample interval (i.e. any I/O activity on that file).

If the average I/O request rate within a sample interval to a file cached in XFC is less than the threshold, the whole data sample for that file is not written. The default value for this threshold is 0.

Enable File IO size stats [No] y

Prerequisite is that the XFCVOLUME metric has been enabled before. Enter Yes/No to enable/disable the XFCFILE.FILE.IOSIZE metric for this collection profile. The default is No.

This XFCVOLUME.FILE.IOSIZE metric provides statistics per I/O size range and file.

- Single block
- 1-4 block
- 5-8 block
- 9-16 block
- 16-32 block
- 33-64 block
- 65-127 block

Enable DEVICE metric [Yes]
On DEVICES (eq. DKA100, DG*, TN*) [*$D*,*DS*]
... IO request threshold [0.00 Req/sec] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the DEVICE metric for this collection profile. The default is Yes.

With the DEVICE metric enabled performance data for all class devices available on OpenVMS can be collected.

One can restrict the data collection to dedicated devices ("On DEVICES" inquiry). Enter all the devices that should be monitored as a comma separated list. VSI PERFDAT V4.1 and higher versions provide the feature to define exclude lists. Devices that should be excluded from being monitored have to be preceded with the '<>' or '!=' tag. VSI PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list. The default is *$D*,*DS* (all disk and shadow devices).

If the command is entered with the /ADVANCED qualifier the user is prompted to define an I/O request threshold. If the average I/O request rate within a sample interval to a device is less than the threshold, the whole data sample for that device is not written. The default value for this threshold is 0.

Enable IO size metric on selected FOD devices [No] y

Prerequisite is that the DEVICE metric has been enabled before. Enter Yes/No to enable/disable the DEVICE.IOSIZE metric for this collection profile. The default is No.

This DEVICE.IOSIZE metric provides statistics per I/O size range and device.

- Single block
- 1-4 block

- 5-8 block
- 9-16 block
- 16-32 block
- 33-64 block
- 65-127 block

This metric is only valid for file oriented devices (FOD). If the device filter list entered (see "Enable DEVICE metric" inquiry) contains no file oriented device(s) no data are collected.

Enable FILE metric on selected FOD devices [No] y

Prerequisite is that the DEVICE metric has been enabled before. Enter Yes/No to enable / disable the DEVICE.FILE metric for this collection profile. The default is No.

The DEVICE.FILE metric provides I/O statistics per file and device. This metric is only valid for file-oriented devices (FOD).

Once the DEVICE.FILE metric is enabled the wizard prompts the user for filter options. Two filter options are available:

- Top I/O rate filter
- Selective file filter

It is recommended to enable one of these filters. If no filter is enabled probably thousands of records are written to the data file that may cause the data file to grow exponentially. This is cumbersome to analyse the data as any statistical method like sorting and correlation applied to this data could take hours.

Top $QIO rate FILE statistics on selected FOD devices [Yes]

Enter Yes/No to enable or disable the top I/O rate filter for the DEVICE.FILE metric enter The default is Yes.

With the top $QIO rate filter enabled only the 15 top I/O rate files per device within a sample interval are stored each sample interval.

The top I/O rate filter and the selective file filter are mutual exclusive. Thus, if you disable the top I/O rate filter the wizard prompts you to enable the selective file filter. Otherwise the selective file filter inquiry (next inquiry) will be skipped

```
You have enabled DEVICE.FILE statistics.
Now you are prompted to enter the file that contains the file filter list for selective
DEVICE.FILE monitoring. All files (file specification syntax = DIRECTORY command) that are
listed within that filter file are subject for XFC.FILE monitoring.
Enter the keyword ALL or the asterisk (*) to disable selective DEVICE.FILE monitoring

        Filter List File name  [ ]: $1$DKB100:[PERFDAT.CFG]SEARCHFILE.DAT
```

The wizard prompts you to enable the selective file filter only if you have disabled the top I/O rate filter (previous inquiry) since these two file filters are mutual exclusive.

If you want to enable selective file filtering for this metric enter the file that contains the file filter list. This filter file is a text file that contains all files (filenames) to monitor. You can use wildcard characters for the file names in the filter file as supported by the DIRECTORY command. Up to 2048 files are supported for selective file filtering. If the files defined by the filter file exceeds the supported number of files the first 2048 files specified are monitored all other files are ignored. In that case the user is informed via OPCOM messages.

The file PERFDAT$CFG:PERFDAT_FILEFILTER.TEMPLATE provided by the installation procedure of PERFDAT is an example of such a filter file.

When selective file filtering is enabled the OpenVMS data collector reads the filter file whenever a new performance data collection file is created ((re)starting the performance data collection or at day change) and caches the file ID's of the files that matches the entries in the filter file. I/O's are only counted for files that matches the cached file ID's. Due to that implementation statistics are not collected for files created between adjacent performance data file creations until the next performance data file is created (day change, (re) start of the performance collection) even if these files match the entries in the filter file since the file ID's of these files are not stored in the file ID cache.

Thus, it is recommended to enable selective file filtering for persistent files (e.g. database files) only. Files listed in the filter file not stored on devices listed in the device filter list of the DEVICE metric ("On DEVICES" inquiry) are not monitored.

If you enter the keyword ALL or an asterisk (*) at the "Filter List File name" prompt selective file filtering will be disabled (= all files will be monitored).

... IO request threshold [0.00 Req/sec] (with /ADVANCED qualifier only)

If the command is entered with the /ADVANCED qualifier the user is prompted to define an I/O request threshold. If the average I/O request rate within a sample interval to a file is less than the threshold the whole data sample for the file is not written. The default value for this threshold is 0.

Do you want to enable per PROCESS collection on selected devices [No] y

Prerequisite is that the DEVICE metric has been enabled before. Enter Yes/No to enable / disable the DEVICE.PPROCESS metric for this collection profile. The default is No.

The DEVICE.PROCESS metric provides I/O statistics per process and device. This metric is only valid for file-oriented devices (FOD).

Once the DEVICE.PROCESS metric is enabled the wizard prompts the user for filter options. Two filter options are available:

- Top I/O rate filter
- Selective process filter

It is recommended to enable one of these filters. If no filter is enabled probably thousands of records are written to the data file that may cause the data file to grow exponentially. This is cumbersome to analyse the data as any statistical method like sorting and correlation applied to this data could take hours.

Top $QIO rate PROCESS statistics on selected devices [Yes]

Enter Yes/No to enable or disable the top I/O rate filter for the DEVICE.PROCESS metric. The default is Yes.

With the top $QIO rate filter enabled only the 15 top I/O rate processes per device within a sample interval are stored each sample interval.

The top I/O rate filter and the selective process filter are mutual exclusive. Thus, if you disable the top I/O rate filter the wizard prompts you to enable the selective process filter. Otherwise the selective process filter inquiry (next inquiry) will be skipped.

On Process [ALL] *RDB*,*SX*

The wizard prompts you to enable the selective process filter only if you have disabled the top I/O rate filter (previous inquiry) since these two filters are mutual exclusive.

Enter all processes to monitor per device as a comma separated list. VSI PERFDAT V4.1 and higher versions provide the feature to define exclude lists. Devices that should be excluded from being monitored have to be preceded with the '<>' or '!=' tag. VSI PERFDAT V3.0 and

higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list. The default is ALL (one can also enter an asterisk (*) instead).

... IO request threshold [0.00 Req/sec] (with /ADVANCED qualifier only)

If the command is entered with the /ADVANCED qualifier the user is prompted to define an I/O request threshold. If the average I/O request rate of a process to a dedicated device within a sample interval is less than the threshold, the whole data sample for that process and device is not written. The default value for this threshold is 0.

Enable per FILE collection for each Process collection on FOD devices [No] y

Prerequisite is that the DEVICE.PROCESS metric has been enabled before. Enter Yes/No to enable/disable DEVICE.PROCESS.FILE metric for this collection profile. The default is No.

This DEVICE.PROCESS.FILE metric provides I/O statistics per file, process and device. This metric is only valid for file oriented devices (FOD). The selective process filter of the DEVICE.PROCESS metric also applies to this metric.

Once the DEVICE.PROCESS.FILE metric is enabled the wizard prompts the user for filter options. Two filter options are available:

- Top I/O rate filter
- Selective file filter

It is recommended to enable one of these filters. If no filter is enabled probably thousands of records are written to the data file that may cause the data file to grow exponentially. This is cumbersome to analyse the data as any statistical method like sorting and correlation applied to this data could take hours.

Top $QIO rate FILE statistics for each Process on FOD devices [Yes]

Enter Yes/No to enable or disable the top I/O rate filter for the DEVICE.PROCESS.FILE metric. The default is Yes.

With the top $QIO rate filter enabled only the 5 top I/O rate files per process within a sample interval are stored each sample interval.

The top I/O rate filter and the selective file filter are mutual exclusive. Thus, if you disable the top I/O rate filter the wizard prompts you to enable the selective process filter. Otherwise the selective file filter inquiry (next inquiry) will be skipped.

You have enabled DEVICE.PROCESS.FILE statistics.
Now you are prompted to enter the file that contains the file filter list for selective DEVICE.PROCESS.FILE monitoring. All files (file specification syntax = DIRECTORY command) that are listed within that filter file are subject for

DEVICE.PROCESS.FILE monitoring.
Enter the keyword ALL or the asterisk (*) to disable selective DEVICE.PROCESS.FILE monitoring

Filter List File name [ ]: *

The wizard prompts you to enable the selective file filter only if you have disabled the top I/O rate filter (previous inquiry) since these two file filters are mutual exclusive.

If you want to enable selective file filtering for this metric enter the file that contains the file filter list. This filter file is a text file that contains all files (filenames) to monitor. You can use wildcard characters for the file names in the filter file as supported by the DIRECTORY command. Up to 2048 files are supported for selective file filtering. If the files defined by the filter file exceeds the supported number of files the first 2048 files specified are monitored all other files are ignored. In that case the user is informed via OPCOM messages.

The file PERFDAT$CFG:PERFDAT_FILEFILTER.TEMPLATE provided by the installation procedure of PERFDAT is an example of such a filter file.

When selective file filtering is enabled the OpenVMS data collector reads the filter file whenever a new performance data collection file is created (re-starting the performance data collection or at day change) and caches the file ID's of the files that matches the entries in the filter file. I/O's are only counted for files that matches the cached file ID's. Due to that implementation statistics are not collected for files created between adjacent performance data file creations until the next performance data file is created (day change, (re) start of the performance collection) even if these files match the entries in the filter file since the file ID's of these files are not stored in the file ID cache.

Thus, it is recommended to enable selective file filtering for persistent files (e.g. database files) only. Files listed in the filter file not stored on devices listed in the device filter list of the DEVICE metric ("On DEVICES" inquiry) are not monitored.

If you enter the keyword ALL or an asterisk (*) at the "Filter List File name" prompt selective file filtering will be disabled (= all files will be monitored).

... IO request threshold [0.00 Req/sec] (with /ADVANCED qualifier only)

If the command is entered with the /ADVANCED qualifier the user is prompted to define an I/O request threshold. If the average I/O request rate of a process to a dedicated file within a sample interval is less than the threshold, the whole data sample for that process and file is not written. The default value for this threshold is 0.

You have enabled File and/or Process File statistics.
It is strongly recommended to exclude all the backup and defragmenter process from file monitoring. If you have backup and/or defragmenter software running on your system and you do not exclude these processes from file monitoring it is very likely that overall system performance may suffer during backup / defragmenting activities!!
Enter NONE for empty process exclusion list

Process to be excluded from file monitoring [] *MDMS*,*NSR*,*ABS*

If you have previously enabled the metric DEVICE.FILE and/or DEVICE.PROCESS.FILE and you have not enabled the top I/ rate filter for one of these metrics you are now prompted to define an exclude list of processes. I/O's done by the processes listed within this exclude list are not counted in the DEVICE.FILE and DEVICE.PROCESS.FILE metric.

Enter the processes as a comma separated list as shown in the example. PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within each string element of the comma separated list.

If you do not want to define a process exclude list enter 'NONE'.

Note

Especially if you have backup and/or de-fragmenting processes running on you system it may happen, that many to all files on your system are accessed during a short time of period. If you do not exclude these processes from data collection, huge amount of data have to be written per sample interval and consequently overall system performance may suffer.

Enable device capacity and path info metrics [Yes]

Enter Yes/No to enable/disable

- DEVICE.CAPACITY
- DEVICE.PATH
- IOPATHES

metrics. The default is Yes.

Enable LAN metric [Yes]
... Octets threshold [0.00 KByte/sec]

Enter Yes/No to enable/disable the LANADAPTER metric for this collection profile. The default is Yes.

If the command is entered with the /ADVANCED qualifier the user is prompted to define an octet rate threshold. If the average octet rate (receives & transmits) of a LAN adapter within a sample interval is less than the threshold, the whole data sample for this LAN adapter is not written. The default value for this threshold is 0.

Enable LAN Device metric [Yes]

Enter Yes/No to enable/disable the LANADAPTER.DEVICE metric for this collection profile. The default is Yes. The prerequisite is that the LAN metric has been enabled before.

Enable LAN PROTOCOL metric [Yes]

Enter Yes/No to enable/disable the LANPROTOCOL metric for this collection profile. The default is Yes. The prerequisite is that the LAN metric has been enabled before.

Enable SCS metric [Yes]
... MSG request threshold [0.00 Req/sec]
... ERROR rate threshold [0.00 1/sec]

Enter Yes/No to enable/disable the SCSPORT, SCSPORT.VC, SCSPORT.VC.CHANNEL metrics for this collection profile. The default is Yes.

If the command is entered with the /ADVANCED qualifier the user is prompted to define message and error rate thresholds. If both, the average total message and the average total error rate of a SCS port within a sample interval are less than these thresholds, the whole data sample for this SCS port (including the virtual circuit and channel statistics) is not written. The default value for these thresholds is 0.

Collection profile wizard forTru64

When adding or modifying collection profiles for Tru64 the collection profile wizard prompts for:

WELCOME to TRU64 performance data profile wizard

Collection sample interval [120 sec]

Enter the collection interval in seconds.

Enable metric TRU64_CPU [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_CPU metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_DISK [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_DISK metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_PROCESS [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_PROCESS metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_DEAMON [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_DEAMON metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_USER [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_USER metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_FILESYS [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_FILSYS metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_SYSTEM [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_SYSTEM metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_NIC [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_NIC (network interconnect) metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric TRU64_IP [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the TRU64_IP metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Collection profile wizard for Brocade switches

When adding or modifying collection profiles for Brocade switches the collection profile wizard prompts for:

WELCOME to BROCADE performance data profile wizard

Collection sample interval [120 sec]

Enter the collection interval in seconds.

Enable metric PORT [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the PORT metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SYSTEM [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SYSTEM metric for this collection profile. The default is Yes.

If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SYSTEM.TEMPERATURE [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SYSTEM.TEMPERATURE metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SYSTEM.FAN [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SYSTEM.FAN metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Collection profile wizard for EVA (HP StorageWorks Virtual Array)

When adding or modifying collection profiles for EVA systems the collection profile wizard prompts for:

WELCOME to EVA performance data profile wizard

Collection sample interval [120 sec]

Enter the collection interval in seconds.

Enable metric ARRAY [Yes]

Enter Yes/No to enable/disable the ARRAY metric for this collection profile. The default is Yes.

Enable metric CTRL [Yes]

Enter Yes/No to enable/disable the CTRL metric for this collection profile. The default is Yes.

Enable metric CTRL.PORT [Yes]

Enter Yes/No to enable/disable the CTRL.PORT metric for this collection profile. The default is Yes.

Enable metric CTRL.HOSTCONN[Yes]

Enter Yes/No to enable/disable the CTRL.HOSTCONN metric for this collection profile. The default is Yes.

Enable metric DISKGROUP [Yes]

Enter Yes/No to enable/disable the DISKGROUP metric for this collection profile. The default is Yes.

Enable metric DISKGROUP.VDISK [Yes]

Enter Yes/No to enable/disable the DISKGROUP.VDISK metric for this collection profile. The default is Yes.

Enable metric DISKGROUP.PDISK [Yes]

Enter Yes/No to enable/disable the DISKGROUP.PDISK metric for this collection profile. The default is Yes.

Enable metric DRM.TUNNEL [Yes]

Enter Yes/No to enable/disable the DRM.TUNNEL metric for this collection profile. The default is Yes.

Collection profile wizard for Solaris

When adding or modifying collection profiles for Solaris the collection profile wizard prompts for:

WELCOME to SOLARIS performance data profile wizard

Collection sample interval [120 sec]

Enter the collection interval in seconds.

Enable metric SUN_DEVICE [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_DEVICE metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SUN_PROCESS [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_PROCESS metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SUN_DEAMON [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_DEAMON metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SUN_NIC [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_NIC(network interconnect) metric for this collection profile. The default is Yes.

If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SUN_IP [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_IP metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SUN_TCP [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_TCP metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SUN_FILESYS [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_FILESYS metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric SUN_SYSTEM [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the SUN_SYSTEM metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Collection profile wizard for Linux

When adding or modifying collection profiles for Linux the collection profile wizard prompts for:

WELCOME to LINUX performance data profile wizard

Collection sample interval [120 sec]

Enter the collection interval in seconds.

Enable metric LINUX_PROCESS [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the LINUX_PROCESS metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric LINUX_DEAMON [Yes]

RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the LINUX_DEAMON metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric LINUX_NIC [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the LINUX_NIC(network interconnect) metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric LINUX_IP [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the LINUX_IP metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric LINUX_TCP [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the LINUX_TCP metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric LINUX_FILESYS [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the LINUX_FILESYS metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

Enable metric LINUX_SYSTEM [Yes]
RemoteSNMPPort [161] (with /ADVANCED qualifier only)

Enter Yes/No to enable/disable the LINUX_SYSTEM metric for this collection profile. The default is Yes.
If the /ADVANCED qualifier is applied the wizard prompts you to enter remote SNMP server port to fetch the data of the metric.

For detailed information about the statistics available for all the metrics that can be enabled by the collection profile wizards please see the manual VSI OpenVMS PERFDAT – Architecture and Technical Description.

Qualifier

/ADVANCED

/NOADVANCED (default)

If the /ADVANCED qualifier is applied to configure an OpenVMS collection profile the profile wizard prompts you to enter threshold values whenever you enable one of the metrics listed below:

- CPU
- PROCESS
- USER
- IMAGE
- ACCOUNT
- XFCVOLUME
- XFCVOLUME.FILE
- DEVICE
- DEVICE.FILE
- DEVICE.PROCESS
- DEVICE.PROCESS.FILE
- LANADAPTER
- SCSPORT

Thresholds can be used to reduce the data rate to the collection file. For more information about setting thresholds see the DESCRIPTION section.

If the /ADVANCED qualifier is applied to configure an SNMP extension collection profile the profile wizard prompts you to enter the remote SNMP server listener port per metric to fetch the performance data.

This is of special interest if the SNMP server listener port on a remote system that provides performance data via SNMP is not the default SNMP listener port (161) or if several SNMP servers running on the remote system are listening on different ports providing different kind of performance data.

If the /ADVANCED qualifier is omitted or the command is entered with /NOADVANCED the user is neither prompted to enter threshold values for OpenVMS collection profiles nor to define the remote SNMP server listener port per metric for non-OpenVMS collection profiles. (All threshold for OpenVMS collection profiles = 0, remote SNMP server ports for all metrics of non-OpenVMS collection profiles = 161).

If you configure an EVA extension collection profile the /ADVANCED qualifier is ignored.

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the newly added collection profile is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming

interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to add a new collection profile valid for non-OpenVMS systems to the collection profile table of the VSI PERFDAT configuration database the /OS_TYPE qualifier is mandatory.

If you are adding a new OpenVMS collection profile you can omit the qualifier since OpenVMS is the default, and the OpenVMS profile wizard is started.

