R&S® HUMS
Health & Utilization Monitoring Service
User Manual

1179351502
Version 03

ROHDE&SCHWARZ
Make ideas real

This document describes the R&S® optional health and utilization monitoring service application available for selected instruments from the following product groups:

- Rohde & Schwarz spectrum analyzers
- Rohde & Schwarz network analyzers
- Rohde & Schwarz signal generators
- Rohde & Schwarz radio communication testers

All supported instruments are listed in a separate chapter of this manual.

Muehldorfstr. 15, 81671 Muenchen, Germany
Phone: +49 89 41 29 - 0
Email: info@rohde-schwarz.com
Internet: www.rohde-schwarz.com

1179.3515.02 | Version 03 | R&S® HUMS

Throughout this manual, products from Rohde & Schwarz are indicated without the ® symbol, e.g. R&S ® HUMS is indicated as R&S HUMS.

Contents

1 Welcome to R&S HUMS

In this day and age of internet of things, more and more devices are connected to the local network, making it even more difficult for the IT department to monitor them. Rohde & Schwarz instruments are also increasingly accessed via their LAN interface, which provides additional convenience features such as remote desktop, SMB file transfer or a web interface. To make it easier to monitor the use and status of instruments, some devices offer a software option, the Health and Utilization Monitoring Service (HUMS).

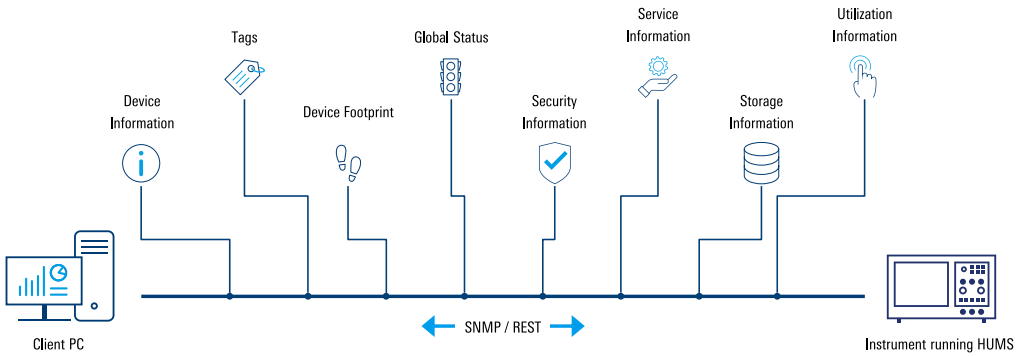
The R&S Health and Utilization Monitoring Service (HUMS) which is available for several R&S products is a software option providing information about a certain instrument such as use and status. Aim is to increase the overall utilization, to avoid downtime and to increase the overall security level of a fleet of instruments.

R&S HUMS provides, for example, information about:

- Instrument identification, hardware components, software packages, licenses
- Usage of remote control, usage via keyboard / mouse, usage of test applications
- Hardware utilization and status, including S.M.A.R.T. data of the system drive
- User-defined static information, for example, an inventory code.

The R&S HUMS software runs as a service in the background on the instrument. It communicates with the instrument's operating system and firmware. For control, R&S HUMS can be accessed via different interfaces within the LAN such as web interface, REST, SNMP or SCPI. R&S HUMS opens the corresponding interface depending on your selection to read the monitoring and utilization data.

This user manual describes how to access and use R&S HUMS.



2 Documentation overview

This section provides an overview of the R&S HUMS user documentation. The documents are available online on the internet or on Gloris.

HUMS user manual

Contains information about the features of the health utilization and monitoring service (HUMS). This includes features available for all products that support HUMS as well as features that are available for certain products only.

The user manual is available for download or immediate display on the internet.

HUMS application note

Contains information about the SNMP and REST API, including a list of all SNMP variables and REST endpoints.

The HUMS application note is available on the internet.

Product user manuals

Contain information about HUMS featuers integrated into the product's firmware or software.

Product user manuals are available on the internet or Gloris, depending on the product.

3 Preparation

3.1 Requirements

R&S HUMS can be accessed via different interfaces such as web interface, SNMP, REST or via SCPI commands. Using the web interface requires a browser to be installed on your PC.

Use one of the following web browsers:

- Microsoft Edge (from version 79)
- Google Chrome (from version 63)
- Mozilla Firefox (from version 66)
- Opera (from version 64).