The supported keywords for the value for the qualifier are:

- OpenVMS -> The node is an OpenVMS node.
- Tru64 -> The node is a Tru64 node.
- Brocade -> The node is a Brocade switch.
- EVA -> HP StorageWorks Virtual Array.
- Solaris -> The node is a Solaris system.
- Linux -> The node is a Linux system.
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of VSI PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Examples

Command to invoke the OpenVMS configuration wizard to add the new collection profile NEW_ONE for OpenVMS:

```
PERFDAT_MGR>ADD PROFILE NEW_ONE
or
PERFDAT_MGR> ADD PROFILE NEW_ONE/OS_TYPE=OPENVMS
```

Command to invoke the Tru64 configuration wizard to add the new collection profile NEW_ONE for Tru64:

```
PERFDAT_MGR>ADD PROFILE NEW_ONE/OS_TYPE=TRU64
```

Command to invoke the BROCADE configuration wizard to add the new collection profile NEW_ONE for BROCADE:

```
PERFDAT_MGR>ADD PROFILE NEW_ONE/OS_TYPE=BROCADE
```

Command to invoke the EVA configuration wizard to add the new collection profile NEW_ONE for HP StorageWorks Virtual Arrays:

```
PERFDAT_MGR>ADD PROFILE NEW_ONE/OS_TYPE=EVA
```

As you can see from these examples the same profile name can be used when defining profiles for different OS types.

ADD REPORT

This command creates a new entry (report) in the report profile table of the PERFDAT configuration database.

Format

ADD REPORT *report_name*

Parameter

report_name
Defines the name of the newly created report profile. The report name references the report profile in the report profile table of the PERFDAT configuration database.

If you add a new report profile the *report_name* has to be unique for the system or application defined by the /OS_TYPE qualifier, but you can reuse the same *report_name* if you define profiles for different systems or applications.

This parameter is mandatory. Wildcard characters within the *report_name* string are not permitted.

Description

Trend, capacity and baseline reports are extracted from performance data either via the auto-trend engine or manually via DQL$. In either case these reports are profile controlled. These report profiles are stored in the report profile table of the PERFDAT configuration database. This command adds a new report profile in that table by invoking the report profile wizard.

Report profile wizard

With the report profile wizard the user defines the trend, capacity and baseline reports. The wizard does not differ depending on the /OS_TYPE qualifier.

The report profile wizard consists of two sections

- Report header section (Trend, capacity and baseline report)
- Report data file layout and statistics section

Report header section

When one applies the ADD REPORT or the MODIFY REPORT command the report wizard starts up and prompts for the type of report to define

WELCOME to OPENVMS performance data capacity report wizard

Report Type Trend Report / Capacity Report / Baseline deviation Report (T/C/B/?=Help) [T]

Enter the report type you want to add/modify. For more information about the different report types enter '?' or refer to the manual VSI OpenVMS PERFDAT– Architecture and Technical Description. The default is (T) trend report.

Default Source Collection Profile (type ? to show available Profiles) [DEFAULT]

Enter the default source collection profile. If you extract this report manually using DQL$, and you do not enter the source collection database alias in the DQL query, the trend engine automatically uses that performance collection database as source database that matches the profile defined in this field. This field has no effect on the auto-trend engine, since the source collection database is defined by the collection profile field in the auto-start table of the PERFDAT configuration database.

This report profile is valid for node(s) [*]:

Enter the nodes as a comma separated list this report profile can be applied to. If you enter the asterisk (*) character or the keyword 'ALL' the report is valid for all nodes. This feature, first introduced with PERFDAT V3.3, enables the user to define node specific reports in a cluster even if the cluster members access the same PERFDAT configuration database.

Auto-enable the report (automatically done by Perfdat Report Engine) [Yes]

Yes	this report is processed by the auto-trend engine
No	this report will not be processed by the auto-trend engine

Depending if you are about to configure a trend (T), a capacity (C) or baseline report (B) the wizard prompts for different options.

Trend report

The trend report specific options are

Period captured by single auto-report (Day, Week, Month, Quarter, Year) [] WEEK

Enter the time period to be captured within a single report data file. You can enter one of the predefined time periods only:

- Day
- Week
- Month
- Quarter
- Year

For more information please refer to the manual VSI OpenVMS PERFDAT– Architecture and Technical Description.

Time compression [1800 sec]

Enter the time compression in seconds. Statistics defined in the report data file layout and statistics section are averaged according to time period enter herein. The default value is 1800 sec. This means that all statistics are averaged on a 30 minutes basis.

Calculate Avg/Max/Std for all statistics (Yes = Avg/Max/Std, No = Avg) [No]

One can choose if the trend engine should calculate average, standard deviation and maximum values for every statistics defined in the report data file layout and statistics section when extracting this report or not.
If the full calculation option (Avg/Max/Std) is enabled the processing time of the report increases. On the other hand, the information that one receives will show how much the data within the time compression range varies from the average value. The default value is No.

Capacity report

The capacity report specific options are

Period captured by single auto-report (Day, Week, Month, Quarter, Year) [] WEEK

Enter the time period to be captured within a single report data file. You can enter one of the predefined time periods only:

- Day
- Week
- Month
- Quarter
- Year

For more information please refer to the manual VSI OpenVMS PERFDAT– Architecture and Technical Description.

Integral based calculation (No = arithmetic mean value) [No]

One can choose to calculate the average value based on the time periods entered below integral or arithmetic based. The result of calculating the average integral or arithmetic based is identical if no data is missing within the time periods selected below for a dedicated element. In the case data does not exist the whole time period (e.g. may be the case for process statistics) the values differ since the arithmetic mean value is defined as the sum of data divided by the count of data. In contrast the integral mean value is defined to be the sum of (data value * sample interval) divided by the time range.

Define the time range(s) (up to 5) for the capacity report -> Enter time (no date) in VMS time format

Start Time of TimeRange [1] [00:00:00]
Stop Time of TimeRange [1] [00:00:00] 05:00
Start Time of TimeRange [2] [05:00:00] 10:00
Stop Time of TimeRange [2] [10:00:00] 16:00
Start Time of TimeRange [3] [16:00:00]
Stop Time of TimeRange [3] [16:00:00]
Start Time of TimeRange [4] [16:00:00]
Stop Time of TimeRange [4] [16:00:00]
Start Time of TimeRange [5] [16:00:00]
Stop Time of TimeRange [5] [16:00:00]

A capacity report calculates just one average value for the busy times of each statistics defined in the report data file layout and statistics section. To define the busy times of the system one can enter up to 5 time periods. One is prompted to enter all time periods. Time periods that have the same start and stop time are ignored for calculation. Thus, if one has only 2 time periods of interest enter the start and stop time of these two time periods as shown in the example above, and leave the default value for the rest of the periods (Period 3 – 5) unchanged.

Do you want to calculate day-to-day deviation too [Yes]

Enable a day-to-day deviation report based on this capacity report. For more information about day-to-day deviation reports please refer to the manual VSI OpenVMS PERFDAT– Architecture and Technical Description.

Baseline Report

The baseline report specific inquiries are similar to these of the capacity report. The difference is that there is no prompt for a time period to be captured within a single report data file, since baseline reports are extended as long as the defined baseline is valid. In addition a baseline report cannot be flagged to extract a day-to-day deviation report based on the baseline report definition.

The baseline report specific options are:

Integral based calculation (No = arithmetic mean value) [No]

One can choose to calculate the average value based on the time periods entered below integral or arithmetic based. The result of calculating the average integral or arithmetic based is identical if no data is missing within the time periods selected below for a dedicated element. In the case data does not exist the whole time period (e.g. may be the case for process statistics) the values differ since the arithmetic mean value is defined as the sum of data divided by the count of data. In contrast the integral mean value is defined to be the sum of (data value * sample interval) divided by the time range.

Define the time range(s) (up to 5) for the capacity report -> Enter time (no date) in VMS time format

Start Time of TimeRange [1] [00:00:00]
Stop Time of TimeRange [1] [00:00:00] 05:00
Start Time of TimeRange [2] [05:00:00] 10:00
Stop Time of TimeRange [2] [10:00:00] 16:00
Start Time of TimeRange [3] [16:00:00]
Stop Time of TimeRange [3] [16:00:00]
Start Time of TimeRange [4] [16:00:00]
Stop Time of TimeRange [4] [16:00:00]
Start Time of TimeRange [5] [16:00:00]
Stop Time of TimeRange [5] [16:00:00]

A baseline report calculates just one average value for the busy time of each statistics defined in the report data file layout and statistics section. To define the busy times of the system one can enter up to 5 time periods. One is prompted to enter all time periods. Time periods that have the same start and stop time are ignored for calculation. Thus, if you have only 2 time periods of interest enter the start and stop time of these two time periods as shown in the example above, and leave the default value for the rest of the periods (Period 3 – 5) unchanged.

Report data file layout and statistics configuration

In the report data file layout and statistics configuration section one has to define all the statistics and elements that will be included in the report. One has the option to redefine the metrics and in addition, one can define if the statistics will be calculated per element or stacked.

As described in the manual VSI OpenVMS PERFDAT – Architecture and Technical Description a data file consists of metrics (comparable to tables of a database). Each metric exists of 1 to n elements, and each element consists of 1 to m statistics. When you enter the data file layout and statistics configuration section you are prompted to enter in order:

1. The name of the metric (target metric) to be created in the report data file. The reason for defining target metric names explicitly is that you can map statistics that belong to the same source metric to different target metrics.
2. The name of the metric within the source collection database that is the data source for the metric defined above. To view all predefined metrics available for the operating system defined by the /OS_TYPE qualifier enter a question mark (?) at the prompt. The wizard checks automatically if the source metric exists in the record descriptor table of the PERFDAT configuration database. If it does not the user is informed and asked if the input shall be accepted anyway.
3. The statistics of the source metric that should be included. You can enter any valid statistics stored in the source database as well as any user defined (calculated) statistics valid for the source metric stored in the stored procedure table of the PERFDAT configuration database (for more information about user defined statistics please refer to the manuals VSI OpenVMS PERFDAT– Architecture and Technical Description and VSI OpenVMS PERFDAT– DQL$ Reference Manual). To view the available statistics of the selected source metric enter a question mark (?) at the prompt. Enter only one statistics per inquiry. You are prompted to enter statistics as long as you do not terminate the inquiry or you have entered the maximum number of 32 statistics configurable per target metric. In order to terminate the statistic inquiry loop press return when you are prompted for input. After each input the wizard checks automatically if the defined statistic exists in the record descriptor table of the PERFDAT configuration database. If it does not one is prompted to re-enter a valid statistic.
4. Enter the elements of the source metric to be included. Enter only one element per inquiry. As with the statistics inquiry you are prompted to enter elements as long as you do not terminate the inquiry or the maximum number of 16 element entries configurable per target metric has been entered. In order to terminate the element inquiry loop press return when you are prompted for input. No element existence check will be performed.
   PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within each element string.

Earlier versions of PERFDAT (V2.3 and older) provide only limited wildcard support. Only the asterisk (*) wildcard character at the start and the end of each element string was supported in order to address a group of elements with one element entry. You were not allowed to use wildcards within the element string.

5. At least you have to choose if the selected statistics shall be calculated per element, or stacked. If you select the stacked option, you have to enter the element name to use for the stacked calculation, since in this case an n to 1 relationship between the source elements and the target element exists. Consequently the trend engine can't derive the target element name from the source element names entered. Thus, it is up to the user to define it.

The inquiry sequence listed above is repeated until you press return when you are prompted for entering a new target metric name (see 1), or you have already configured the maximum number of target metrics allowed. The limit is 16 metrics.

Qualifier

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the newly added report is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to add a new report profile valid for non-OpenVMS systems to the report profile table of the VSI PERFDAT configuration database the /OS_TYPE qualifier is mandatory.

If you are adding a new report profile that is valid for OpenVMS you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS -> The node is an OpenVMS node.
- Tru64 -> The node is a Tru64 node.
- Brocade -> The node is a Brocade switch.
- RDB -> Imported RDB database performance data
- EVA -> HP StorageWorks Virtual Array.
- Solaris -> The node is a Solaris system.
- Linux -> The node is a Linux system.
- CACHE -> Imported CACHE database performance data
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Examples

Example 1:

This example shows how to configure a capacity report named NEWONE valid for OpenVMS:

- The default source collection profile is DEFAULT.
- This report shall be automatically processed by the auto-trend engine on node BCSXTC and VMSTM1.
- The time period captured within a single report data file = one week.
- Two time ranges for the busy times of the system are configured (0000 – 0500, 1000 – 1600).
- No day-to-day deviation report shall be created based on that report definition.
- The report consists of 3 target metrics derived from the SYSTEM and the PROCESS metric.
    - The target metrics PROCESS and PRC_STACK maps the same statistics (CPU load, DIO, BIO) of the same source metric PROCESS. For both target metrics the element list includes all elements of the source metric since a wildcard is entered at the element inquiry. Thus, the target metric PROCESS contains as many elements as the source metric. For metric PRC_STACK the stacked option is used. Thus, the target metric PRC_STACK contains only one element. The name of that element has to be defined manually – in this case we choose "PRC_SUM".
    - The target metric SYSTEM maps the CPU load, direct I/O and buffered I/O rate form the source metric SYSTEM. The element list includes all elements of the source metric since a wildcard is entered at the element inquiry.

```
PERFDAT_MGR>ADD REPORT NEWONE

    WELCOME to OPENVMS performance data capacity report wizard

Report Type Trend Report / Capacity Report / Baseline deviation Report (T/C/B/?=Help) [T] C
Default Source Collection Profile (type ? to show available Profiles) [DEFAULT]↲

This report profile is valid for node(s) [*]:BCSXTC, VMSTM1↲
Auto-enable the report (automatically done by Perfdat Report Engine) [Yes] ↲
Period captured by single auto-report (Day, Week, Month, Quarter, Year) [] WEEK

Integral based calculation (No = arithmetic mean value) [No] ↲

Define the time range(s) (up to 5) for the capacity report -> Only enter time (no date) in VMS
time format

Start Time of TimeRange [1] [00:00:00]↲
Stop Time of TimeRange [1] [00:00:00] 05:00
Start Time of TimeRange [2] [05:00:00] 10:00
Stop Time of TimeRange [2] [10:00:00] 16:00
Start Time of TimeRange [3] [16:00:00]↲
Stop Time of TimeRange [3] [16:00:00]↲
Start Time of TimeRange [4] [16:00:00]↲
```

```
Stop Time of TimeRange [4] [16:00:00]↵
Start Time of TimeRange [5] [16:00:00]↵
Stop Time of TimeRange [5] [16:00:00]↵
Do you want to calculate day-to-day deviation too [Yes] No

Enter Metrix Name to add (16 Char max.) [] SYSTEM
    ... derived form Source Metrix (type ? to show all predefined Metrices) [] SYSTEM
    Now enter all stats of SYSTEM to be added [type ? to see available stats, Return to exit]
        Stats to add [] iCpuLoad
        Stats to add [] iIoDios
        Stats to add [] iIoBios
        Stats to add []↵
    Now enter all Elements of SYSTEM to be added [Return to exit]
        Elements to add [] *
        Elements to add []↵
    Stacked calculation of selected Elements [No] ↵

Enter Metrix Name to add (16 Char max.) [] PROCESS
    ... derived from Source Metrix (type ? to show all predefined Metrics) [] PROCESS
    Now enter all stats of PROCESS to be added [type ? to see available stats, Return to exit]
        Stats to add [] iCpuLoad
        Stats to add [] iDio
        Stats to add [] iBio
        Stats to add []↵
    Now enter all Elements of PROCESS to be added [Return to exit]
        Elements to add [] *
        Elements to add []↵
    Stacked calculation of selected Elements [No] ↵

Enter Metric Name to add (16 Char max.) [] PRC_STACK
    ... derived form Source Metric (type ? to show all predefined Metrics) [] PROCESS
    Now enter all stats of PROCESS to be added [type ? to see available stats, Return to exit]
        Stats to add [] iCpuLoad
        Stats to add [] iDio
        Stats to add [] iBio
        Stats to add []↵
    Now enter all Elements of PROCESS to be added [Return to exit]
        Elements to add [] *
        Elements to add []↵
    Stacked calculation of selected Elements [No] Yes
          In stacked mode the primary key values have to be defined by the user
        Selected Metric defines 1 primary keys - you have to define all of them
        Enter Name for Key  [] PRC_SUM

Enter Metrix Name to add (16 Char max.) []↵
```

Example 2:

This example shows how to configure a trend report named NEWONE valid for TRU64:

- The default source collection profile is DEFAULT.
- This report shall be automatically processed by the auto-trend engine for all TRU64 collections performed by the local node (=all TRU64 nodes the local node runs the SNMP data collection and hosts the data).
- The time period captured within a single report data file = one week.
- Time compression is 1800 sec (half an hour).
- Full statistics calculation option is disabled.
- The report consists of 3 target metrics derived from the SYSTEM and the PROCESS metric.
    - The target metrics PROCESS and PRC_STACK maps the same statistics (CPU load, InBlk, OutBlk) of the same source metric TRU64_PROCESS. For both target metrics the element list includes all elements of the source metric since a wildcard is entered at the element inquiry. Thus, the target metric PROCESS contains as many elements as the source metric. For metric PRC_STACK the stacked option is used. Thus, the target metric PRC_STACK contains only one element. The name of that element has to be defined manually – in this case we choose “PRC_SUM”.
    - The target metric SYSTEM maps the CPU load and the page fault rate form the source metric TRU64_SYSTEM. The element list includes all elements of the source metric since a wildcard is entered at the element inquiry.

```
PERFDAT_MGR>ADD REPORT NEWONE/OS_TYPE=TRU64

    WELCOME to OPENVMS performance data capacity report wizard

Report Type Trend Report / Capacity Report / Baseline deviation Report (T/C/B/?=Help) [T] ↵
Default Source Collection Profile (type ? to show available Profiles) [DEFAULT]↵

This report profile is valid for node(s) [*]:↵
Auto-enable the report (automatically done by Perfdat Report Engine) [Yes] ↵
Period captured by single auto-report (Day, Week, Month, Quarter, Year) [] WEEK
Time compression [1800] ↵
Calculate Avg/Max/Std for all statistics (Yes = Avg/Max/Std, No = Avg) [No]: ↵

Enter Metrix Name to add (16 Char max.) [] SYSTEM
    ... derived form Source Metrix (type ? to show all predefined Metrics) [] TRU64_SYSTEM
    Now enter all stats of SYSTEM to be added [type ? to see available stats, Return to exit]
        Stats to add [] CpuLoad
        Stats to add [] PflTot
        Stats to add []↵
    Now enter all Elements of SYSTEM to be added [Return to exit]
        Elements to add [] *
        Elements to add []↵
    Stacked calculation of selected Elements [No] ↵

Enter Metrix Name to add (16 Char max.) [] PROCESS
```

... derived form Source Metrix (type ? to show all predefined Metrics) [] TRU64_PROCESS
Now enter all stats of PROCESS to be added [type ? to see available stats, Return to exit]
Stats to add [] CpuLoad
Stats to add [] InBlk
Stats to add [] OutBlk
Stats to add []↵
Now enter all Elements of PROCESS to be added [Return to exit]
Elements to add [] *
Elements to add []↵
Stacked calculation of selected Elements [No] ↵

Enter Metric Name to add (16 Char max.) [] PRC_STACK
... derived form Source Metric (type ? to show all predefined Metrics) [] TRU64_PROCESS
Now enter all stats of PROCESS to be added [type ? to see available stats, Return to exit]
Stats to add [] iCpuLoad
Stats to add [] InBlk
Stats to add [] OutBlk
Stats to add []↵
Now enter all Elements of PROCESS to be added [Return to exit]
Elements to add [] *↵
Elements to add []↵
Stacked calculation of selected Elements [No] Yes
In stacked mode the primary key values have to be defined by the user
Selected Metric defines 1 primary keys - you have to define all of them
Enter Name for Key [] PRC_SUM

Enter Metrix Name to add (16 Char max.) []↵

CHECK LICENSE

This command reads the license table of the PERFDAT configuration databaseand displays the status of each license key found (type of license, valid/expired).

Format

CHECK LICENSE

Parameter

None

Description:

This command reads the license table of the PERFDAT configuration database and displays the status of each license key found (type of license, valid/expired).