SNMP-specific requirements:

To read out the data, a third party monitoring software running in the background is required. You can download one of the following software (recommendation):

- Open Source:
  - Net-SNMP
  - Icinga 2
  - Nagios Core
  - Pandora FMS
- Purchasable:
  - SolarWinds Network Performance Monitor
  - ManageEngine OpManager
  - Paessler PRTG Network Monitor

3.2 Installation

R&S HUMS is already part of the firmware. Separate installation is not required.

4 Operation

R&S HUMS supports various application programming interfaces (APIs) from which device information can be retrieved such as identification, BIOS, date, time, system, health and utilization data and software information.

You can access R&S HUMS via the following interfaces and protocols such as:

- Web interface
- SNMP
- REST (HTTP)
- SCPI.

4.1 Access via web interface

1. Open your internet browser.
   For supported browsers, see Chapter 3, "Preparation", on page 7.
2. Enter the IP address or device name of your instrument.
   For information about the IP address or device name, see settings of your instrument.

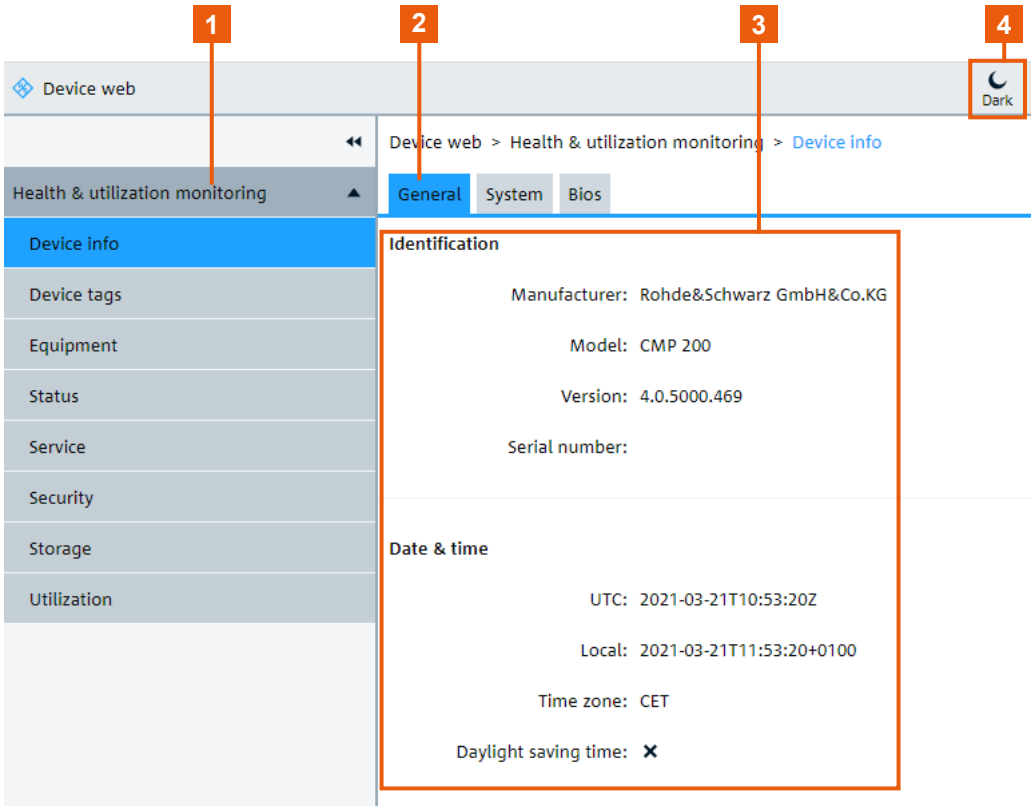
   The "Device Web" interface opens.
3. Select "Health and Utilization Monitoring" from the menu.

   The web interface of R&S HUMS opens.

Graphical user interface (GUI)

The following figure gives an overview of the graphical user interface of R&S HUMS.

Depending on your instrument, you can also access the R&S HUMS web interface locally. For more information, see the main user manual of your instrument.



1 = Menu including categories of the usage data (expanded)
2 = Subcategories
3 = Details of selected usage data
4 = Display mode

4.2 Access via SNMP

SNMP (Simple Network Management Protocol) is a protocol which provides its users a set of operations that allows many instruments to be managed remotely.

SNMP queries and manipulates data in the form of variables arranged in a tree structure. Each variable has an address that describes where it is located in the tree.This address is called object identifier (OID). An OID is described with a sequence of numbers separated by a dot, e.g. "**1.3.6.1.4.1.2566.125.2.1.2.6.2.1**". The length of the row represents how deep you are in the tree and each number represents the index in the respective node.

An OID has a similar format to an IP address. As with IP addresses, there is a name resolution. This is not done by a DNS (= Domain Name System) but by a so-called **Management Information Base (MIB)**. In this case, the necessary R&S HUMS name resolutions in the form of MIB files are provided by the device and can be loaded as an archive.

Before accessing R&S HUMS via SNMP interface, check the SNMP-specific requirements Chapter 3, "Preparation", on page 7.

Downloading SNMP MIB files

1. Open your internet browser.
   For supported browsers, see Chapter 3, "Preparation", on page 7.
2. Enter the R&S HUMS specific address
   http://<DEVICE-ADDRESS>/api/hums/v1/documents?name=snmp
   to load the MIB files.
   Replace the *<DEVICE-ADDRESS>* part with the IP address or device name of your instrument.
   For information about the device name, check the settings of your instrument.
3. Press "Enter".

   The MIB files are automatically downloaded onto your device.
4. Now you can import the downloaded MIB files (ZIP file) into the third-party monitoring software you installed before.

   Depending on your software, it displays the tree structure and the OIDs or nodes.
5. You can start to query and evaluate specific data points.

Example (manual query):

Here is an example to query data manually. In this case, the tool snmpwalk is used.

1. Open a command line.
2. Enter the R&S HUMS specific command
   `-c <SNMP_COMMUNITY> -m +<MIB> <DEVICE-ADDRESS> <OID>`
   including the MIB and OID you want to retrieve data from.
   Replace the `<SNMP_COMMUNITY>` part with the appropriate SNMP community and the `<DEVICE-ADDRESS>` part with the IP address or device name of your instrument.
   Replace the MIB and OID with the appropriate nodes. For all available SNMP-specific MIBs and OIDs, see Application Note Instrument Health & Utilization Monitoring.
   Example command line:
   `snmpwalk -v2c -c public 192.168.0.1 .1.3.6.1.4.1.2566.125.2`
   -v2c = SNMPv2
   public = SNMPv2 community
   192.168.0.1 = Device address
   .1.3.6.1.4.1.2566.125.1 = numeric OID
3. Press "Enter".

   The requested data is returned.

4.3 Access via REST (HTTP)

Representational State Transfer (REST) is a convention on how to design an HTTP interface that accesses resources. In this case, a REST interface can be used as an alternative to SNMP to read or write variable data.

How to get REST API specification

1. Open your internet browser.
   For supported browsers, see Chapter 3, "Preparation", on page 7.
2. Enter the R&S HUMS specific address
   http://<DEVICE-ADDRESS>/api/hums/v1/documents?name=rest.
   Replace the *<DEVICE-ADDRESS>* part with the IP address or device name of your instrument.
   For information about the IP address or device name, check the settings of your instrument.
3. Press "Enter".

   The API specifications open.
4. Now you can start to query and evaluate specific data points.

For a more visual presentation of the retrieved data, alternatively you can use REST API tools such as Postman or Swagger editor.

Depending on your tool, enter the following command including the specific REST node, e.g. *http://<DEVICE-ADDRESS>/api/hums/v1/<RESTNODE>*.

4.4 Access via SCPI commands

If performance is an issue, we recommend using SNMP or REST instead of SCPI. Using SCPI for HUMS analysis can slow down the instrument performance, because it communicates with the firmware. SNMP or REST communicate directly with the HUMS service and therefore have no effect on the firmware itself.

Standard Commands for Programmable Instruments (SCPI) is an instruction set to control instruments and another method to access R&S HUMS. Commands are program messages that a controller (e.g. a PC) sends to the instrument or software. They operate its functions ('setting commands' or 'events') and request information ('query commands'). Some commands can only be used in one way, others work in two ways (setting and query). If not indicated otherwise, the commands can be used for settings and queries.

The syntax of a SCPI command consists of a header and, usually, one or more parameters. To use a command as a query, you have to append a question mark after the last header element, even if the command contains a parameter.