Example

PerfDat_MGR>CHECK LICENSE

PERFDAT_MGR-E-LICEXP, auto registered temp license expired on 6-APR-2005 09:35:11
PERFDAT_MGR-I-LICVALID, valid single node license key found /<license key>/
PERFDAT_MGR-I-LICSNMP, node is licensed to monitor 64 concurrent SNMP agents
PERFDAT_MGR-I-LICIP, node is licensed to monitor 64 concurrent EVA arrays
PERFDAT_MGR-I-LICIP, node is licensed to monitor 64 concurrent IP agents

CHECK ALERT

This command reads the alert definition file defined by the *alert-definition-file* parameter and checks if all alert blocks defined within are valid. If an invalid line item is detected the line item and the line number is displayed.

Format

CHECK ALERT *alert-definition-file*

Parameter

alert-definition-file
This parameter defines the alert definition file to be checked. An alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc.

Wildcard characters within the *alert-definition-file*string are not permitted.

Description

This command reads the alert definition file defined by the *alert-definition-file* parameter and checks if all alert blocks defined within are valid. If an invalid line item is detected the line item and the line number is displayed.

Whenever you define a new alert definition file it is recommended to use this command before enabling online alerting for an active performance data collection using the new alert definition file in order to check for syntax errors and missing items in the alert blocks within the alert definition file.

Example

In this example the default OpenVMS alert definition file PERFDAT$CFG:PERFDAT_ALERT_OPENMS.CFG provided by the installation procedure of PERFDAT is checked.

```
PerfDat_MGR>CHECK ALERTPERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG

PERFDAT-I-ALERTCFG, no warning cmd file defined.
PERFDAT-I-ALERTCFG, no critical cmd file defined.
PERFDAT-I-ALERTCFG, no warning cmd file defined.
PERFDAT-I-ALERTCFG, no critical cmd file defined.
PERFDAT-I-ALERTVALID, alert definition file /PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG/ is
valid
```

COPY PROFILE

This command creates a new entry (profile) in the collection profile table of the PERFDAT configuration database from an existing one.

Format

COPY PROFILE *source-profile-name dest-profile-name*

Parameter

source-profile-name
Defines the name of an existing collection profile in the collection profile table of the PERFDAT configuration database. Wildcard characters are not permitted.

dest-profile-name
Defines the name of the new collection profile to be created in the collection profile table of the PERFDAT configuration database into which the content of the source collection profile is copied. Wildcard characters are not permitted.

Description

This command creates a new entry (profile) in the collection profile table of the PERFDAT configuration database, assigns the name specified by the *dest-profile-name* parameter to that entry and copies the content of the source collection profile defined by the *source-profile-name* parameter.

Qualifier

/OS_TYPE=*system*|*application-name*
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier is mandatory if you want to copy an existing collection profile for a supported non-OpenVMS system.

If you copy an OpenVMS collection profile you can omit the qualifier.

For more information about the /OS_TYPE qualifier see the ADD PROFILE command description.

Examples

Command to copy the existing DEFAULT collection profile valid for OpenVMS to the NEW_ONE collection profile.

```
PerfDat_MGR> COPY PROFILE DEFAULT NEW_ONE
PERFDAT_MGR-I-CFGSUCC, Profile copied
```

Command to copy the existing DEFAULT collection profile valid for Tru64 to the NEW_ONE collection profile.

```
PerfDat_MGR> COPY PROFILE DEFAULT NEW_ONE/OS_TYPE=Tru64
PERFDAT_MGR-I-CFGSUCC, Profile copied
```

Command to copy the existing DEFAULT collection profile valid for EVA systems to the NEW_ONE collection profile.

```
PerfDat_MGR> COPY PROFILE DEFAULT NEW_ONE/OS_TYPE=EVA
PERFDAT_MGR-I-CFGSUCC, Profile copied
```

COPY REPORT

This command creates a new entry (report) in the report profile table of the PERFDAT configuration database from an existing one.

Format

COPY REPORT *source-report-name dest-report-name*

Parameter

source-report-name
Specifies the name of an existing report profile in the report profile table of the PERFDAT configuration database. Wildcard characters are not permitted.

dest-report-name
Specifies the name of the new report profile to be created in the report profile table of the PERFDAT configuration database into which the content of the source report profile is copied. Wildcard characters are not permitted.

Description

Creates a new entry (report) in the report profile table of the PERFDAT configuration database, assigns the name specified by the *dest-report-name* parameter to that entry and copies the content of the source report profile defined by the *source-report-name* parameter.

Qualifier

/OS_TYPE=*system*|*application-name*
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier is mandatory if you want to copy an existing report profile for a supported non-OpenVMS system.

If you copy an OpenVMS report profile you can omit the qualifier.

For more information about the /OS_TYPE qualifier see the ADD REPORT command description.

Examples

Command to copy the existing WEEK report profile valid for OpenVMS to the NEW_ONE report profile.

```
PerfDat_MGR>COPY REPORT WEEK NEW_ONE
PERFDAT_MGR-I-CFGSUCC, Report copied
```

Command to copy the existing DEFAULT report profile valid for Tru64 to the NEW_ONE collection profile.

```
PerfDat_MGR> COPY REPORT WEEK NEW_ONE/OS_TYPE=Tru64
PERFDAT_MGR-I-CFGSUCC, Report copied
```

Command to copy the existing DEFAULT report profile valid for EVA systems to the NEW_ONE collection profile.

```
PerfDat_MGR> COPY REPORT WEEK NEW_ONE/OS_TYPE=EVA
PERFDAT_MGR-I-CFGSUCC, Report copied
```

DEFINE ARCHIVE

Changes the control parameters for the archiving process in the archive control table of the PERFDAT configuration database.

Format

DEFINE ARCHIVE

Parameter

None

Description

The DEFINE ARCHIVE command changes the control parameters for the archiving process in the archive control table of the PERFDAT configuration database. This is done by applying different qualifiers.

The performance data archiving process is a background task that starts automatically if:

- the OpenVMS data collector is started
- the SNMP extension is started
- the EVA extension is started

In addition the archiving process can be manually started with the START ARCHIVE command. Its main tasks are:

- Archiving performance data to the archive node on a daily basis
- Data housekeeping - purging log files and deleting expired performance data files (files that are older then the keep time defined by the /KEEP_DAYS qualifier) on the local node

At startup the content of the archive control table is loaded into the volatile archive table. The parameters of the volatile archive table actually control the behavior of the archiving process. Thus, the parameters of the archive control table database are the initial startup parameters of the archiving process.

Qualifier

/DISABLE

The archiving process will be started in passive mode when the archiving process launches. In passive mode the archiving process does not archive any performance data file independently of the archive date/time and keep time defined by the /KEEP_DAYS qualifier.

This qualifier has no effect on the housekeeping functionality of the archiving process. Housekeeping will be done in passive mode too.

This qualifier is mutual exclusive to the /ENABLE qualifier.

/ENABLE

The archiving process will be started in active mode when the archiving process launches. Active mode means that the archiving process starts data archiving in accordance to the archive date/time and keep time defined by the /KEEP_DAYS qualifier.

This qualifier has no effect on the housekeeping functionality of the archiving process. Housekeeping will be performed in active and passive mode.

This qualifier is mutual exclusive to the /DISABLE qualifier.

/KEEP_DAYS=number of days

The value of this qualifier defines how long (how many days) performance data shall be kept in the PERFDAT$DB_ARCHIVE directory on the local node before the archiving process deletes these data files.

/TIME_OF_DAY=OpenVMS date/time

Defines the time of day the next archive run will be triggered. The format is standard OpenVMS date/time format. Since the archiving process is defined to be triggered daily any input but hours and minutes are ignored.

When the archiving process starts up, the next archive activation date/time is calculated based on these settings.

Example

```
PERFDAT_MGR>DEFINE ARCHIVE/ENABLE/TIME_OF_DAY=03:00/KEEP_DAYS=28
```

In this example the archive time is 03:00. Thus, the archiving process will be triggered every day at 03:00. Any performance data collection file in the directory PERFDAT$DB_ARCHIVE will be deleted unconditionally if its creation date is older then the time the archiving process is triggered minus 28 days.

After startup, the first archive date/time is calculated based on the current date/time and the time information stored in the archive control table of the PERFDAT configuration database (defined by this command). The next archive date/time is calculated to be closest to the current date/time and based on the archive time setting (in this example 03:00) stored in the archive control table.

- Assume the archiving process starts up at 02-OCT-2003 16:00.
  The first archiving date/time will be 03-OCT-2003 03:00
- Assume the archiving process starts up at 14-NOV-2003 01:00.
  The first archiving date/time will be 14-NOV-2003 03:00.

DELETE AUTOSTART

Deletes a node entry from the auto-start table of the PERFDAT configuration database.

Format

DELETE AUTOSTART *node_name*

Parameter

node_name
Specifies the name of the entry to be deleted from the auto-start table of the PERFDAT configuration database.

This parameter is mandatory.

Description

Deletes a node from the auto-start table of the PERFDAT configuration database.

For more information about the auto-start database please see the manual VSI OpenVMS PERFDAT – Architecture and Technical Description or the ADD AUTOSTART command description.

Qualifier

/OS_TYPE=*system*|*application-name*
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier is mandatory if you are deleting a non-OpenVMS node entry from the auto-start table of the PERFDAT configuration database.

If you delete an OpenVMS node you can omit the qualifier.

For more information about the /OS_TYPE qualifier see the ADD AUTOSTARTcommand description.

Examples

Command to delete the OpenVMS node entry VMSTM1 from the auto-start table:

```
PerfDat_MGR> DELETE AUTOSTART VMSTM1
PERFDAT_MGR-I-NODEDEL, node /VMSTM1/ deleted from autostart database
```

Command to delete the Brocade node entry TEST22 from the auto-start table:

```
PerfDat_MGR> DELETE AUTOSTART TEST22/OS_TYPE=BROCADE
PERFDAT_MGR-I-NODEDEL, node /TEST22/ deleted from autostart database
```

Command to delete the EVA system entry TEST22 from the auto-start table:

```
PerfDat_MGR> DELETE AUTOSTART TEST22/OS_TYPE=EVA
PERFDAT_MGR-I-NODEDEL, node /TEST22/ deleted from autostart database
```

DELETE PROFILE

Deletes a collection profile from the collection profile table of the PERFDAT configuration database.

Format

DELETE PROFILE *profile_name*

Parameter

profile_name
Specifies the collection profile name to be deleted from the collection profile table of the PERFDAT configuration database.

Description

Deletes a collection profile from the collection profile table of the PERFDAT configuration database.

For more information about the collection profile table please see the manual VSI OpenVMS PERFDAT – Architecture and Technical Description.

Qualifier

/OS_TYPE=*system*|*application-name*
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier is mandatory if you are deleting a collection profile valid for a non-OpenVMS system from the collection profile table of the PERFDAT configuration database.

If you delete an OpenVMS collection profile you can omit the qualifier.

For more information about the /OS_TYPE qualifier see the ADD PROFILEcommand description.

Examples

Command to delete the collection profile NEW_ONE valid for OpenVMS from the collection profile table:

```
PerfDat_MGR> DELETE PROFILE NEW_ONE
PERFDAT_MGR-I-CFGSUCC, Profile /NEW_ONE/ deleted for OS Type /OpenVMS/
```

Command to delete the collection profile NEW_ONE valid for TRU64 from the collection profile table:

```
PerfDat_MGR> DELETE PROFILE NEW_ONE/OS_TYPE=Tru64
PERFDAT_MGR-I-CFGSUCC, Profile /NEW_ONE/ deleted for OS Type /TRU64/
```

Command to delete the collection profile NEW_ONE valid for EVA systems from the collection profile table:

```
PerfDat_MGR> DELETE PROFILE NEW_ONE/OS_TYPE=EVA
PERFDAT_MGR-I-CFGSUCC, Profile /NEW_ONE/ deleted for OS Type /EVA/
```

DELETE REPORT

Deletes a report profile from the report profile table of the PERFDAT configuration database.

Format

DELETE REPORT *report_name*

Parameter

report_name
Specifies the report profile name to be deleted from the report profile table of the PERFDAT configuration database.

Description

Deletes a report profile from the report profile table of the PERFDAT configuration database.

For more information about the report profile table please see the manual VSI OpenVMS PERFDAT – Architecture and Technical Description.

Qualifier

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier is mandatory if you are deleting a report profile valid for a non-OpenVMS system from the report profile table of the PERFDAT configuration database.

If you delete an OpenVMS report profile you can omit the qualifier.

For more information about the /OS_TYPE qualifier see the ADD REPORTcommand description.

Examples

Command to delete the report profile NEW_ONE valid for OpenVMS from the report profile table:

```
PerfDat_MGR> DELETE REPORT NEW_ONE
PERFDAT_MGR-I-CFGSUCC, Profile /NEW_ONE/ deleted for OS Type /OpenVMS/
```

Command to delete the report profile NEW_ONE valid for TRU64 from the report profile table:

```
PerfDat_MGR> DELETE REPORT NEW_ONE/OS_TYPE=Tru64
PERFDAT_MGR-I-CFGSUCC, Profile /NEW_ONE/ deleted for OS Type /TRU64/
```

Command to delete the report profile NEW_ONE valid for EVA systems from the report profile table:

```
PerfDat_MGR> DELETE REPORT NEW_ONE/OS_TYPE=EVA
PERFDAT_MGR-I-CFGSUCC, Profile /NEW_ONE/ deleted for OS Type /EVA/
```

DISABLE ALERT

Disable online alerting for an active performance data collection that has online alerting enabled.

Format

DISABLE ALERT *collection-profile*

Parameter

collection-profile
Specifies the active performance data collection to disable online alerting. This parameter is mandatory.

Description

Disable online alerting. The *collection-profile* name parameter specifies the active performance data collection that has online alerting enabled. With the SHOW COLLECTION command you can check if online alerting is enabled for the performance data collection specified by the *collection-profile* parameter.

If you disable alerting for a SNMP or EVA performance data collection you have to apply the qualifiers listed below

- /NODE mandatory
- /OS_TYPE mandatory

If you disable alerting for an application performance data collection you have to apply the qualifiers listed below

- /OS_TYPE mandatory

If you want to disable alerting for an OpenVMS performance data collection both qualifiers can be omitted.

Qualifiers

/NODE=*node_name*

This qualifier is mandatory if you want to disable online alerting for an active SNMP or EVA performance data collection. It specifies the node name of the remote system to disable online alerting.

If you disable online alerting for an active OpenVMS or application performance collection this qualifier is ignored.

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application that runs a data collection started with the collection profile defined by the *collection-profile* parameter.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to disable online alerting for an active non-OpenVMS performance data collection the /OS_TYPE qualifier is mandatory.

If you disable online alerting for an active OpenVMS performance data collection you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Example

In this example online alerting will be disabled for the OpenVMS performance data collection started with the collection profile TESTSEL.

First we check if online alerting is actually enabled:

```
PerfDat_MGR> SHOW COLLECTION TESTSEL/OS_TYPE=OpenVMS

PROFILE: TESTSEL            OS Type: OPENVMS

Collection sample interval: 30 sec
SYSTEM metrix enabled: Yes
CPU metrix enabled: Yes
PROCESS metrix enabled: Yes
     On Process: ALL
     USER metrix enabled: Yes
          On USER: ALL
     IMAGE metrix enabled: Yes
          On IMAGE: ALL
     ACCOUNT metrix enabled: Yes
          On ACCOUNT: All
VOLUME metrix enabled (based on XFC stats): Yes
```

```
        On Volumes: ALL
        IO size stats enabled: No
        FILE metrix based on XFC stats enabled: Yes
            Selective File Filtering: ENABLED
            File Filter List Ptr: $1$DKB100:<PERFDAT.REL30.SRC.V732>SEARCHFILE.DAT
            Open File Collection only: Yes
            IO size stats enabled: No
DEVICE metrix enabled: Yes
        On DEVICES: *$D*, *DSA*
        IO size  metrix on selected FOD devices enabled: No
        FILE metrix on selected FOD devices enabled: Yes
            Selective File Filtering: ENABLED
            File Filter List Ptr: $1$DKB100:<PERFDAT.REL30.SRC.V732>SEARCHFILE.DAT
        Per PROCESS collection on selected devices enabled: Yes
            On Process: ALL
            Per FILE collection enabled for each Process collection on FOD devices: Yes
            Selective File Filtering: DISABLED
        Processes excluded from file monitoring: NONE
Device capacity and path info metrix enabled: Yes
LAN metrix enabled: Yes
        LAN Device metrix enabled: Yes
        LAN PROTOCOL metrix enabled: Yes
SCS metrix enabled: Yes

        Collection started at  5-AUG-2005 00:00:30.00

        Online alerting enabled: Yes
        Alert definition file in use: PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG

        Actual Sample Count: 1750
        Actual Record Count in Storage Area: 419.349 [*1000]

        Data written until now: 139.601 MB
        Actual data rate:       67.748 kB

        Collection data can be accessed online: No
```

Online alerting is enabled using the alert definition file PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG. Now we disable online alerting for the OpenVMS performance data collection TESTSEL with

```
PerfDat_MGR> DISABLE ALERT TESTSEL
PERFDAT_MGR-I-ALERTSUC, disabled alerting for collection /TESTSEL/
```

or

```
PerfDat_MGR> DISABLE ALERT TESTSEL/OS_TYPE=OpenVMS
PERFDAT_MGR-I-ALERTSUC, disabled alerting for collection /TESTSEL/
```

ENABLE ALERT

Enable online alerting for an active performance data collection that has online alerting disabled.

Format

ENABLE ALERT *collection-profile*

Parameter

collection-profile
Specifies the active performance data collection to enable online alerting. This parameter is mandatory.

Description

Enable online alerting. The *collection-profile* name parameter specifies the active performance data collection that has online alerting disabled. With the SHOW COLLECTION command you can check if online alerting is disabled for the active performance data collection specified by the *collection-profile* parameter.

If you enable alerting for a SNMP or EVA performance data collection you have to apply the qualifiers listed below

- /NODE mandatory
- /OS_TYPE mandatory

If you enable alerting for an application performance data collection you have to apply the qualifiers listed below

- /OS_TYPE mandatory

If you want to enable alerting for an OpenVMS performance data collection both qualifiers can be omitted.

Prerequisite for enabling online alerting for an active performance data collection is that a valid alert definition file exists.

The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc.

If you omit the /FILENAME qualifier default alert definition files are used depending on the value of the /OS_TYPE qualifier as listed below:

OS_TYPE	Default alert definition file
OpenVMS	PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG
Tru64	PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG
Brocade	PERFDAT$CFG:PERFDAT_ALERT_BROCADE.CFG
EVA	PERFDAT$CFG:PERFDAT_ALERT_EVA.CFG

No predefined alert definition files exist for Solaris and Linux systems and any application. Thus, if you want to enable online alerting for an active Solaris, Linux or application performance data collection and you omit the /ALERT_FILENAME qualifier the command fails.

The available alert methods to inform system management about alert conditions are:

- OPCOM messages (always)
- User definable command scripts to be executed on warning or critical conditions. These command scripts are defined within the alert blocks of the alert definition file.

Warning alert OPCOM messages always start with PERFDAT-W-ALERT. Critical alert OPCOM messages always start with PERFDAT-E-ALERT. Alert clear messages that signal to system management that a particular alert condition is no longer true starts with PERFDT-I-ALERT.

Each OPCOM message contains full information about the alert condition:

- Severity (PERFDAT-E-ALERT, PERFDAT-W-ALERT, PERFDT-I-ALERT)
- Node name the alert was triggered
- Element that exceeds the warning/critical threshold
- Statistics that exceeds the warning/critical threshold
- Metric the statistics belongs to
- Value of the statistics
- Threshold value
- Comparison operator (LT. EQ, GT)

Independently of the alert method defined in the alert definition file all alerts are written to a log file. The log file is stored in the directory PERFDAT$ALERT and its format is:

PERFDAT_ALERT_*node*.LOG_*date*

where:

- *node* local node name
- *date* date the log file was created

On day change a new log file is created. Thus, one log file contains all alerts triggered on one day by all performance data collections running on the local node that have online alerting enabled.

Alert definition file description

The alert definition file is a text file that consists of alert blocks. An alerting block defines the alerting rules for specific statistics of a metric. There exists no restriction that a specific metric can be defined in one alert block within an alert definition file only. You can define several alert blocks that refer to the same metric. To add a comment in the alert definition file place an exclamation mark (!) at the first position of the line that contains the comment.