A header contains one or more keywords, separated by a colon. Header and parameters are separated by a "white space" (ASCII code 0 to 9, 11 to 32 decimal, e.g. blank). If there is more than one parameter for a command, they are separated by a comma from one another.

Only the most important characteristics that you need to know when working with SCPI commands are described here. For a complete description, refer to the main user manual of your instrument.

5 General monitoring parameters

The R&S HUMS application provides monitoring parameters available on all instrument type that support the application.

In addition to these general monitoring parameters, there are also monitoring parameters that are specific to certain instrument types. For more information about these, see Chapter 6, "Instrument specific monitoring parameters", on page 28.

5.1 Table functions

Several of the dialogs available in the R&S HUMS application contain tables that you can configure. Using these tables is based on the same principle. This section gives you information about the table functions and how to work with them.

Column selector
Most tables contain the icon. Use this icon to select the columns you want to be displayed in the table.

- To display all parameters:
  Select the icon and select "Show all".
- To display specific parameters:
  Select the ones you want to display.

Filter and sorting function
Most tables contain the icon. Use this icon to sort the values inside the table in alphabetical order or vice versa or filter by a specific character or name.

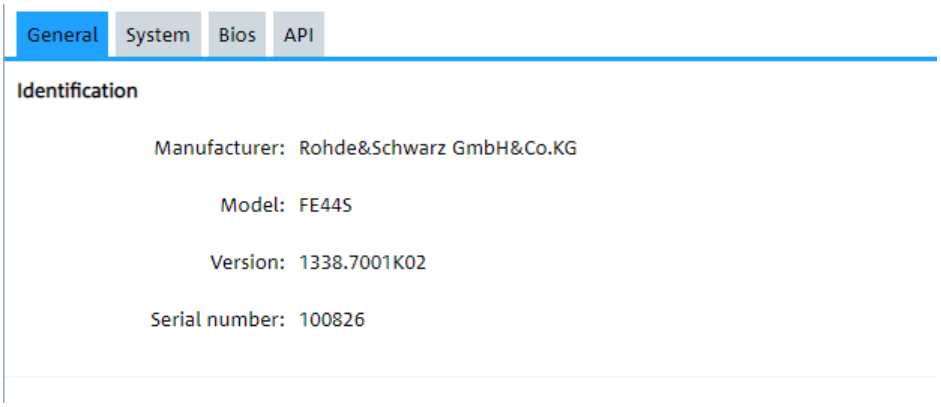
- To activate alphabetical order:
  Select the column you want to sort. Select "Sort 1 to N".
- To activate descending order:
  Select the column you want to sort. Select "Sort N to 1".
- To filter by a specific character or name:
  Select the column you want to filter. In the search bar, type in your preferred character or name.

You can only activate one filter at a time.

5.2 Device information

Access: "Device Info"

The "Device Info" dialog contains general, system-specific and BIOS-related information about the connected instrument.



General
The "General" tab contains general information about the instrument that is monitored.

The "Identification" list contains basic information about the instrument, for example the manufacturer, the instrument model or the serial number.

The "Date and Time" list shows the current time (UTC and local) and the time zone set on the instrument and the state of daylight saving time.

System
The "System" tab contains information about the system components of the instrument that is monitored. This information includes, for example, information about the network configuration (IP address, hostname etc.), the operating system (type, version etc.), memory usage and hardware components (manufacturer, model etc.).

On instruments running on a Linux operating system, the "System" characteristics are only partially evaluated.

Remote command:
`DIAGnostic:HUMS:SYSTem:INFO?` on page 39

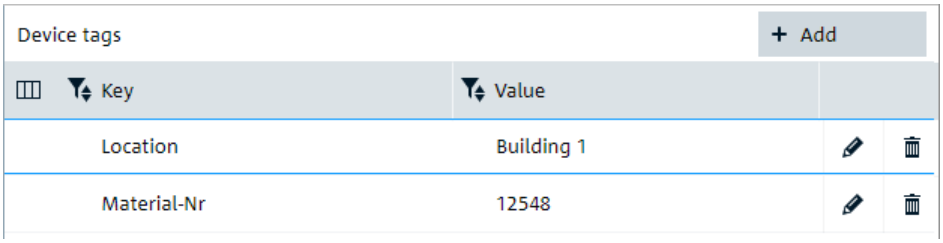
BIOS
The "Bios" tab contains detailed information about the BIOS on the monitored instrument, including manufacturer, version and release date.

Remote command:
`DIAGnostic:HUMS:BIOS?` on page 31

API
The "API" tab contains the SNMP MIB (a `.zip` file) and the REST API (a `.yaml` file) as a download.

5.3 Device tags

Access: "Device Tags"



The "Device Tags" contains functionality to manage device tags.

A device tag is a label to assign to your instrument. You can create any device tag for your instrument and define it by a specific key and value.

Add
Adds a new device tag.

Key
Defines a key for your device tag. A device tag key represents the type of tag.

Value
Defines the actual value of the device tag or key.

Example:
- "Key" = Location
- "Value" = Building 1

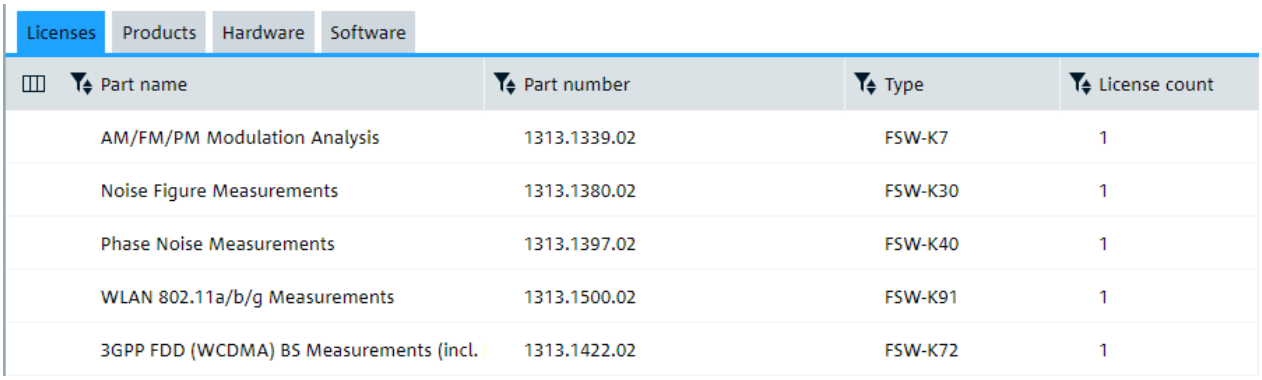
You can also sort the device tags inside the table. See Chapter 5.1, "Table functions", on page 13.

You can use the edit icon to change the tag later or you can use the delete icon to delete the tag.

Remote command:
Add or modify a tag: `DIAGnostic:HUMS:TAGS[:VALue]` on page 41
Return all tags: `DIAGnostic:HUMS:TAGS:ALL?` on page 41
Delete a tag: `DIAGnostic:HUMS:TAGS:DELete` on page 42

5.4 Equipment

Access: "Equipment"



The "Equipment" dialog contains information about hardware and software configuration of your instrument.

This information can be useful when problems occur with the instrument and you require support from Rohde & Schwarz.

Note that tabs or individual columns in the "Equipment" dialog can be empty, depending on the connected instrument.

Licenses
The "Licenses" tab provides an overview of all installed hardware and software licenses (optional features).

Every listed license is described by its:
- Part name (description of the license)
- Part number (order number of the license)
- (Part) type (name of the license)
- License count (number of installations of a specific license)

You can also sort each column in alphabetical or descending order. See Chapter 5.1, "Table functions", on page 13.

Products
The "Products" tab provides an overview of the components that are a separate product.

Every listed component is described by its:
- Part name (description of the product)
- Part number (order number of the product)
- (Part) type (name of the product)

You can also sort each column in alphabetical or descending order. See Chapter 5.1, "Table functions", on page 13.

Hardware
The "Hardware" tab provides an overview of the installed hardware components on your device.

Every listed component is described by its:
- Name (name of the component)
- Part name (description of the component)
- Part number
- Serial number

- Production date
- Hardware code
- Read code
- Test instruction

You can also sort the columns in alphabetical or descending order, see Chapter 5.1, "Table functions", on page 13.

Software
The "Software" tab provides an overview of the installed software components on your instrument.

Every listed component is described by its:
- Manufacturer (name and description of the software; includes the manufactorer if applicable)
- Version (software version, can be empty if software is part of a package)
- Status
- Vendor name
- Configuration
- Release state

You can also sort each column in alphabetical or descending order. See Chapter 5.1, "Table functions", on page 13.