Each alert definition block starts with the
ADD ALERT:
clause and ends with the
END ALERT:
clause.

Each alert block contains several parameter keywords as described below:

- *OSTYPE:*
  Defines the OS type the alert block applies to. Supported OS types are:
  - OpenVMS
  - Tru64
  - Brocade
  - EVA
  - Solaris
  - Linux
  - Name of any application that has the VSI PERFDAT API implemented. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database

- *METRIX:*
  Defines the metric to be monitored. There is no restriction that a specific metric can be defined only once in an alert block. This means that you can define several alert blocks that refer to the same metric.

- *STATISTICS:*
  Defines all the statistics of the metric defined by the METRIX parameter keyword to be monitored. Use the comma (,) or the OR sign (|) as list separators for the statistics list. You can enter any valid statistics directly collected by the data collector as well as any user defined (calculated) statistics stored in the stored procedure table of the PERFDAT configuration database valid for the metric defined by the METRIX keyword (for more information about user defined statistics please refer to the manuals VSI OpenVMS PERFDAT– Architecture and Technical Description and VSI OpenVMS PERFDAT – DQL$ Reference Manual). Up to 16 statistics can be defined in the list.

- *SCALED_BY:*
  Enter the scaling value for the statistics defined with the STATISTICS parameter keyword enter the scaling values as comma (,) or OR sign (|) separated list. The first entry in the SCALED_BY list applies to the

first entry in the STATISTICS list, the second entry in the SCALED_BY list applies to the second entry in the STATISTICS list and so on.
You can enter either a number (e.g. 100) or a valid statistic. E.g. you want to scale OpenVMS system CPU load to the range [0 … 100 %], enter iCpuLoad (overall CPU load) within the STATISTICS list and iCpuCnt (number of CPU's) at the appropriate position in the SCALED_BY list.
If no statistics shall be scaled you can omit this parameter keyword.
If some of the statistics of the STATISTICS list shall not be scaled enter either '1' or nothing at the appropriate position.

E.g.:

STATISTICS: iCpuLoad, iMpSync, iInter

If you want to scale iCpuLoad and iInter to the range [0 .. 100%] but not iMpSync enter either

SCALED_BY: iCpuCnt, 1, iCpuCnt

or

SCALED_BY: iCpuCnt,, iCpuCnt

- *OPERATOR:*
  Enter the operators to use for comparing the values the statistics defined with the STATISTCS parameter keyword and the appropriate threshold values. Enter the operators as comma (,) or OR sign (|) separated list. The first entry in the OPERATOR list applies to the first entry in the STATISTICS list, the second entry in the OPERATOR list applies to the second entry in the STATISTICS list and so on.

  Valid Operators:
  - GT alert condition: stats value > threshold
  - LT alert condition: stats value < threshold
  - EQ alert condition: stats value = threshold

  If you omit this parameter keyword GT is assumed for all statistics defined in the alert block.
  If you enter this parameter keyword in an alert block you have to enter a valid operator for each element in the STATISTICS list. Otherwise reading the alert block fails.

- *ELEMENTS:*
  With the ELEMENTS parameter keyword you define all the elements the rules defined within the alert block shall be applied to. Use the comma (,) or the OR sign (|) as list separators for the element list. The element list can contain elements the rules defined by the alert block apply to as well as elements that should be excluded from online performance alerting. Elements that should be excluded from online performance alerting have to be preceded with the '!=' or '<>' tag in the comma separated list of the ELEMENT parameter. Full wildcard search is supported for each element in the list

  E.g.:

  ELEMENTS: *PERF%AT*, SWAP*, <>*SRV*, TCPIP$

If you enter the element list as shown above the alert block is valid for all elements that match either the wildcard search criterion
  - *PERF%AT*
  - SWAP*

but not (exclude tag <> is present)
  - *SRV*

or
  - matches TCPIP$ literally.

- *STACKED:*
  YES | NO -> If you enter YES the statistic values of all elements that matches the criteria defined by the ELEMENTS parameter keyword are summarized (stacked calculation) per statistic of the STATISTICS list and the alert rule defined by the alert block are applied to the stacked values of the defined statistics.

- *STACKED_ELEM_NAME:*
  If you have entered YES to the STACKED parameter keyword it is mandatory to define this parameter keyword. If you have enabled stacked calculation defined and the stacked values of one or more statistics defined by the alert block exceeds the thresholds defined, the element name entered here is used to signal the alert. The STACKED_ELEM_NAME string length is limited to 16 characters.

- *SAMPLE_COUNT:*
  The SAMPLE_COUNT parameter keyword defines the number of performance data collection samples to average the values of the statistics for each element defined by the alert block. At the end of the time range defined by this parameter keyword it is checked if one of the averaged values of the defined statistics has exceeded the thresholds defined in the alert block. If this is the case the user is alerted according to the alert definitions (WARNING_CMD / CRITICAL_CMD). Thus, the alert monitoring time interval can be greater but not less then the collection sample interval. This is sometimes helpful to suppress alerting on resource load spikes. In order to avoid getting an alert for each resource load spike the SAMPLE_COUNT has to be at minimum 2.

- *WARNING_THRESHOLDS:*
  Enter the warning level thresholds for the statistics defined by the STATISTCS parameter keyword. Enter these threshold values as comma (,) or OR sign (|) separated list.
  The first entry in the WARNING_THRESHOLDS list applies to the first entry in the STATISTICS list, the second entry in the WARNING_THRESHOLDS list applies to the second entry in the STATISTICS list and so on.
  Whenever the average value of the statistics of an element defined within this alert block exceeds the value defined herein, a warning OPCOM message is sent and the command file defined by the WARNING_CMD parameter keyword is triggered. If you want to

disable warning processing for a statistic of the STATISTICS list enter -1 at the appropriate position in this list.

- *WARNING_CMD:*
  Command file to be triggered in case of a warning condition. Seven parameters are applied to this user definable command file.
  - P1 Node name
  - P2 Metric name
  - P3 Statistic
  - P4 Element name
  - P5 Average value of the statistics
  - P6 threshold value defined by WARNING_THRESHOLD
  - P7 2 (= warning severity code)

- *CRITICAL_THRESHOLDS:*
  Enter the critical level thresholds for the statistics defined by the STATISTCS parameter keyword. Enter these threshold values as comma (,) or OR sign (|) separated list.
  The first entry in the CRITICAL_THRESHOLDS list applies to the first entry in the STATISTICS list, the second entry in the CRITICAL_THRESHOLDS list applies to the second entry in the STATISTICS list and so on.
  Whenever the average value of the statistics of an element defined within this alert block exceeds the value defined herein, a critical OPCOM message is sent and the command file defined by the CRITICAL_CMD parameter keyword is triggered.
  If you want to disable warning processing for a statistic of the STATISTICS list enter -1 at the appropriate position in this list.

- *CRITICAL_CMD:*
  Command file to be triggered in case of a critical condition. Seven parameters are applied to this user definable command file.
  - P1 Node name
  - P2 Metric name
  - P3 Statistic
  - P4 Element name
  - P5 Average value of the statistics
  - P6 threshold value defined by CRITICAL_THRESHOLD
  - P7 3 (= error severity code)

- *CLEARALERTMSG:*
  The online performance alerting sub-system can be configured to send an alert clear notification whenever an alert condition for a particular statistics is no longer true and the value of the statistics is again within the excepted range. If the CLEARALERTMSG parameter is not defined or if the value FALSE is assigned to this parameter the alert clear message feature is disabled. If the value TRUE is assigned to the CLEARALERTMSG parameter the alert clear message feature is enabled.

- *CLEARALERT_CMD:*

Command file to be triggered if a previous alert condition is no longer true and the alert clear message feature is enabled by the *CLEARALERTMSG* parameter. Seven parameters are applied to this user definable command file.

- P1 Node name
- P2 Metric name
- P3 Statistic
- P4 Element name
- P5 Average value of the statistics
- P6 Critical an warning threshold – format: "*error*/*warning*"
- P7 1 (= clear severity code)

- *SUBMIT_QUEUE:*
  This parameter keyword defines the batch queue to submit the alert content files. Each alert block is processed independently by the online alerting subsystem. When the alerting subsystem processes an alert block it opens a warning alert content and a critical content file for the alert block in the directory PERFDAT$ALERT (file name ERRMSG_*metric_nn_node*.COM where *metric* is the metric defined by the METRIX parameter keyword of the alert block, *nn* is a number and *node* is the local node name). Depending if a warning, a critical or a clear alert condition or is triggered a new record is stored in either of this files containing the appropriate user defined action script and the parameters (P1 – P7) as defined above. When the alert processing has completed for the alert block these alert content files are closed. If these alert content files contain valid records they are queued to the batch queue defined herein. If these files contain no records the files are unconditionally deleted, otherwise they are deleted after the batch job has completed successfully. If the batch job fails these files are kept. Thus one can check if all alerts have been submitted by checking the directory PERFDAT$ALERT for files named ERRMSG*.COM.
  The submit queue must exist, and have to be in idle or busy state. Otherwise the alert block will be marked as invalid when enabling online alerting and the alert processing defined by this alert block will not be performed.

Qualifier

/ALERT_FILENAME = file-name

The /ALERT_FILENAME qualifier specifies the alert definition file to use. The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc.

If you omit the qualifier default alert definition files are used depending on the value of the /OS_TYPE qualifier as listed below:

OS_TYPE	Default alert definition file
OpenVMS	PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG
Tru64	PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG

Brocade	PERFDAT$CFG:PERFDAT_ALERT_BROCADE.CFG
EVA	PERFDAT$CFG:PERFDAT_ALERT_EVA.CFG

No predefined alert definition files exist for Solaris and Linux systems and any application. Thus, if you want to enable online alerting for an active Solaris, Linux or application performance data collection and you omit the /ALERT_FILENAME qualifier the command fails.

/NODE=*node_name*

This qualifier is mandatory if you want to enable online alerting for an active SNMP or EVA performance data collection. It specifies the node name of the remote system to enable online alerting.

If you enable online alerting for an active OpenVMS or application performance collection this qualifier is ignored.

/OS_TYPE=*system*|*application-name*
/OS_TYPE=OPENVMS (default)

The /OS_TYPE qualifier defines the system or application that runs a data collection started with the collection profile defined by the *collection-profile* parameter.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to enable online alerting for an active non-OpenVMS performance data collection the /OS_TYPE qualifier is mandatory.

If you enable online alerting for an active OpenVMS performance data collection you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Examples

Command example

In this example online alerting will be enabled for the OpenVMS performance data collection started with the collection profile TESTSEL.

First we check if online alerting is actual disabled:

```
PerfDat_MGR>SHOW COLLECTION TESTSEL/OS_TYPE=OpenVMS

PROFILE: TESTSEL            OS Type: OPENVMS

Collection sample interval: 30 sec
SYSTEM metrix enabled: Yes
CPU metrix enabled: Yes
PROCESS metrix enabled: Yes
      On Process: ALL
      USER metrix enabled: Yes
            On USER: ALL
      IMAGE metrix enabled: Yes
            On IMAGE: ALL
      ACCOUNT metrix enabled: Yes
            On ACCOUNT: All
VOLUME metrix enabled (based on XFC stats): Yes
      On Volumes: ALL
      IO size stats enabled: No
      FILE metrix based on XFC stats enabled: Yes
            Selective File Filtering: ENABLED
            File Filter List Ptr: $1$DKB100:<PERFDAT.REL30.SRC.V732>SEARCHFILE.DAT
            Open File Collection only: Yes
            IO size stats enabled: No
DEVICE metrix enabled: Yes
      On DEVICES: *$D*, *DSA*
      IO size  metrix on selected FOD devices enabled: No
      FILE metrix on selected FOD devices enabled: Yes
            Selective File Filtering: ENABLED
            File Filter List Ptr: $1$DKB100:<PERFDAT.REL30.SRC.V732>SEARCHFILE.DAT
      Per PROCESS collection on selected devices enabled: Yes
            On Process: ALL
            Per FILE collection enabled for each Process collection on FOD devices: Yes
            Selective File Filtering: DISABLED
      Processes excluded from file monitoring: NONE
Device capacity and path info metrix enabled: Yes
LAN metrix enabled: Yes
      LAN Device metrix enabled: Yes
      LAN PROTOCOL metrix enabled: Yes
SCS metrix enabled: Yes

      Collection started at  5-AUG-2005 00:00:30.00

      Online alerting enabled: No

      Actual Sample Count: 1750
      Actual Record Count in Storage Area: 419.349 [*1000]

      Data written until now: 139.601 MB
      Actual data rate:      67.748 kB

      Collection data can be accessed online: No
```

Alerting is disabled for the OpenVMS performance data collection started with the collection profile TESTSEL. Now we enable online alerting for this performance data collection using the alert definition file PERFDAT$CFG:ALERT.CFG with:

```
PerfDat_MGR> ENABLE ALERTTESTSEL /ALERT_FILENAME=PERFDAT$CFG:ALERT.CFG
PERFDAT_MGR-I-ALERTSUC, enabled alerting for collection /TESTSEL/
```

or

```
PerfDat_MGR> ENABLE ALERT TESTSEL/OS_TYPE=OpenVMS
                              /ALERT_FILENAME=PERFDAT$CFG:ALERT.CFG
PERFDAT_MGR-I-ALERTSUC, enabled alerting for collection /TESTSEL/
```

Alert block configuration examples

Example 1

The first example shows how to configure an alert block valid for the SYSTEM metric of OpenVMS performance data collections. The statistics iCpuLoad, iInter and iMPSync are defined to be monitored. These statistics are scaled to the range of [0 … 100%] since iCpuCnt (number of CPU's) is used as the scaling parameter for all of them. The alert monitoring time interval is set to 2. Thus, if you enable online alerting for an OpenVMS data collection with a sample interval of 2 min using an alert definition file containing this alert block, the time interval to check the alert conditions is 4 min, and the 4-minute average values of all statistics defined within the alert block are compared with the thresholds.

A warning alert is triggered if:

- iInter > 20 % of CPU resources are used
- iMPSync > 20 % of CPU resources are used

No warning processing is performed for the statistic iCpuLoad.

A critical alert is triggered if:

- iCpuLoad > 90 % of CPU resources are used
- iInter > 60 % of CPU resources are used
- iMPSync > 60 % of CPU resources are used

The alert mechanism is sending OPCOM messages only since no warning and critical alert command scripts are defined and therefore no submit queue is defined.

```
ADD ALERT:

        OSTYPE: OpenVMS
        METRIX: SYSTEM
        STATISTICS: iCpuLoad, iIntr, iMPSync
        SCALED_BY: iCpuCnt, iCpuCnt, iCpuCnt
        OPERATOR: GT, GT, GT, GT
        ELEMENTS: *
        STACKED: NO
        STACKED_ELEM_NAME:
```

```
        SAMPLE_COUNT: 2
        WARNING_THRESHOLD: -1, 20, 20
        WARNING_CMD:
        CRITICAL_THRESHOLD: 90, 60, 60
        CRITICAL_CMD:
        SUBMIT_QUEUE:

END ALERT:
```

Example 2

This example shows how to configure an alert block valid for the PROCESS metric of OpenVMS performance data collections. The collected statistics iCpuLoad, iKernel and the calculated statistics $iCpuNorm are defined to be monitored. These statistics are not scaled. The alert monitoring time interval is set to 10. Thus, if you enable online alerting for an OpenVMS data collection with a sample interval of 2 min using an alert definition file containing this alert block, the time interval to check the alert conditions is 20 min, and the 20 minute average values of all statistics defined are compared with the thresholds. Since NO has been applied to the STACKED parameter keyword the alert processing is performed for each process (per process alerting)

A warning alert is triggered if:

- iCpuLoad > 40 % of CPU resources are used
- iKernel > 30 % of CPU resources are used
- $iCpuNorm > 20 % of CPU resources are used

No warning processing is performed for the statistic iCpuLoad.

A critical alert is triggered if:

- iCpuLoad > 80 % of CPU resources are used
- iKernel > 40 % of CPU resources are used
- $iCpuNorm > 40 % of CPU resources are used

For the warning processing the alert mechanism is sending OPCOM messages only since no warning alert command script has been defined.

For the critical processing the alert mechanism is sending OPCOM messages and triggering the critical alert command script

- DSA1:[ALERT]PROCESS_ALERT.COM

In the case of critical alert conditions this command script is executed once per element and statistic that fulfills the critical alert condition in a batch job submitted to the queue VMSTM1$QUEUE.

```
ADD ALERT:

        OSTYPE: OpenVMS
        METRIX: PROCESS
        STATISTICS: iCpuLoad, iKernel, $iCpuNorm
        SCALED_BY: 1, 1, 1
        OPERATOR: GT, GT, GT
        ELEMENTS: *
        STACKED: NO
        STACKED_ELEM_NAME:
```

```
        SAMPLE_COUNT: 10
        WARNING_THRESHOLD: 40, 30, 20
        WARNING_CMD:
        CRITICAL_THRESHOLD: 80, 40, 40
        CRITICAL_CMD: DSA1:[ALERT]PROCESS_ALERT.COM
        SUBMIT_QUEUE: VMSTM1$BATCH

END ALERT:
```

Example 3

The last example shows how to configure an alert block valid for the PROCESS metric of OpenVMS performance data collections in stacked mode. The settings in the alert block are almost identical to the example before. Since YES has been assigned to the STACKED parameter keyword the alert conditions are not checked for each element that matches the ELEMENTS filter list, but the statistic values of all elements that matches the criteria defined by the ELEMENTS parameter keyword are summarized and the alert rule defined by the alert block are applied to the stacked values of the defined statistics.

If an alert condition is triggered the string ‘SPW processes’ is used as the element name to signal the alert.

```
ADD ALERT:

        OSTYPE: OpenVMS
        METRIX: PROCESS
        STATISTICS: iCpuLoad, iKernel
        SCALED_BY: 1, 1
        OPERATOR: GT, GT
        ELEMENTS: SPW*
        STACKED: YES
        STACKED_ELEM_NAME: SPW process
        SAMPLE_COUNT: 10
        WARNING_THRESHOLD: 40, 30
        WARNING_CMD:
        CRITICAL_THRESHOLD: 80, 40
        CRITICAL_CMD: DSA1:[ALERT]PROCESS_ALERT.COM
        SUBMIT_QUEUE: VMSTM1$BATCH

END ALERT:
```

EXIT

This command terminates the PERFDAT_MGR session.

Format

EXIT

Description

This command terminates the PERFDAT_MGR session.

EXPORT PROFILE

Exports an existing collection profile from the collection profile table of the PERFDAT configuration database to a transport file.

Format

EXPORT PROFILE *profile_name*

Parameter

profile_name
Specifies an existing collection profile stored in the collection profile table of the PERFDAT configuration database to export to the transport file.

Description

Exports an existing collection profile defined by the *profile_name* parameter from the collection profile table of the PERFDAT configuration database to a transport file.

The transport file has to be defined by the /FILENAME qualifier.

If you omit the optional qualifier /OS_TYPE the export routine searches by default for a matching OpenVMS collection profile in the collection profile table of the PERFDAT configuration database.

If you want to export a collection profile valid for a non-OpenVMS system the /OS_TYPE qualifier is mandatory.

Qualifier

/FILENAME=file_name

Specifies the file name of the transport file. If the file does not exist the file is created automatically. This qualifier is mandatory

Note

Do not try to create the transport file manually. Special file attributes are required for the transport file.

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the exported collection profile is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to export an existing collection profile for a non-OpenVMS system to a transport file the /OS_TYPE qualifier is mandatory.

If you export an existing OpenVMS collection profile you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Example

In this example the OpenVMS collection profile TESTSEL, the Tru64 collection profile DEFAULT and the EVA collection profile DEFAULT are exported to the transport file SYS$LOGIN:TRANSPORT.DAT. Since the transport does not exist the file is created with the first export command:

```
PerfDat_MGR> EXPORT PROFILE TESTSEL/FILE=SYS$LOGIN:TRANSPORT.DAT

Export File SYS$LOGIN:TRANSPORT.DAT does not exist. Do you want to create it [Yes]: ↵
PERFDAT_MGR-I-EXPSUCC, Profile /TESTSEL/ for OS Type /OPENVMS/ exported

PerfDat_MGR> EXPORT PROFILE DEFAULT
                          /FILE=SYS$LOGIN:TRANSPORT.DAT/OS_TYPE=Tru64
PERFDAT_MGR-I-EXPSUCC, Profile /DEFAULT/ for OS Type /TRU64/ exported

PerfDat_MGR> EXPORT PROFILE DEFAULT
                          /FILE=SYS$LOGIN:TRANSPORT.DAT/OS_TYPE=EVA
PERFDAT_MGR-I-EXPSUCC, Profile /DEFAULT/ for OS Type /EVA/ exported
```

EXPORT REPORT

Exports an existing report profile from the report profile table of the PERFDAT configuration database to a transport file.

Format

EXPORT REPORT*report_name*

Parameter

report_name
Specifies an existing report profile stored in the report profile table of the PERFDAT configuration database to export to the transport file.

Description

Exports an existing report profile defined by the *report_name* parameter from the report profile table of the PERFDAT configuration database to a transport file.

The transport file has to be defined by the /FILENAME qualifier.

If you omit the optional qualifier /OS_TYPE the export routine searches by default for a matching OpenVMS report profile in the report profile table of the PERFDAT configuration database.

If you want to export a report profile valid for a non-OpenVMS systems the /OS_TYPE qualifier is mandatory.

Qualifier

/FILENAME=file_name

Specifies the file name of the transport file. If the file does not exist the file is created automatically. This qualifier is mandatory

Note

Do not try to create the transport file manually. Special file attributes are required for the transport file.

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the exported report profile is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to export an existing report profile for a non-OpenVMS system to a transport file the /OS_TYPE qualifier is mandatory.

If you export an existing OpenVMS report profile you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS
- Tru64
- Brocade
- RDB
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Example

In this example the OpenVMS report profile WEEK, the Tru64 report profile WEEK and the EVA report profile WEEK are exported to the transport file SYS$LOGIN:TRANSPORT.DAT. Since the transport does not exist the file is created with the first export command:

```
PerfDat_MGR> EXPORT REPORT WEEK/FILE=SYS$LOGIN:TRANSPORT.DAT

Export File SYS$LOGIN:TRANSPORT.DAT does not exist. Do you want to create it [Yes]:↵
PERFDAT_MGR-I-EXPSUCC, Report profile /WEEK/ for OS Type /OPENVMS/ exported

PerfDat_MGR> EXPORT REPORT WEEK
                        /FILE=SYS$LOGIN:TRANSPORT.DAT/OS_TYPE=tru64
PERFDAT_MGR-I-EXPSUCC, Report profile /WEEK/ for OS Type /TRU64/ exported

PerfDat_MGR> EXPORT REPORT WEEK
                        /FILE=SYS$LOGIN:TRANSPORT.DAT/OS_TYPE=EVA
PERFDAT_MGR-I-EXPSUCC, Report profile /WEEK/ for OS Type /EVA/ exported
```

FLUSHNAME_SERVER

This command flushes the whole performance database name server cache on the local node and triggers the DQL_NAME service to rebuild it.

Format

FLUSHNAME_SERVER

Parameter

None

Description

This command flushes the whole performance database name server cache on the local node and triggers the DQL_NAME service to rebuild it.

For more information about the performance database file name cache and the DQL_NAME service please refer to the START NAME_SERVER command description or to the manual VSI PERFDAT– Architecture and Technical Description.

Example

```
PerfDat_MGR> FLUSH NAME_SERVER

PERFDAT_MGR-I-STARTFLUSH, start flushing invalid DQL name server cache entries
PERFDAT_MGR-I-CACHEFLUSH, DQL name server cache entries flushed
```

HELP

This command invokes the online help.

Format

HELP

Description

This command invokes the online help.

IMPORT PROFILE

Imports a collection profile from a transport file to the collection profile table of the PERFDAT configuration database.

Format

IMPORT PROFILE *profile_name*

Parameter

profile_name
Specifies the name of the collection profile to be imported from a transport file.

Description

Imports a collection profile defined by the *profile_name* parameter from a transport file to the collection profile table of the PERFDAT configuration database.

The transport file has to be defined by the /FILENAME qualifier.

If you omit the optional qualifier /OS_TYPE the import routine searches by default for a matching OpenVMS collection profile in the transport file.

If you want to import a collection profile valid for a non-OpenVMS system the /OS_TYPE qualifier is mandatory.

Qualifier

/FILENAME=file_name

File name of the transport file. This qualifier is mandatory.

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the imported collection profile is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to import a collection profile for a non-OpenVMS system to a transport file the /OS_TYPE qualifier is mandatory.

If you import an OpenVMS collection profile you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

/UPDATE

If the collection profile already exists in the collection profile table of the PERFDAT configuration database you can apply the /UPDATE qualifier to update that collection profile. Otherwise the import of the collection profile fails.

Example

In this example the OpenVMS collection profile TESTSEL and the EVA collection profile DEFAULT are imported from the transport file SYS$LOGIN:TRANSPORT.DAT. Both collection profiles already exist in the collection profile table of the PERFDAT configuration database. Thus, we have to apply the /UPDATE qualifier.

```
PerfDat_MGR> IMPORT PROFILE TESTSEL /FILE=SYS$LOGIN:TRANSPORT.DAT/UPDATE
PERFDAT_MGR-I-IMPSUCC, Profile /TESTSEL/ for OS Type /OPENVMS/ imported

PerfDat_MGR> IMPORT PROFILE DEFAULT
                    /FILE=SYS$LOGIN:TRANSPORT.DAT/OS_TYPE=EVA/UPDATE
PERFDAT_MGR-I-IMPSUCC, Profile /DEFAULT/ for OS Type /EVA/ imported
```

IMPORT REPORT

Imports a report profile from a transport file to the report profile table of the PERFDAT configuration database.

Format

IMPORT REPORT *report_name*

Parameter

report_name
Specifies the name of the report profile to be imported from a transport file.

Description

Imports a report profile defined by the *profile_name* parameter from a transport file to the report profile table of the PERFDAT configuration database.

The transport file has to be defined by the /FILENAME qualifier.

If you omit the optional qualifier /OS_TYPE the import routine searches by default for a matching OpenVMS report profile in the transport file.

If you want to import a report profile valid for a non-OpenVMS system the /OS_TYPE qualifier is mandatory.

Qualifier

/FILENAME=file_name

File name of the transport file. This qualifier is mandatory.

/OS_TYPE= system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the imported report profile is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to import a report profile for a non-OpenVMS system to a transport file the /OS_TYPE qualifier is mandatory.

If you import an OpenVMS report profile you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS
- Tru64
- Brocade
- RDB
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

/SOURCE=source_collection_profile

When you update an existing report profile, this qualifier can be applied to select a different source collection profile as that stored in the report header section of the existing report profile. This qualifier is ignored if the /UPDATE qualifier is not applied.

/UPDATE

If the report profile already exists in the report profile table of the PERFDAT configuration database you can apply the /UPDATE qualifier to update that collection profile. Otherwise the import of the collection profile fails.

When a report profile is updated not the whole content of the report profile is updated but only the statistics in the report layout and statistics section of the report profile and, in case the /SOURCE qualifier is applied, the default source collection profile in the report header section. The report header section is left unchanged.

For more information about the layout of a report profile please see the ADD REPORT command description.

Example

In this example the OpenVMS and the EVA report profiles WEEK are imported from the transport file SYS$LOGIN:TRANSPORT.DAT. Both report profiles already exist in the report profile table of the PERFDAT configuration database. Thus, we have to apply the /UPDATE qualifier.

```
PerfDat_MGR> IMPORT REPORT WEEK /FILE=SYS$LOGIN:TRANSPORT.DAT/UPDATE
PERFDAT_MGR-I-IMPSUCC, Report /WEEK/ for OS Type /OPENVMS/ imported
```

```
PerfDat_MGR>IMPORT REPORT WEEK
                    /FILE=SYS$LOGIN:TRANSPORT.DAT/OS_TYPE=EVA/UPDATE
PERFDAT_MGR-I-IMPSUCC, Report /WEEK/ for OS Type /EVA/ imported
```

LAUNCH

Launches the whole PERFDAT environment or parts of it depending on the *launch_keyword* parameter.

Format

LAUNCH *launch_keyword*

Parameter

launch_keyword
Specifies the parts of the PERFDAT environment to be started. Supported keywords are:

- DQL$SRV
- PDBC$SRV
- ALL
- PERFDAT
- PERFDAT_EVA
- PERFDAT_SNMP

Description

Launches the whole PERFDAT environment or parts of it depending on the *launch_keyword* parameter.

LAUNCH ALL
Launches the whole PERFDAT environment by executing the command scripts

- SYS$STARTUP:PERFDAT$STARTUP.COM
- SYS$STARTUP:PERFDAT_EVA$STARTUP.COM
- SYS$STARTUP:PERFDAT_SNMP$STARTUP.COM

The following tasks are performed:

- Auto-archiving process startup
- Performance database filename cache service
- DQL$SRV service definition and initialization
- PDBC$SRV service definition and initialization
- EVA master &working process startup
- SNMP master & agent process startup
- OpenVMS data collector startup
- Performance data collection startup for all nodes (OpenVMS & non-OpenVMS nodes) defined in the auto-start table of the PERFDAT configuration database the local node is valid for.

Prerequisite: You have to be logged in as a privileged user.
The privileges required are:

- CMKRNL
- NETMBX

- OPER
- SYSLCK
- SYSPRV
- TMPMBX
- WORLD

LAUNCH DQL$SRV

Defines, initializes and starts the DQL$SRV service by executing the command script SYS$STARTUP:DQLSRV$STARTUP.COM.

LAUNCH PDBC$SRV

Defines, initializes and starts the PDBC$SRV service by executing the command script SYS$STARTUP:PDBC$STARTUP.COM.

LAUNCH PERFDAT

Starts all components of the PERFDAT environment but the SNMP extension by executing the command script

- SYS$STARTUP:PERFDAT$STARTUP.COM

The following actions are performed:

- Auto archiving process startup
- Performance database filename cache service
- DQL$SRV service definition and initialization
- PDBC$SRV service definition and initialization
- OpenVMS data collector startup
- If the local node refers to an existing entry in the auto-start table of the PERFDAT configuration database a performance collection as defined in that entry is started automatically.

Prerequisite: You have to be logged in as a privileged user.
The privileges required are:

- CMKRNL
- NETMBX
- OPER
- SYSLCK
- SYSPRV
- TMPMBX
- WORLD

LAUNCH PERFDAT_EVA

Starts the PERFDATEVA extension by executing the command script

- SYS$STARTUP:PERFDAT_EVA$STARTUP.COM.

The following tasks are performed:

- Auto archiving process startup
- Performance database filename cache service
- DQL$SRV service definition and initialization
- PDBC$SRV service definition and initialization
- EVA master &process process startup
- For all EVA systems entered in the auto-start table of the VSI PERFDAT configuration database a performance collection is automatically started as defined in that auto-start entries if the local node is defined to be the EVA agent (=node that runs the EVA data collection)

Prerequisite: You have to be logged in as a privileged user.
The privileges required are:

- CMKRNL
- NETMBX
- OPER
- SYSLCK
- SYSPRV
- TMPMBX
- WORLD

LAUNCH PERFDAT_SNMP

Starts the PERFDAT SNMP extension by executing the command script

- SYS$STARTUP:PERFDAT_SNMP$STARTUP.COM.

The following tasks are performed:

- Auto archiving process startup
- Performance database filename cache service
- DQL$SRV service definition and initialization
- PDBC$SRV service definition and initialization
- SNMP master & agent process startup
- For all nodes entered in the auto-start table of the VSI PERFDAT configuration database that are defined to be monitored via SNMP a performance collection is automatically started as defined in that auto-start entries if the local node is defined to be the SNMP agent (=node that runs a remote collection via SNMP)

Prerequisite: You have to be logged in as a privileged user.
The privileges required are:

- CMKRNL
- NETMBX
- OPER
- SYSLCK
- SYSPRV

- TMPMBX
- WORLD

Qualifier

/BATCH

Obsolete

Prior to VSI PERFDAT V3.2 ECO1 all privileged users but the user SYSTEM had to apply the /BATCH qualifier to launch the whole environment to start the OpenVMS data collector or the SNMP extension. Otherwise the command failed. Beginning with VSI PERFDAT V3.2 ECO 1 any privileged user can execute the LAUNCH command without applying the /BATCH qualifier. It is still available for compatibility reasons.

Startup Queue

Logical PERFDAT$STARTUP_QUEUE

Almost all jobs of VSI PERFDAT have to run under the DQL$SRV user name and UIC[1]. Thus, if a user starts any of the VSI PERFDAT components using the LAUNCH command the batch command script

SYS$STARTUP:PERFDAT$STARTUP_BATCH.COM

is executed. This batch command script submits the appropriate startup script into a batch queue on behalf of the DQL$SRV user. This startup batch queue can be user defined with the logical PERFDAT$STARTUP_QUEUE. If the batch queue referred by the logical exists and its status is idle, busy or available the startup scripts are submitted into this batch queue.

Otherwise the PERFDAT$STARTUP_BATCH.COM creates and initializes a temporary batch queue to execute the startup scripts.

The logical PERFDAT$STARTUP_QUEUE has to be defined system wide.

```
$ DEFINE/SYSTEM PERFDAT$STARTUP_QUEUE queue-name
```

In order to define the logical permanently it is strongly recommended to define the logical in:
SYS$STARTUP:PERFDAT$LOGICALS_CUSTOM.COM.

If this file does not exist in SYS$STARTUP copy the template file PERFDAT$CFG:PERFDAT$LOGICALS_CUSTOM.TEMPLATE either into SYS$COMMON:[SYS$STARTUP] or SYS$SPECIFIC:[SYS$STARTUP] depending if you want to maintain node-specific logical definition files or you want to maintain just one common logical definition file which contains the node-specific logicals.

[1]The DQL$SRV user account and is automatically created when HP PERFDAT is installed on a system.

LOAD LICENSE

Loads a valid license key into the license table of the PERFDAT configuration database

Format

LOAD LICENSE *key*

Parameter

key
License key to be loaded

Description

Checks if the license *key* is valid and loads the key into the license table of the PERFDAT configuration database.

Example

This example shows how to load a temporary license

PerfDat_MGR> LOAD LICENSE <temp. license key>
PERFDAT_MGR-I-LICSUC, successfully loaded key /<temp. license key>/

LOAD METRIX_DESCRIPTION

Uploads the valid metric descriptors from an import file into the record descriptor table of the PERFDAT configuration database.

Format

LOAD METRIX_DESCRIPTION *import_filename*

Parameter

import_filename
Specifies the file that contains valid metric descriptors.

Description

Uploads the valid metric descriptors from an import file into the record descriptor table of the PERFDAT configuration database.

This command is reserved for use by VSI support only.

MODIFY AUTOSTART

This command invokes the auto-start configuration wizard to modify an existing entry in the auto-start table of the PERFDAT configuration.

Format

MODIFY AUTOSTART *node_name*

Parameter

node_name
Specifies the node name that refers to an existing entry in the auto-start table of the PERFDAT configuration database.

This parameter is mandatory. Wildcards are not supported.

Description

This command invokes the auto-start configuration wizard to modify an existing entry in the auto-start table of the PERFDAT configuration.

The auto-start table of the VSI PERFDAT configuration database contains all required start-up parameters to start performance data collections automatically when launching the OpenVMS data collector, the SNMP extension, the EVA extension or when an application starts that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT collection database. The OpenVMS data collector, the SNMP extension, the EVA extension and the VSI PERFDAT API access this table. These components of VSI PERFDAT check if any performance data collections are defined to be started on the local node and they start them automatically if appropriate auto-start entries exist. This is done by checking the content of every entry in this table.

The entries of that auto-start table are also read by the auto-trend engine to determine if any trend and capacity report shall be processed. It checks:

- if the local node itself is registered in the auto-start table
- if the local node is defined as the agent (=node that runs a SNMP or an EVA data collection) for any remote system (Tru64, Brocade, EVA, Solaris, Linux) registered therein.
- if the local node is configured to start application data collections automatically whenever applications are started on the local node that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database.

If this is the case the reports defined in the report profile table of the PERFDAT configuration database marked to be processed by the auto-trend engine and valid for these auto-start entries are processed (the auto-trend engine checks if the OS type of the auto-started performance data collection matches the OS type of the report profiles).

Depending on the /OS_TYPE qualifier the auto-start configuration wizard prompts the user for different inputs.

/OS_TYPE = OpenVMS

- Collection profile to auto-start when launching the OpenVMS data collector. This field is also read by the auto-trend engine to determine the source collection database for capacity and trend report processing.
- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting
  First introduced with V3.0, PERFDAT provides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable period of time the system manager will be alerted via OPCOM messages and user definable command procedures.
- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc. The default alert definition file is
  PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG
- Data flush time
  Each performance data collection started creates a new data file daily. The time entered at the data flush time prompt defines the time of day a new data file shall be created for the auto-started performance data collection. Enter a valid time string only.

/OS_TYPE = (Tru64, Brocade, Solaris, Linux)

- Collection profile to auto-start when launching the SNMP extension. This field is also read by the auto-trend engine to determine the source collection database for capacity and trend report processing.
- IP address of the remote node
- SNMP community string
- Agent node
  The agent node defines where to run the SNMP data collection and the node to run the auto-trend engine for processing data collected for the remote system referred by this entry.
- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting

First introduced with V3.0, PERFDAT provides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable period of time the system manager will be alerted via OPCOM messages and user definable command procedures.

- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc. Depending on the value of the /OS_TYPE qualifier the default alert definition file is:
  - TRU64
    PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG
  - BROCADE
    PERFDAT$CFG:PERFDAT_ALERT_BROCADE.CFG
  - No default alert definition files are available for SOLARIS and LINUX
- Data flush time
  Each performance data collection started creates a new data file daily. The time entered at the data flush time prompt defines the time of day a new data file shall be created for the auto-started performance data collection. Enter a valid time string only.

/OS_TYPE = EVA

- Collection profile to auto-start when launching the EVA extension.
  This field is also read by the auto-trend engine to determine the source collection database for capacity and trend report processing.
- EVA access device
  Enter the console access device to the EVA (HP StorageWorks Virtual Array) system you want to monitor. You can access the console of an EVA system only if the 'Console LUN ID' parameter of the EVA system is greater than zero. If the 'Console LUN ID' parameter is greater than zero, and you have executed the MCR SYSMAN IO AUTOCONFIGURE command you will get a 1GGAxxx device, where xxx = 'Console LUN ID' parameter value of the EVA system. This is the device you have to enter.
- Agent node
  The agent node defines where to run the EVA data collection and the node to run the auto-trend engine for processing data collected for the EVA system referred by this entry.
- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting
  First introduced with V3.0, PERFDAT provides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable

period of time the system manager will be alerted via OPCOM messages and user definable command procedures.

- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc. The default alert definition file is:
  - PERFDAT$CFG:PERFDAT_ALERT_EVA.CFG
- Data flush time
  Each performance data collection started creates a new data file daily. The time entered at the data flush time prompt defines the time of day a new data file shall be created for the auto-started performance data collection. Enter a valid time string only.

/OS_TYPE = *application-name*

- Collection profile to be used by the VSI PERFDAT API to auto-start an application data collection when a process of the application defined by the /OS_TYPE qualifier (*application-name* parameter) is started on the node defined by the auto-start *node_name* parameter of the ADD AUTOSTART command.
- Online access
- Auto report start time
  It defines the start date for data processing the next time the auto-trend engine is triggered (collected data are processed from this date/time on).
- Online alerting
  First introduced with V3.0, PERFDAT provides a performance alerting (watchdog) feature for real time monitoring of dedicated statistics collected by an active performance data collection. Whenever one of these statistics exceeds free definable thresholds for a definable period of time the system manager will be alerted via OPCOM messages and user definable command procedures.
- Alert definition file in case of online alerting is enabled.
  The alert definition file contains the alert blocks that define the statistics to monitor, the warning and critical threshold values, the file names of the user definable command procedures etc.

The predefined alert definition files

- PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG
- PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG
- PERFDAT$CFG:PERFDAT_ALERT_BROCADE.CFG
- PERFDAT$CFG:PERFDAT_ALERT_EVA.CFG

are part of the distribution kit. For detailed information about how to configure alert blocks within an alert definition file please refer to ENABLE ALERTcommand description.

Qualifier

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application that is referenced by the modified auto-start entry.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to modify the auto-start entry of any non-OpenVMS node and to start the appropriate auto-start configuration wizard the /OS_TYPE qualifier is mandatory.

If you are modifying the auto-start entry for an OpenVMS node you can omit the qualifier since the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS -> The node is an OpenVMS node.
- Tru64 -> The node is a Tru64 node.
- Brocade -> The node is a Brocade switch.
- EVA -> HP StorageWorks Virtual Array
- Solaris -> The node is a Solaris node
- Linux -> The Node is a Linux node
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Examples

The MODIFY AUTOSTART commands invokes the auto-start wizard that prompts you for all parameters that can be changed by the user.

Example 1
This example shows how to change the start-up parameters for the OpenVMS node VMSTM4 stored in the auto-start table.

In this example the auto-start collection profile is changes from DEFAULT to 2MIN. All other parameters are left unchanged.

```
PerfDat_MGR> MODIFY AUTOSTART VMSTM4

    Autostart Profile [DEFAULT] 2MIN↵
    Allow online access [No]↵
    Enter Auto Report Start date [ 1-MAR-2005] ↵
    Enable online alerting  [Yes]: ↵
```

```
    Enter alert definition file [PERFDAT$CFG:PERFDAT_ALERT_OPENVMS.CFG]: ↲
    Data flush (data file close) time [08:00:00] ↲
```

Example 2

This example shows how to change the start-up parameters for the Tru64 node named MIANIX stored in the auto-start table.

In this example the agent node (= node that runs the SNMP data collection and hosts the data) is changed from BSCXTC to VMSTM4, and the collection profile is changed from DEFAULT to 2MIN. All other parameters are left unchanged

```
PerfDat_MGR> MODIFY AUTOSTART MIANIX/OS_TYPE=TR64

    Autostart Profile [DEFAULT] 2MIN↲

    You are configuring a remote host -> enter the IP address of the remote host.
        You can enter the IP address or the full qualified IP host name.
        It is recommended to enter the IP Address.

    IP address of remote host  [16.55.40.10]: ↲
    Enter community string [public] ↲
    Enter Node to host the data [BCSXTC] VMSTM4 ↲
    Allow online access  [No] ↲
    Enter Auto Report Start date [10-MAR-2005]: ↲
    Enable online alerting  [Yes]: ↲
    Enter alert definition file [PERFDAT$CFG:PERFDAT_ALERT_TRU64.CFG]: ↲
    Data flush (data file close) time [03:00:00] ↲
```

Example 3

This example shows how to change the start-up parameters for the HP StorageWorks Virtual Array (EVA) system named EVA1 stored in the auto-start table.

In this example the agent node (= node that runs the EVA data collection and hosts the data) is changed from BSCXTC to VMSTM4, and the collection profile is changed from DEFAULT to 2MIN. All other parameters are left unchanged

```
PerfDat_MGR> MODIFY AUTOSTART EVA/OS_TYPE=EVA

    Autostart Profile [DEFAULT] 2MIN↲
EVA access device [$1$GGA90:]: ↲
    Enter Node to host the data [BCSXTC] VMSTM4 ↲
    Allow online access [Yes] ↲
    Enter Auto Report Start date [10-JAN-2008]: ↲
  Enable online alerting [No]: ↲
    Data flush (data file close) time [03:00:00] ↲
```

MODIFY PROFILE

This command modifies an existing entry (profile) in the collection profile table of the PERFDAT configuration database.

Format

MODIFY PROFILE *profile_name*

Parameter

profile_name
Defines the name of an existing collection profile to modify. The profile name references the collection profile in the collection profile table of the PERFDAT configuration database.

If you add a new profile the *profile_name* has to be unique for the system or application defined by the /OS_TYPE qualifier, but you can reuse the same *profile_name* if you define profiles for different systems or applications.

This parameter is mandatory. Wildcards are not supported.

Description

Any performance data collection is profile controlled. The collection profiles are stored in the collection profile table of the PERFDAT configuration database. This command reads an existing entry (profile) referenced by the *profile_name* parameter from that table and invokes the collection profile wizard for modifying the profile.

The collection profile wizard prompts the user for the sample interval and the metrics to be enabled. Since the metrics available for the supported systems (OpenVMS, Tru64, Brocade, EVA, Solaris, Linux, applications that use VSI PERFDAT API) differ, the profile collection wizard prompts for different inputs, depending on the /OS_TYPE qualifier.

For detailed description of the profile wizards for:

- OpenVMS
- TRU64
- BROCADE
- EVA
- Solaris
- Linux

please refer to the command description of ADD PROFILE

Qualifier

/ADVANCED
/NOADVANCED (default)

If the /ADVANCED qualifier is applied to configure an OpenVMS collection profile the profile wizard prompts you to enter threshold values whenever you enable one of the metrics listed below:

- CPU
- PROCESS
- USER
- IMAGE
- ACCOUNT
- XFCVOLUME
- XFCVOLUME.FILE
- DEVICE
- DEVICE.FILE
- DEVICE.PROCESS
- DEVICE.PROCESS.FILE
- LANADAPTER
- SCSPORT

Thresholds can be used to reduce the data rate to the collection file. For more information about setting thresholds see the DESCRIPTION section of the ADD PROFILE command description.

If the /ADVANCED qualifier is applied to modify an existing SNMP extension collection profile the profile wizard prompts you to enter the remote SNMP server listener port per metric to fetch the performance data.

This is of special interest if the SNMP server listener port on a remote system that provides performance data via SNMP is not the default SNMP listener port (161) or if several SNMP servers running on the remote system are listening on different ports providing different kind of performance data.

If the /ADVANCED qualifier is omitted or the command is entered with /NOADVANCED the user is neither prompted to enter threshold values for OpenVMS collection profiles nor to define the remote SNMP server listener port per metric for non-OpenVMS collection profiles. (All threshold for OpenVMS collection profiles = 0, remote SNMP server ports for all metrics of non-OpenVMS collection profiles = 161).

If you modify an existing EVA extension collection profile the /ADVANCED qualifier is ignored.

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the modified collection profile is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming

interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to modify an existing profile valid for non-OpenVMS systems to the collection profile table of the VSI PERFDAT configuration database the /OS_TYPE qualifier is mandatory.

If you are modifying an existing OpenVMS collection profile you can omit the qualifier since OpenVMS is the default, and the OpenVMS profile wizard is started.

The supported keywords for the value for the qualifier are:

- OpenVMS -> The node is an OpenVMS node.
- Tru64 -> The node is a Tru64 node.
- Brocade -> The node is a Brocade switch.
- EVA -> HP StorageWorks Virtual Array.
- Solaris -> The node is a Solaris system.
- Linux -> The node is a Linux system.
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of VSI PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Examples

Command to invoke the OpenVMS collection profile wizard to modify the collection profile NEW_ONE valid for OpenVMS:

```
PERFDAT_MGR> MODIFY PROFILE NEW_ONE
or
PERFDAT_MGR> MODIFYPROFILE NEW_ONE/OS_TYPE=OPENVMS
```

Command to invoke the Tru64 collection profile wizard to modify the collection profile NEW_ONE valid for Tru64:

```
PERFDAT_MGR> MODIFY PROFILE NEW_ONE/OS_TYPE=TRU64
```

Command to invoke the BROCADE collection profile wizard to modify the collection profile NEW_ONE valid for Brocade:

```
PERFDAT_MGR> MODIFY PROFILE NEW_ONE/OS_TYPE=BROCADE
```

Command to invoke the EVA collection profile wizard to modify the collection profile NEW_ONE valid for EVA systems:

```
PERFDAT_MGR> MODIFY PROFILE NEW_ONE/OS_TYPE=EVA
```

MODIFY REPORT

This command modifies an existing entry (profile) in the collection profile table of the PERFDAT configuration database.

Format

MODIFY REPORT *report_name*

Parameter

report_name
Defines the name of an existing report profile to modify. The profile name references the report profile in the report profile table of the PERFDAT configuration database.

This parameter is mandatory. Wildcards are not supported.

Description

Trend, capacity and baseline reports are extracted from performance data either via the auto-trend engine or manually via DQL$. In either case these reports are profile controlled. The report profiles are stored in the report profile table of the PERFDAT configuration database. This command invokes the report profile wizard to modify an existing report profile.

For detailed description of the report profile wizard please refer to the command description of ADD REPORT.

Qualifier

/OS_TYPE=*system*|*application-name*
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application the modified report profile is valid for.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to modify an existing report profile valid for non-OpenVMS systems to the report profile table of the VSI PERFDAT configuration database the /OS_TYPE qualifier is mandatory.

If you are modifying an existing profile that is valid for OpenVMS you can omit the qualifier since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS -> The node is an OpenVMS node.
- Tru64 -> The node is a Tru64 node.
- Brocade -> The node is a Brocade switch.
- RDB -> Imported RDB database performance data
- EVA -> HP StorageWorks Virtual Array.
- Solaris -> The node is a Solaris system.
- Linux -> The node is a Linux system.
- CACHE -> Imported CACHE database performance data
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

Example

See the examples in the command description of ADD REPORT.

SET ARCHIVE

The SET ARCHIVE command changes the performance data archiving control parameters in the volatile archive table.

Format

SET ARCHIVE

Parameter

None

Description

The SET command changes the performance data archiving control parameters in the volatile archive table. This is done by applying different qualifiers.

The performance data archiving process is a background task that can be started manually using the START ARCHIVE command or automatically when the OpenVMS data collector of VSI PERFDAT is started. Its main tasks are:

- Archiving performance data to the archive node on a daily basis
- Data housekeeping - purging log files and deleting expired performance data files (files that are older then the keep time defined by the /KEEP_DAYS qualifier) on the local node

The parameters of the volatile archive table actually control the behavior of the archiving process. Changing any control parameters by the SET command affects the behavior of the archiving process immediately.

Note

The SET ARCHIVE command does not affect the permanent database. Thus, after restarting the archiving process any archiving control settings previously changed by the SET command are gone.

Qualifier

/DISABLE

Advices the performance data archiving process to switch to passive mode. In passive mode the archiving process does not archive any performance data file independently of the archive date/time and keep time defined by the /KEEP_DAYS qualifier.

This qualifier has no effect on the housekeeping functionality of the archiving process. Housekeeping will be done in passive mode too.

This qualifier is mutual exclusive to the /ENABLE qualifier.

/ENABLE

Advices the performance data archiving process to switch to active mode. Active mode means that the archiving process starts data archiving in accordance to the archive date/time and keep time defined by the /KEEP_DAYS qualifier.

This qualifier has no effect on the housekeeping functionality of the archiving process. Housekeeping will be performed in active and passive mode.

This qualifier is mutual exclusive to the /DISABLE qualifier.

/KEEP_DAYS=number of days

The value of this qualifier defines how long (how many days) performance data shall be kept in the PERFDAT$DB_ARCHIVE directory on the local node before the archiving process deletes these data files.

/TIME=OpenVMS date/time

Specifies the date and time to trigger the next data archiving run. After the date and time specified by this qualifier has expired the data archiving will be re-triggered daily again. The new time of day to trigger the archiving process daily is the time defined by the time field of the value of this qualifier. (E.g. /TIME = 25-OCT-200509:00 -> daily retrigger at 09:00 starting from 26-OCT-2003)

The format is standard OpenVMS date/time format.

Example

PerfDat_MGR> SET ARCHIVE/ENABLE/TIME=24-SEP-200303:00/KEEP_DAYS=28

In this example the next archive date/time is set to 24-SEP-2003 03:00. From that date on data archiving is retriggered daily at 03:00. The archiving process is set to active. Any performance data collection file in the directory PERFDAT$DB_ARCHIVE will be deleted unconditionally if its creation date is older then the time the archiving process is triggered minus 28 days.

SETNAME_SERVER

This command is used to change the time to leave duration of the performance database file name cache entries.

Format

SETNAME_SERVER

Parameter

None

Description

This command is used to change the time to leave duration of the performance database file name cache entries.

As explained in the START NAME_SERVER command description the performance database file name cache service DQL_NAME provides a file name cache to all PERFDAT components. This cache contains the file header information of all PERFDAT data files locally stored.

The TTL (time to leave) parameter defines the time duration the entries in the performance database file name cache are valid. Every entry has to be updated once during the TTL duration by DQL_NAME process. If TTL duration time expires and the entries in the cache have not been updated for any reason the cache is marked invalid and from this time on all PERFDAT components will fetch file header information direct from the data files until the performance database file name cache service DQL_NAME starts processing again. Thus, the TTL parameter defines the cache entry life-time.

The default value of the TTL parameter is 30 minutes.

The TTL parameter is changed by assigning an integer value to the /TTL qualifier. The /TTL qualifier is mandatory. The value assigned to the /TTL qualifier defines the TTL duration in minutes. The new TTL value takes effect immediately after the current TTL period has expired.

The SET NAME_SERVER command only affects the current TTL setting of the performance database name server cache. For permanent change of the TTL value re-define the logical DQL$_NAME_SRV_TTL in SYS$STARTUP:DQL$LOGICALS.COM manually. Otherwise the default value (30 min) is used the next time DQL_NAME is restarted.

For more information about the performance database file name cache and the DQL_NAME service please refer to the START NAME_SERVER command description or to the manual VSI PERFDAT – Architecture and Technical Description.

Qualifier

/TTL=cache time to leave duration in minutes

The value assigned to the /TTL qualifier defines the new cache time to leave duration in minutes. The /TTL qualifier is mandatory. The new TTL value takes effect immediately after the current TTL period has expired.

The SET NAME_SERVER command only affects the current TTL setting of the performance database name server cache. For permanent change of the TTL value re-define the logical DQL$_NAME_SRV_TTL in SYS$STARTUP:DQL$LOGICALS.COM manually. Otherwise the default value (30 min) is used the next time DQL_NAME is restarted.

Example

This example shows how to change the current time to leave duration of the performance database file name cache.

```
PerfDat_MGR> SET NAME_SERVER/TTL=60

   Name Server Settings:
         TTL cache interval: 60 min

PERFDAT_MGR-I-DEFNAME, in order to make the TTL change permanent update the definition
                of logical DQL$_NAME_SRV_TTL in SYS$STARTUP:DQL$LOGICALS.COM.
```

SHOW ARCHIVE

Displays the startup control parameter settings for the archiving process stored in the archive control table of the PERFDAT configuration database and the actual control parameters in use stored in the volatile archive table.

Format

SHOW ARCHIVE

Parameter

None

Description

Displays the startup control parameter settings for the archiving process stored in the archive control table of the PERFDAT configuration database and the actual control parameters in use stored in the volatile archive table.

Example

```
PerfDat_MGR> SHOW ARCHIVE

Archive Definitions:

    Database Settings
        Enabled:            TRUE
        Time of Day:                02:00:00
        Days to keep:               30
.   Current Settings
        Enabled:            TRUE
        Next Archive Time: 6-AUG-2005 02:00:00.00
        Days to keep:               30
```

SHOW AUTOSTART

Displays the content of entries (auto-start nodes) stored in the auto-start table of the PERFDAT configuration database.

Format

SHOW AUTOSTART *node_name*

Parameter

node_name
Specifies the node name that refers to existing entries in the auto-start table of the PERFDAT configuration database.

PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within the *node_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *node_name* string but the asterisk (*) character as input for the *node_name* parameter for full wildcard operation.

This parameter is optional.

Description

Displays the content of the entries (auto-start nodes) stored in the auto-start table of the PERFDAT configuration database that matches the *node_name* parameter and the value of the /OS_TYPE qualifier.

PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within *node_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *node_name* string but the asterisk (*) character as input for the *node_name* parameter for full wildcard operation.

If you want to display all auto-start entries omit the *node_name* parameter or use an asterisk (*) and omit the /OS_TYPE qualifier.

The /OS_TYPE qualifiers can be applied to selectively display the auto-start entries of the auto-start table that refer to a specific systems or applications.

Qualifier

/OS_TYPE=system|application-name

The /OS_TYPE qualifiers can be applied to selectively display the auto-start entries of the auto-start table that refer to a specific system or application.

The supported keywords for the value for the qualifier are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database

For detailed information about the /OS_TYPE qualifier please see ADD AUTOSTART command description.

Example

Command to show all entries stored in the auto-start table:

```
PERFDAT_MGR>S HOW AUTOSTART
```
or
```
PERFDAT_MGR> SHOW AUTOSTART *
```

Command to show all OpenVMS entries stored in the auto-start table:

```
PERFDAT_MGR> SHOW AUTOSTART/OS_TYPE=OpenVMS
```
or
```
PERFDAT_MGR> SHOW AUTOSTART */OS_TYPE=OpenVMS
```

Command to show the OpenVMS entry that refers to the node VMSTM1 stored in the auto-start table:

```
PERFDAT_MGR> SHOW AUTOSTART VMSTM1/OS_TYPE=OpenVMS
```

Command to show the Tru64 entry that refers to the node MIANIX stored in the auto-start table:

```
PERFDAT_MGR> SHOW AUTOSTART MIANIX/OS_TYPE=Tru64
```

Command to show the EVA entry that refers to the node EVA system EVA1 stored in the auto-start table:

```
PERFDAT_MGR> SHOW AUTOSTART EVA1/OS_TYPE=EVA
```

SHOW COLLECTION

Shows the status of performance data collections active on the local node started with the collection profile defined by the profile-name parameter

Format

SHOW COLLECTION *profile_name*

Parameter

profile_name
Specifies the collection profile name an active collection was started with.

PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within *profile_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *profile_name* string but the asterisk (*) character as input for the *profile_name* parameter for full wildcard operation.

This parameter is optional.

Description

Shows the status of performance data collections active on the local node started with the collection profile defined by the *profile_name* parameter. The *profile_name* parameter is optional.

PERFDAT V3.0 and higher versions provide full wildcard support for the *profile_name* parameter. Asterisk (*) and percent sign (%) wildcard characters can be placed anywhere within the*profile_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *profile_name* string but the asterisk (*) character as input for the *profile_name* parameter for full wildcard operation.

The /OS_TYPE qualifiers can be applied to selectively display the status of active performance data collections for systems or applications specified by this qualifier.

The /NODE qualifier can be used to filter for specific nodes. PERFDAT V3.0 and higher versions provide full wildcard support for the value of the /NODE qualifier. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within the *node_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcards.

The SHOW COLLECTION command defines a number of global symbols that provides information about the status of the OpenVMS data collector, the SNMP extension, the EVA extension and the collection addressed by the SHOW COLLECTION command. These symbols are:

- $PERFDAT_ACTIVE
  If the value of this symbol is "TRUE" (string value) the OpenVMS data collector is running. Otherwise the value is "FALSE".

- $PERFDAT_EVA_ACTIVE
  If the value of this symbol is "TRUE" (string value) the EVA extension is running. Otherwise the value is "FALSE".

- $PERFDAT_SNMP_ACTIVE
  If the value of this symbol is "TRUE" (string value) the SNMP extension is running. Otherwise the value is "FALSE".

- $PERFDAT_COLL_ACTIVE
  If the value of this symbol is "TRUE" (string value)the collection addressed by the command is active. Otherwise the value is "FALSE".

These symbols simplify the creation of DCL script to monitor VSI PERFDAT status.

Qualifier

/ADVANCED
/NOADVANCED (default)

If the /ADVANCED qualifier is applied to show the status of an OpenVMS performance data collection the threshold values of the metrics enabled of the OpenVMS performance data collections are displayed.

Thresholds can be used to reduce the data rate to the collection file. For more information about setting thresholds see the DESCRIPTION section of the ADD PROFILE command description.

If the /ADVANCED qualifier is applied to display the status of non-OpenVMS performance data collections processed by the SNMP extension the remote SNMP server listener port used to fetch data for each metric configured are displayed.

The /ADVANCED qualifier is ignored for EVA performance data collections.

/BRIEF

Using the /BRIEF qualifier the collection profile names used to start the performance data collections, their system type, system node name and their current operational state are displayed but no additional control settings in order to get an overview which data collections are active on the local node.

The following keywords can be displayed in the status column of the output:

Status	Description
ACTIVE	The performance data collection has been successfully started and data sampling is in progress.
SCHEDULED: *time*	The performance data collection has been triggered with the SUBMIT COLLECTION command and is scheduled to be started at the time displayed.
INIT CONFIG	This operational state is only displayed for a EVA and SNMP performance data collection (Brocade, Solaris Linux etc.). The EVA and SNMP performance extension is attempting to solicit a response from the remote system (i.e. waiting for system authentication SNMP response). This status indicates that no data samples have been collected up to this point. This operational state should only be transient; if this is not the case then this indicates a possible misconfiguration.
NOT RESPONDING	This operational state is only displayed for EVA performance data collections. The EVA performance extension has successfully started the data collection and data samples have already been collected, but currently the EVA associated with the performance data collection does not respond to performance data fetch requests.
UNKNOWN	The operational state of the performance data collection is unknown due to communication problems between PERFDAT_MGR and the data collector running a particular collection. This operational state indicates a software problems. Restart the data collector associated with a particular data collection and the PEERFDAT_MGR utility.

Depending whether or not the /OS_TYPE qualifier applied, brief information is displayed about all data collections or only about these data collections that matches the /OS_TYPE filter criterion.

/NODE=*node_name*

Applying the /NODE qualifier displays status information about all performance collections active on/for the nodes that match the value (*node_name*) of this qualifier.

PERFDAT V3.0 and higher versions provide full wildcard support for the value of the /NODE qualifier. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within the *node_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcards.

/OS_TYPE=*system*|*application-name*

The /OS_TYPE qualifier can be applied to selectively display the status of performance data collections of a specific type (system or application).

Valid keywords are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database

In order to obtain more information about the /OS_TYPE qualifier please see the ADD PROFILEcommand description.

Examples

Command to display all performance data collection active on the local node

```
PERFDAT_MGR> SHOW COLLECTION /BRIEF
```
or
```
PERFDAT_MGR> SHOW COLLECTION */BRIEF
```

Command to display the full status of the OpenVMS performance data collection started with the collection profile DEFAULT

```
PERFDAT_MGR> SHOW COLLECTION DEFAULT/ADVANCED
```
or
```
PERFDAT_MGR> SHOW COLLECTION DEFAULT/OS_TYPE=OpenVMS/ADVANCED
```

Command to display the status of the OpenVMS performance data collection started with the collection profile DEFAULT without threshold information

```
PERFDAT_MGR> SHOW COLLECTION DEFAULT
```
or
```
PERFDAT_MGR> SHOW COLLECTION DEFAULT/OS_TYPE=OpenVMS
```

Command to display the full status of the performance data collection active on the local node for node MIANIX started with the collection profile DEFAULT

```
PERFDAT_MGR> SHOW COLLECTION DEFAULT/OS_TYPE=Tru64/NODE=MIANIX
```

Command to display the status of the EVA performance data collection active to monitor the HP StorageWorks Virtual Array (EVA) EVA1 on the local started with the collection profile DEFAULT

```
PERFDAT_MGR> SHOW COLLECTION DEFAULT/OS_TYPE=EVA/NODE=EVA1
```

Examples

The short DCL script shown below demonstrates the use of the global status symbols provided by the SHOW COLLECTION command to monitor the status of VSI PERFDAT.

The script checks if the OpenVMS data collector is running and a data collection started with the profile DEFAULT is active.

If the OpenVMS data collector it will be automatically restarted by the script. If the OpenVMS data collector is running but not the data collection DEFAULT, the data collection will be re-started.

```
$ MCR PERFDAT_MGR SHOW COLLECTION DEFAULT/OS=OPENVMS
$ IF $PERFDAT_ACTIVE .NES. "TRUE"
$ THEN
$         MCR PERFDAT_MGR LAUNCH PERFDAT
$         EXIT
$ ENDIF
$!
```

```
$ IF $PERFDAT_COLL_ACTIVE .NES. "TRUE"
$ THEN
$          MCR PERFDAT_MGR START COLL DEFAULT/AUTOSTART
$          EXIT
$ ENDIF
$!
$ EXIT
```

For detailed information about the status symbols provided by the SHOW COLLECTION command please refer to the description section.

SHOWNAME_SERVER

This command displays the current TTL setting of the performance database file name cache service DQL_NAME.

Format

SHOWNAME_SERVER

Parameter

None

Description

This command displays the current TTL setting of the performance database file name cache service DQL_NAME.

In case the DQL_NAME process is not running the user is prompted to (re)start it.

For more information about the performance database file name cache and the DQL_NAME service please refer to the START NAME_SERVER command description or to the manual VSI PERFDAT– Architecture and Technical Description.

Example

```
PerfDat_MGR> SHOW NAME_SERVER

  Name Server Settings:
        TTL cache interval: 60 min
```

SHOW PROCESS

Displays all VSI PERFDAT sub-systems and the processes associated with the sub-systems running on the local system.

Format

SHOW PROCESS

Description

Displays all VSI PERFDAT sub-systems and the processes associated with the sub-systems running on the local system.

The output contains 5 columns:

- VSI PERFDAT sub-system
  Name of the VSI PERFDAT subsystem. The subsystems listed below are displayed:
  - OpenVMS data collector
  - SNMP extension
  - EVA extension
  - Archive service
  - File name cache service
  - DQL interface
  - Auto-trend engine
- Process Name
  Name of the processes associated with the VSI PERFDAT sub-system.
- Pid
  Process ID
- State
  The following status keywords can be displayed:

State	Description
RUNNING	Process is running on the local system.
NOT STARTED	The VSI PERFDAT sub-system has not been started. Dependent on the configuration this may indicate a normal or an abnormal state. For example, if there is no EVA system to be monitored by the local system there is no need to start the EVA extension or if the local node is acting purely as a so called VSI PERFDAT Archive Node, but is not collecting local VSI PERFDAT OpenVMS performance data then there is no need to start the OpenVMS data collector.
MISSING	VSI PERFDAT sub-system has not been

	started, but it is strongly recommended to start the particular sub-system. If this state reoccurs after a restart please consult the log files to investigate the possible causes.
NO PROCESS	The “DQL interface” and “Auto-trend engine” sub-systems have no dedicated processes assigned. This means that the processes of these sub-systems are created event driven (i.e. GUI connection is established to the local node, report processing is triggered by the archive process). NO PROCESS means that the sub-system is available, but currently no process of the sub-system is currently running on the local node.

- Comment

  Additional information about the process or, if the sub-system is not started or missing, about to start the particular sub-system.

Example

```
PerfDat_MGR>SHOW PROCESS

VSI PERFDAT Sub-system    Process Name     Pid         State         Comment
-------------------------------------------------------------------------------
OpenVMS data collector    PERFDAT          246EA89A    RUNNING
SNMP Extension            PERFDAT_SNMP     ---         NOT STARTED   (Start cm...)
EVA Extension             PERFDAT_EVA      246EA8AD    RUNNING       (master p...)
                          PERFDAT_EVA_0    246EA8AE    RUNNING       (working ...)
                          PERFDAT_EVA_1    246EA8B1    RUNNING       (working ...)
Archive service           PERFDAT_ARCHIVE  246E9144    RUNNING
File name cache service   DQL_NAME         ---         MISSING       (Start cm...)
DQL interface             DQL$SRV_BG51298  246EA9D4    RUNNING       (DQL server)
                          DQL$SRV_BG51316  246EA9D8    RUNNING       (DQL server)
                          DQL$SRV_BG51334  246EA9E0    RUNNING       (DQL server)
                          PDBC$SR_BG51256  246EA9BC    RUNNING       (GUI conn...)
                          PDBC$SR_BG51311  246EA9D7    RUNNING       (GUI conn...)
                          PDBC$SR_BG51329  246EA9DE    RUNNING       (GUI conn...)
Auto-trend engine                                      NO PROCESS
```

SHOW PROFILE

Displays the content of existing collection profile(s) stored in the collection profile table of the PERFDAT configuration database.

Format

SHOW PROFILE *profile_name*

Parameter

profile_name
Specifies existing collection profile(s) stored in the collection profile table of the PERFDAT configuration database.

PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within *profile_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *profile_name* string but the asterisk (*) character as input for the *profile_name* parameter for full wildcard operation.

The parameter is optional.

Description

Displays the content of existing collection profile(s) stored in the collection profile table of the PERFDAT configuration database. The *profile_name* parameter is optional.

PERFDAT V3.0 and higher versions provide full wildcard support for the *profile_name* parameter. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within *profile_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *profile_name* string but the asterisk (*) character as input for the *profile_name* parameter for full wildcard operation.

The /OS_TYPE qualifiers can be applied to selectively display collection profiles valid for particular systems or applications.

Qualifier

/ADVANCED
/NOADVANCED (default)

If the /ADVANCED qualifier is applied for OpenVMS collection profiles the profile wizard displays the threshold values configured for the metrics listed below:

- CPU
- PROCESS
- USER
- IMAGE
- ACCOUNT
- XFCVOLUME
- XFCVOLUME.FILE
- DEVICE
- DEVICE.FILE
- DEVICE.PROCESS
- DEVICE.PROCESS.FILE
- LANADAPTER
- SCSPORT

Thresholds can be used to reduce the data rate to the collection file. For more information about setting thresholds see the DESCRIPTION section of the ADD PROFILE command description.

If the /ADVANCED qualifier is applied to display SNMP collection profiles the profile wizard displays the remote SNMP server listener port configured for each metric.

The /ADVANCED qualifier is ignored for EVA collection profiles.

/BRIEF

Using the /BRIEF qualifier the existing collection profile names stored in the collection profile table of the PERFDAT collection database are displayed but no content in order to get an overview which collections profiles are already defined.

Depending whether or the /OS_TYPE qualifier is applied brief information is displayed about all collection profiles or only about these collection profiles stored in the collection profile table that are valid for the operating system defined by the /OS_TYPE qualifier.

/OS_TYPE=*system*|*application-name*

The /OS_TYPE qualifiers can be applied to selectively display collection profiles valid for particular systems or applications.

Valid keywords are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that has the VSI PERFDAT API implemented. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

In order to obtain more information about the /OS_TYPE qualifier please see the ADD PROFILEcommand description.

Examples

Command to display the content of the collection profile NEW_ONE valid for OpenVMS stored in the collection profile table:

```
PERFDAT_MGR>SHOW PROFILE NEW_ONE
or
PERFDAT_MGR>SHOW PROFILE NEW_ONE/OS_TYPE=OPENVMS
```

Command to display the content of the collection profile NEW_ONE valid for Tru64 stored in the collection profile table:

```
PERFDAT_MGR> SHOW PROFILE NEW_ONE/OS_TYPE=TRU64
```

Command to display the content of the collection profile NEW_ONE valid for Brocade switches stored in the collection profile table:

```
PERFDAT_MGR> SHOW PROFILE NEW_ONE/OS_TYPE=BROCADE
```

Command to display the content of the collection profile NEW_ONE valid for HP StorageWorks Virtual Array (EVA) systems stored in the collection profile table:

```
PERFDAT_MGR> SHOW PROFILE NEW_ONE/OS_TYPE=EVA
```

SHOW REPORT

Displays the content of existing report profile(s) stored in the report profile table of the PERFDAT configuration database.

Format

SHOW REPORT *report_name*

Parameter

report_name
Specifies existing report profile(s) stored in the report profile table of the PERFDAT configuration database.

PERFDAT V3.0 and higher versions provide full wildcard support. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within *report_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *report_name* string but the asterisk (*) character as input for the *report_name* parameter for full wildcard operation.

The parameter is optional.

Description

Displays the content of existing report profile(s) stored in the report profile table of the PERFDAT configuration database. The *report_name* parameter is optional.

PERFDAT V3.0 and higher versions provide full wildcard support for the *report_name* parameter. Asterisk (*) and per cent sign (%) wildcard characters can be placed anywhere within *report_name* string. Earlier versions of PERFDAT (V2.3 and lower) do not support wildcard characters within the *report_name* string but the asterisk (*) character as input for the *report_name* parameter for full wildcard operation.

The /OS_TYPE qualifiers can be applied to selectively display report profiles valid for particular systems or applications.

Qualifier

/BRIEF

Using the /BRIEF qualifier the existing report profile names stored in the report profile table of the PERFDAT collection database are displayed but no content in order to get a quick overview which report profiles are already defined.

Depending whether or not the /OS_TYPE qualifier is applied brief information is displayed about all report profiles or only about these report profiles stored

in the collection profile table that are valid for the operating system defined by the /OS_TYPE qualifier.

/OS_TYPE=*system*|*application-name*

The /OS_TYPE qualifiers can be applied to selectively display report profiles valid for particular systems or applications.

Valid keywords are:

- OpenVMS
- Tru64
- Brocade
- RDB
- EVA
- Solaris
- Linux
- Name of any application that has the VSI PERFDAT API implemented. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

In order to obtain more information about the /OS_TYPE qualifier please see the ADD REPORT command description.

Examples

Command to display the content of the report profile NEW_ONE valid for OpenVMS stored in the report profile table:

```
PERFDAT_MGR> SHOW REPORT NEW_ONE
```
or
```
PERFDAT_MGR> SHOW REPORT NEW_ONE/OS_TYPE=OPENVMS
```

Command to display the content of the report profile NEW_ONE valid for Tru64 stored in the report profile table:

```
PERFDAT_MGR> SHOW REPORT NEW_ONE/OS_TYPE=TRU64
```

Command to display the content of the report profile NEW_ONE valid for Brocade switches stored in the report profile table:

```
PERFDAT_MGR> SHOW REPORT NEW_ONE/OS_TYPE=BROCADE
```

Command to display the content of the report profile NEW_ONE valid for HP StorageWorks Virtual Array (EVA) systems stored in the report profile table:

```
PERFDAT_MGR> SHOW REPORT NEW_ONE/OS_TYPE=EVA
```

SHOW VERSION

This command displays the version of PERFDAT.

Format

SHOW VERSION

Description

This command displays the version of PERFDAT. The version information is directly requested from the SW-components active on the local node. Thus, the command fails if PERFDAT is down.

SHUTDOWN

Shutdown of the whole PERFDAT environment or parts of it depending on the shutdown_keyword parameter.

Format

SHUTDOWN shutdown_keyword

Parameter

shutdown_keyword
Specifies the parts of the PERFDAT environment to be shut down. Supported keywords are:

- ALL
- PERFDAT
- PERFDAT_EVA
- PERFDAT_SNMP

Description

Shutdown of the whole PERFDAT environment or parts of it depending on the shutdown_keyword parameter.

SHUTDOWN ALL

Shutdown of the whole PerfDat environment.

Actions performed:

- Shutdown of the archiving process.
- Shutdown of the performance database file name cache service
- Stops all active collections of the OpenVMS data collector
- Stops all active collections of the EVA extension
- Stops all active collections of the SNMP extension
- Shuts down the OpenVMS data collector process
- Shuts down the EVA extension master process
- Shuts down the SNMP extension master process

SHUTDOWN PERFDAT

Shutdown of the OpenVMS performance data collector and the archiving process.

Actions performed:

- Stops all active collections of the OpenVMS data collector
- Shuts down the OpenVMS data collector process

SHUTDOWN PERFDAT_EVA

Shutdown of the VSI PERFDATEVA extension (master &working processes).

Actions performed:

- Stops all active collections of the EVA extension
- Shuts down the EVA extension master process

SHUTDOWN PERFDAT_SNMP

Shutdown of the VSI PERFDATSNMP extension (master & agent processes).

Actions performed:

- Stops all active collections of the SNMP extension
- Shuts down the SNMP extension master process

START ARCHIVE

Starts the auto archiving process PERFDAT_ARCHIVE.

Format

START ARCHIVE

Parameter

None

Description

Starts the auto archiving process PERFDAT_ARCHIVE.

START COLLECTION

Start a new performance data collection using an existing collection profile stored in the collection profile table of the VSI PERFDAT configuration database.

Format

START COLLECTION *profile_name*

Parameter

profile_name
Defines the name of an existing collection profile stored in the collection profile table of the VSI PERFDAT configuration database. The use of wildcard characters is not permitted.

Description

Start a new performance data collection. The *profile_name* parameter specifies an existing collection profile in the collection profile table of the VSI PERFDAT configuration database. This profile contains the control parameters for the newly started performance data collection.

Depending on the keyword applied to the /OS_TYPE qualifier the start request is forwarded either to the OpenVMS data collector, the SNMP extension, the EVA extension or the application that used the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database.

If the /OS_TYPE qualifier is omitted or /OS_TYPE=OpenVMS the start request is forwarded to the OpenVMS data collector. One can start up to three concurrent OpenVMS performance data collection.

If one of the supported keywords

- TRU64
- BROCADE
- Solaris
- Linux

is applied to the /OS_TYPE qualifier the start request is forwarded to the SNMP extension to start a non-OpenVMS data collection. Only one data collection per non-OpenVMS system can be started via the SNMP extension.

If you start a SNMP data collection for a non-OpenVMS system you have to apply the qualifiers listed below:

- /ADDRESS mandatory
- /NODE mandatory
- /OS_TYPE mandatory
- /SOURCE_ADDRESS optional

If you apply the keyword:

- EVA

to the /OS_TYPE qualifier, the start request is forwarded to the EVA extension to start a HP StorageWorks Virtual Array (EVA) performance data collection.

If you start a EVA performance data collection you have to apply the qualifiers listed below:

- /DEVICE mandatory
- /NODE mandatory
- /OS_TYPE mandatory
- /OPENVMS_STYLE optional
- /INSTANT_UPDTE optional

In order to start an application data collection the only the /OS_TYPE qualifier has to be applied. It specifies the application which uses the VSI PERFDATAPI to insert data into the distributed VSI PERFDAT performance database. If components (programs) of such an application are started on all cluster members the application data collection is started cluster-wide.

For detailed information about the architecture of the VSI PERFDAT API and how to manage application data collections please refer to the manual:

- VSI PERFDAT - API user's guide

The use of wildcards in the *profile_name* parameter is not permitted.

If you manually start a performance data collection with the START COLLECTION command online alerting is disabled by default. If you want to enable online alerting for the performance data collection actually started enter the ENABLE ALERT command afterwards.

Qualifier

/ADDRESS=*node_name*

This qualifier is mandatory and valid for starting a SNMP data collection.

It specifies the IP-address of the remote system. You can either enter the full qualified node name or the IP-address. It is recommended to apply the IP-address.

This qualifier requires the /OS_TYPE qualifier to be defined too. Otherwise the command fails.

This qualifier is mandatory if you start a new data collection for a non-OpenVMS system via the SNMP extension by applying one of the supported keywords

- TRU64
- Brocade
- Solaris
- Linux

to the /OS_TYPE qualifier.

/AUTOSTART

If you apply the /AUTOSTART qualifier all collection startup parameters required to start a data collection are fetched from the auto-start table of the VSI PERFDAT configuration database if a valid entry exists for the local node. If no valid entry exists the command fails.

The /AUTOSTART qualifier can only be applied to start OpenVMS data collections. If you start non-OpenVMS data collections (i.e. remote data collections via the SNMP extension) with the /AUTOSTART qualifier the qualifier is ignored.

/COMMUNITY="*community string"*

This qualifier is mandatory and valid for starting a SNMP data collection.

If you started a SNMP collection manually with the START COLLECTION command, the statistics (performance data) of each metric are requested with a default community string. The default community string depends on the value assigned to the /OS_TYPE qualifier which is mandatory when starting an SNMP performance data collection The default community string for the supported OS types is stored in the record descriptor table of the VSI PERFDAT configuration database.

- Tru64 public
- Brocade Switches FibreChannel
- Solaris public
- Linux public

The community string can be (re)defined by applying the /COMMUNITY qualifier. In case the /COMMUNITY qualifier is present, the string value assigned is used as the community string to fetch data from the remote host via SNMP *GET* requests. Since the SNMP community string check is case sensitive it is strongly recommended to use quotation marks to specify the community string. If quotation marks are omitted the community string is converted to upper case.

/DEFERRED_WRITE=*deferred write timer [msec]*

If VSI PERFDAT data collections are started on several nodes using the same sample interval, and the VSI PERFDAT data disks of all these nodes are configured on the same external storage (i.e. EVA array) this may cause I/O bursts at the end of each sample interval since all data collections try to write the collected data records to their data files concurrently.

To overcome such I/O bursts VSI PERFDAT provides the deferred data write option for OpenVMS and SNMP data collections. This means that the data records are not immediately inserted into the data files at the end of a sample interval but after the deferred data write timer has expired. The value assigned to the /DEFERRED_WRITE qualifier defines the time delay (deferred

data write timer) in milliseconds between the end of a sample interval and the time the data records are inserted into the data files. Thus, if OpenVMS or SNMP data collections are started with different deferred data write timers I/O bursts can be inhibited even if the VSI PERFDAT data disk of all nodes within an environment access the same external storage.

The deferred write timer must not be greater than half of the sample interval defined in the collection profile used to start the OpenVMS or SNMP collection. If the user assigns a value to the /DEFERRED_WRITE qualifier that is greater than half of the sample interval the deferred write timer is automatically set to half of the sample interval.

If the /DEFERRED_WRITE qualifier is omitted or the value assigned is zero the deferred write option is disabled for the data collection.

The /DEFERRED_WRITE qualifier is not supported for EVA performance data collection.

/DEVICE=*EVA access device (1GGAxxx)*

This qualifier is mandatory and valid for starting a HP StorageWorks Virtual Array (EVA) data collection.

In order to collect performance data of an EVA system, VSI PERFDAT has to have access to the console of the EVA system. OpenVMS can access the console of an EVA system only if the 'Console LUN ID' parameter of the EVA system is greater than zero. If the 'Console LUN ID' parameter is greater than zero, and you have executed the MCR SYSMAN IO AUTOCONFIGURE command a 1GGAxxx device will be available, where xxx = 'Console LUN ID' parameter value of the EVA system. This is the access device to the EVA console.

/FLUSH_TIME=*time of day*

Each performance data collection started creates a new data file daily. With the /FLUSH_TIME qualifier you can define the time of day the new data file shall be created. If you omit the qualifier new data files are created at day change.

Enter a valid time string only. Otherwise the start command fails.

/INSTANT_UPDATE
/NOINSTANT_UPDATE (default)

The optional /INSTANT_UPDATE qualifier is only valid for EVA (HP StorageWorks Enterprise Array) data collections.

The EVA extension of VSI PERFDAT automatically detects EVA configuration changes (i.e. new virtual disks) without being connected to the SAN appliance. In order to collect performance data for newly configured EVA items (virtual disk, disk groups, physical disks, host connections) a full EVA configuration

scan has to be performed in order to update the friendly name table of the EVA data collection. Due to the design of the EVA management interface (management commands are serialized) an EVA configuration scan of the EVA extension may slowdown any other utilities that access the EVA including Command View.

This qualifier defines whether or not an EVA configuration scan is triggered instantaneously when an EVA configuration change is detected. If the instantaneous configuration update option is disabled the EVA configuration scan is scheduled to be executed at midnight. Thus, the EVA extension does not influence active Command View sessions. The penalty incurred with this setting is that the performance data of any new configuration item is not collected until the EVA configuration scan has been performed (i.e. at midnight).

If the user wants to start an EVA data collection with instantaneous configuration option enabled the /INSTANT_UPDATE qualifier has to be applied. If the qualifier is omitted or the /NOINSTANT_UPDATE qualifier is applied the EVA collection is started with the instantaneous configuration option disabled.

/NODE=*node_name*

This qualifier is mandatory and valid for starting a HP StorageWorks Virtual Array (EVA) data collection and SNMP data collections.

It specifies the name of the remote system when starting a SNMP or EVA performance data collection.

This qualifier requires the /OS_TYPE qualifier to be defined too. Otherwise the command fails.

Supported keywords are:

- TRU64
- Brocade
- EVA
- Solaris
- Linux

/OPENVMS_STYLE
/OPENVMS_STYLE (default)

The optional /OPENVMS_STYLE qualifier is only valid for EVA (HP StorageWorks Enterprise Array) data collections. It defines whether the performance data of a virtual disk with an OS unit ID assigned that is greater than zero will be stored using the OpenVMS FC device format (1DGAxxx, where xxx = OS unit ID of the virtual disk) or with its friendly name assigned by CV/EVA (CommandView/EVA).

/OS_TYPE=*system*|*application-name*
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier defines the system or application to start a new data collection.

VSI PERFDAT provides a SNMP extension to monitor the performance of non-OpenVMS nodes via SNMP, an EVA extension to monitor the performance of EVA (HP StorageWorks Virtual Array) systems and an application programming interface that can be used by any application to insert data into the distributed VSI PERFDAT performance database. In order to start a new performance data collection the /OS_TYPE qualifier is mandatory.

If one starts a new OpenVMS performance data collection this qualifier can be omitted since OpenVMS is the default.

The supported keywords for the value for the qualifier are:

- OpenVMS -> The node is an OpenVMS node.
- Tru64 -> The node is a Tru64 node.
- Brocade -> The node is a Brocade switch.
- EVA -> HP StorageWorks Virtual Array.
- Solaris -> The node is a Solaris system.
- Linux -> The node is a Linux system.
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

The list of supported remote systems may extend with the next releases of PERFDAT. Thus, the list of supported keywords for the /OS_TYPE qualifier may extend too.

/SHARE
/NOSHARE (default)

If you start a data collection with the /SHARE qualifier the data of the actual collection (= data stored in the data collection file actually accessed by the OpenVMS data collector, SNMP extension or EVA extension) is online accessible via the DQL interface

Note

Using the /SHARE qualifier makes data online accessible but the penalty is that the write performance of the data collector decreases and in consequence overall system performance may suffer due to excessive locking activity.

The /SHARE qualifier is ignored if one starts an application data collection. Due to the design of the VSI PERFDAT API application data collection files are always opened shareable.

/SOURCE_ADDRESS=*IP-address*

This optional qualifier is valid for starting a SNMP data collection.

The /SOURCE_ADDRESS qualifier can be applied to define the source IP address of the UDP/IP socket created and used by the VSI PERFDAT SNMP extension to request SNMP performance data from a particular non OpenVMS system.

If the /SOURCE_ADDRESS qualifier is omitted the IP address returned by a name server lookup request for the node name of the local system is used.

Examples

Example 1

Command to start a new OpenVMS performance data collection on the local node using the collection profile NEW_ONE. Since the /SHARE qualifier is not present performance data are not inline accessible as long the OpenVMS data collector accesses the file.

```
PERFDAT_MGR> START COLLECTION NEW_ONE
or
PERFDAT_MGR> START COLLECTION NEW_ONE/OS_TYPE=OPENVMS
```

If data of the data collection file actually accessed by the OpenVMS data collector the collection have to be started with the /SHARE qualifier.

```
PERFDAT_MGR> START COLLECTION NEW_ONE/SHARE
or
PERFDAT_MGR> START COLLECTION NEW_ONE/OS_TYPE=OPENVMS/SHARE
```

Example 2

Command to start a new performance data collection on the local node for the Tru64 node MIANIX using the collection profile NEW_ONE. Since the /SHARE qualifier is not present performance data are not inline accessible as long the OpenVMS data collector accesses the file. The IP address of node MIANIX is 16.55.41.74.

```
PERFDAT_MGR> START COLLECTION NEW_ONE/OS_TYPE=TRU64
                                        /NODE=MIANIX/ADDRESS=16.55.41.74
```

If data of the data collection file actually accessed by the OpenVMS data collector the collection have to be started with the /SHARE qualifier.

```
PERFDAT_MGR> START COLLECTION NEW_ONE/OS_TYPE=TRU64
                                      /NODE=MIANIX/ADDRESS=16.55.41.74/SHARE
```

Example 3

Command to start a new EVA performance data collection for the HP StorageWorks Virtual Array EVA1 using the collection profile NEW_ONE. The EVA console access device is 1GGA90. The virtual disks will be stored using their friendly names even if their OS ID is greater than zero, since the

/NOOEPNVMS_STYLE qualifier is applied. The /SHARE qualifier is applied. Thus, data collected will be online accessible.

```
PERFDAT_MGR> START COLLECTION NEW_ONE/OS_TYPE=EVA
                    /NODE=EVA1/DEVICE=$1$GGA90:/NOOPENVMS_STYLE/SHARE
```

START NAME_SERVER

Starts the performance database file name cache service DQL_NAME.

Format

START NAME_SERVER

Parameter

None

Description

This command starts the performance database filename cache service DQL_NAME.

The distributed performance database is organized in way such that there is no persistent root file for any PERFDAT performance database on disk (see manual VSI PERFDAT– Architecture and Technical description). All meta-data (field and record descriptors, data link descriptors, index reference table descriptor …) necessary to access the data is stored in the header of each physical storage area. The advantage is that data files can be moved to any OpenVMS node and the data file stays read accessible without any additional actions such as data conversion, unload and load operations. On the other hand the meta-data have to fetched again any time a user connects to the distributed performance database via the DQL$ utility or the GUI in order to create a virtual root file (data link cache) required to access the data.

Prior to VSI PERFDATV3.2 the meta-data were fetched by performing a full data file header scan on all members of the PERFDAT community. Thus, prior to VSI PERFDATV3.2 the initial connect request to the distributed performance database took few seconds up to minutes.

In order to solve that performance issue VSI PERFDAT provides the performance database file name cache service DQL_NAME first introduced with VSI PERFDAT V3.2.

The performance database file name cache service DQL_NAME provides a database file name cache to all PERFDAT components that contains full header information about all PERFDAT database files locally stored. As long as the performance database file name cache service DQL_NAME is available and the database file name cache is marked valid all PERFDAT components obtain database file header information from that cache rather than scanning the files on disk. Thus, the initial connect request speeds up dramatically (ten times and more) compared to PERFDAT V3.1 and lower versions of PERFDAT.

The file name cache is updated by the DQL_NAME process:

- periodically once per TTL (time to leave) duration.

- whenever a PERFDAT component creates, renames or deletes a database file.

The TTL (time to leave) parameter defines the time duration the entries in the performance database file name cache are valid. Every entry has to be updated once during the TTL duration by DQL_NAME process. If TTL duration time expires and the entries in the cache have not been updated for any reason the cache is marked invalid and from this time on all PERFDAT components will fetch the file header information direct from disk on connect requests until the performance database file name cache service DQL_NAME starts processing again. Thus, the TTL parameter defines the cache entry lifetime.

The default value of the TTL parameter is 30 minutes.

Note

If you delete any PERFDAT data file in the directory PERFDAT$DB manually the file name cache will not be updated automatically until TTL time expires. Thus, in this case you have to trigger a rebuild of the performance database file name cache manually to keep the cache up to date. To perform a cache rebuild you can either use the FLUSH NAME_SERVER command or you can stop the file name cache service with the STOP NAME_SERVER command and restart it again using this command (has the same effect as the FLUSH NAME_SERVER command). If you run PERFDAT in a cluster you have to rebuild the file name cache on all cluster members.

You do not have to start the performance database file name cache service DQL_NAME manually after re-starting the PERFDAT environment/DQL interface. This is automatically done by the PERFDAT and DQL startup routines.

Example

```
PerfDat_MGR> START NAME_SERVER

DQL-I-STARTUP, Starting DQL name server

%RUN-S-PROC_ID, identification of created process is 2060066B

PERFDAT_MGR-I-STARTSUCC, name server started
```

STOP ARCHIVE

Stops the archiving process PERFDAT_ARCHIVE.

Format

STOP ARCHIVE

Parameter

None

Description

Stops the archiving process PERFDAT_ARCHIVE.

STOP COLLECTION

Stops active performance data collections started with the collection profile defined by the *profile_name* parameter.

Format

STOP COLLECTION *profile_name*

Parameter

profile_name
Specifies the collection profile name an active collection was started with. This parameter is mandatory. The use of wildcard characters is not permitted.

Description

Stops active performance data collections started with the collection profile defined by the *profile_name* parameter.

The use of wildcards in the *profile_name* parameter is not is not permitted.

Depending on the keyword applied to the /OS_TYPE qualifier the stop request is forwarded either to the OpenVMS data collector, the SNMP extension, the EVA extension or the application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database.

If the /OS_TYPE qualifier is omitted or /OS_TYPE=OpenVMS the stop request is forwarded to the OpenVMS data collector.

Stopping active performance data collections for a non-OpenVMS system requires the /OS_TYPE qualifier to be applied. If you do not apply the /NODE qualifier when stopping a SNMP or EVA data collections, all data collections of the type defined by the /OS_TYPE qualifier are stopped. Otherwise only the data collections active for the node specified by the /NODE qualifier is stopped.

Due to the design of the VSI PERFDAT API application data collections are always stopped cluster-wide. For detailed information about the architecture of the VSI PERFDAT API and how to manage application data collections please refer to the manual:

- VSI PERFDAT - API user's guide

Qualifier

/NODE=*node_name*

This optional qualifier can be applied if one wants to stop a data collection for a specific node.

If you do not apply the /NODE qualifier when stopping non-OpenVMS data collections the performance data collections active on the local node for all nodes running the operating system defined by the /OS_TYPE qualifier are stopped.

The /NODE qualifier is ignored if one stops an application data collection. Due to the design of the VSI PERFDAT API application data collections are always stopped cluster-wide. For detailed information about the architecture of the VSI PERFDAT API and how to manage application data collections please refer to the manual:

- VSI PERFDAT - API user's guide

/OS_TYPE=system|application-name
/OS_TYPE=OpenVMS (default)

The /OS_TYPE qualifier is mandatory if one stops data collections for non-OpenVMS systems. If an OpenVMS data collection is stopped the qualifier can be omitted since OpenVMS is the default.

Supported keywords are:

- OpenVMS
- Tru64
- Brocade
- EVA
- Solaris
- Linux
- Name of any application that uses the VSI PERFDAT API to insert data into the distributed VSI PERFDAT performance database. An application database descriptor with the same name must exist in the record descriptor table of the VSI PERFDAT configuration database.

If no additional qualifier but /OS_TYPE is entered with the STOP command the performance data collections for all nodes that have been started with the collection profile defined by the profile_name parameter and that match the /OS_TYPE value are stopped.

In order to obtain more information about the /OS_TYPE qualifier please see the ADD PROFILE command description.

Examples

Example 1

Command to stop the OpenVMS performance data collection using the collection profile NEW_ONE:

```
PERFDAT_MGR> STOP COLLECTION NEW_ONE
or
PERFDAT_MGR> STOP COLLECTION NEW_ONE/OS_TYPE=OPENVMS
```

Example 2

Command to stop all Tru64 performance data collections active on the local node using the collection profile NEW_ONE:

```
PERFDAT_MGR> STOP COLLECTION NEW_ONE/OS_TYPE=TRU64
```

Example 3

Command to stop the Tru64 performance data collections for node MIANIX using the collection profile NEW_ONE:

```
PERFDAT_MGR> STOP COLLECTION NEW_ONE/OS_TYPE=EVA/NODE=MIANIX
```

Example 4

Command to stop the EVA performance data collections for the HP StorageWorks Virtual Array (EVA) EVA1 using the collection profile NEW_ONE:

```
PERFDAT_MGR> STOP COLLECTION NEW_ONE/OS_TYPE=EVA/NODE=EVA1
```

STOP NAME_SERVER

Stops the performance database file name cache service DQL_NAME.

Format

STOP NAME_SERVER

Parameter

None

Description

Stops the performance database file name cache service DQL_NAME and the database file cache is marked invalid.

For more information about the performance database name cache service please refer to the START NAME_SERVER command description or to the manual VSI PERFDAT – Architecture and Technical Description.

Example

```
PerfDat_MGR> STOP NAME_SERVER
PERFDAT_MGR-I-SUCCESS, name server process stopped
```

SUBMIT COLLECTION

Schedules a new performance data collection to be started by the scheduler of PERFDAT using an existing collection profile stored in the collection profile table of the PERFDAT configuration database.

Format

SUBMIT COLLECTION *profile_name*

Parameter

profile_name
Defines the name of an existing collection profile stored in the collection profile table of the PERFDAT configuration database. The use of wildcard characters is not permitted.

Description

Schedules a new performance data collection to be started using the collection profile profile-name. In contrast to the START COLLECTION command the performance data collection is not started directly, but the command is forwarded to the scheduler of PERFDAT.

This command is only valid for submitting OpenVMS collections

The OpenVMS data collector checks if any active (already started) or scheduled performance data collection is using the same collection profile. In that case the submit command is rejected.

Depending on the qualifiers applied, the data collection is periodically retriggered or a single shot collection.

Submitting a performance collection that is retriggered periodically by applying the /AFTER, /UNTIL and /RESTART_INTERVAL qualifiers permanently allocates a collection slot of the OpenVMS data collector as long as it is stopped manually even if the collection is not active. This is done in order to guarantee that the performance collection can be restarted in time.

The maximal number of concurrent collections is limited to three.

If you schedule a performance data collection to be started by the PERFDAT scheduler, online alerting is disabled by default. If you want to enable online alerting for the performance data collection actually scheduled enter the ENABLE ALERT command afterwards.

Qualifier

/AFTER = OpenVMS date/time format

Specifies the time the scheduler starts the performance data collection the first time. If you omit the qualifier the collection is started immediately.

/RESTART_INTERVAL = OpenVMS delta date/time format

With the /RESTART_INTERVAL qualifier you can define cyclic performance collection activations. One the performance collection is started it will be periodically re-activated with a time delay defined by this qualifier.

If you omit the qualifier the scheduled collection is a single shot action.

The qualifier requires the /AFTER and /UNTIL qualifier to be defined too.

Make sure that the delta time defined by this qualifier is greater then the time defined by the /AFTER qualifier minus the time defined by the /UNTIL qualifier.

/SHARE
/NOSHARE (default)

If you start a performance data collection with the /SHARE qualifier the data of the actual collection is online accessible via the DQL interface.

Note

Using the /SHARE qualifier makes data online accessible but the penalty is that the write performance of the data collector decreases and in consequence overall system performance may suffer due to excessive locking activity.

/UNTIL = OpenVMS date/time format

Specifies the time the scheduler stops the performance collection. If you omit the qualifier the collection stays active until you stop it manually.

Example

In this example the collection profile DEFAULT is used to submit a new OpenVMS performance data collection. This performance data collection will be started at 05-AUG-200517:00 and will be automatically stopped at 06-AUG-200500:00. Since the /RESTART_INTERVAL qualifier is present the collection will be restarted periodically. The restart interval is 1 day. Thus, the performance data collection will be restarted daily at 17:00 and daily stopped the next day at 00:00.

```
PerfDat_MGR> SUBMIT COLLECTION DEFAULT/AFTER=05-AUG-2005:17:00
                               /UNTIL=06-AUG-2005:00:00/RESTART=0001-00:00:00

PERFDAT_MGR-I-COLSUC, collection started successfull
PERFDAT_MGR-I-COLSTART, collection profile DEFAULT scheduled for start at  5-AUG-2005
17:01:00.00
```

UNLOAD LICENSE

Deletes a license key from the license table of the PERFDAT configuration database

Format

UNLOAD LICENSE *key*

Parameter

key
Specifies the license key to be deleted from the license table of the PERFDAT configuration database.

Description

Deletes the license key specified by the *key* parameter from the license table of the PERFDAT configuration database. The license keys stored in the license table and their status can be checked by the CHECK LICENSE command.

Example

This example shows how to unload a temporary license key.

```
PerfDat_MGR> CHECK LICENSE

PERFDAT_MGR-I-LICVALID, valid single node license key found
                    /2-01-40-91-179-8F-FD-1F-E1-C6-D8-DB-C6-72-10-37-29-37-033-037/
PERFDAT_MGR-I-LICSNMP, node is licensed to monitor 64 concurrent SNMP agents
PERFDAT_MGR-I-LICIP, node is licensed to monitor 64 concurrent IP agents

PerfDat_MGR> UNLOAD LICENSE
                    2-09-5A-A5-41C-23-2B-1F-A5-A8-D8-E7-DB-A8-10-57-63-37

PERFDAT_MGR-I-UNLOADSUCC, license
                /2-09-5A-A5-41C-23-2B-1F-A5-A8-D8-E7-DB-A8-10-57-63-37/ unloaded
```