Remote command:
`DIAGnostic:HUMS:EQUipment?` on page 34

5.5 Event history

Access: "Event History"

The "Event History" dialog shows a list of all logged events that have occurred on the connected instrument. Note that the screenshot below is an example and can show status messages that are not available on your instrument.

There are 6369 events: 2971 176 3222

Severity	Timestamp	Message	Details	Source	Event ID
	2022-04-01T09:43:40Z	Firmware Update	5.10-22.3.22.0 Beta	Device	6331
	2022-04-01T09:43:40Z	Firmware Startup		Device	6330
	2022-04-01T09:35:50Z	Firmware Shutdown		Device	6329
	2022-03-31T09:40:11Z	Deviation from Self Alignment Temperature	Deviations detected	Device	6327
	2022-04-01T09:43:40Z	UNCAL	No correction values are available	Device	6332

The type of evaluated events depends on the instrument type HUMS is running on.

For each event, the table contains various additional information, like a a description. Note, however, that the level of detail of the additional information depends on the type of event and the connected instrument type.

Severity
Shows the severity of the event.

- Information
  Event that is not critical and is logged for informational purposes.
- Warning
  Event that can have a negative effect on the instrument performance.
- Error
  Event that will have a negative effect on the instrument performance if it is not resolved.

Remote command:
Query event list: `DIAGnostic:HUMS:DEVice:HISTory?` on page 32
Clear event list: `DIAGnostic:HUMS:DEVice:HISTory:DELete:ALL` on page 33

Timestamp
Shows the time and date when the event has occured.

Message
Shows a short description of the event.

The message usually appears in the user interface of the connected instrument.

Details
Shows a more comprehensive description of the event, for example which firmware version was installed.

Source
Shows the source of the event.

- Device
  Event was logged automatically by the instrument.
- Custom
  Event was added manually. Adding events manually is possible with a SCPI command.

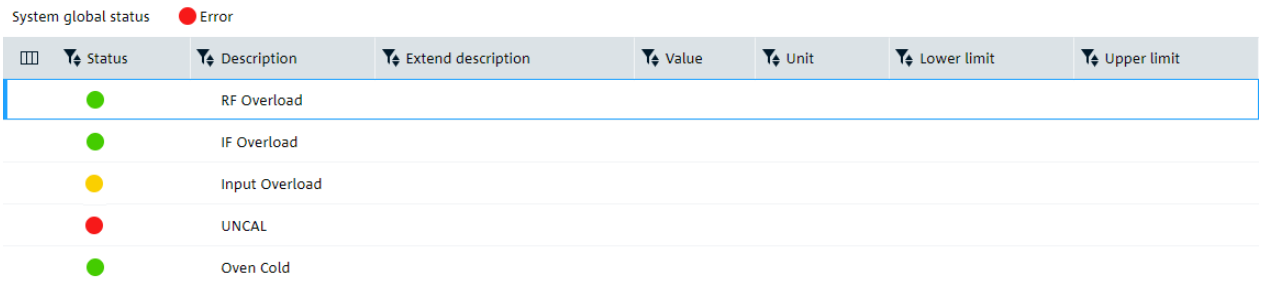
Remote command:
Add an event: `DIAGnostic:HUMS:DEVice:HISTory:EVENt:ADD` on page 33

Event ID
Shows the ID of the event. The ID is a unique identifier for each event in consecutive order.

5.6 Status

Access: "Status"

The "Status" dialog displays the status messages of the connected instrument and its applications or properties. Note that the screenshot below is an example and can show status messages that are not available on your instrument.



The number and type of evaluated status information depends on the instrument type HUMS is running on. Depending on the instrument type, it is also possible to use utilizations and see how often a certain status message has occurred.

The "System global status" represents the overall status of all individual status from the table. This means if only one status turns yellow or red, the overall system status turns yellow or red as well.

For each status message, the table contains various additional information, like a description or a value. Note, however, that this additional information is not necessarily displayed. Which information is displayed depends on the status message and the connected instrument type.

Status
The status is indicated in varying colors.

- Green — The system is OK.
- Yellow — There are some issues with the system which might be critical later.
- Red — Problems with the system occurred and the status is critical.

Remote command:
`DIAGnostic:HUMS:SYSTem:STATus?` on page 39
`DIAGnostic:HUMS:SYSTem:STATus:SUMMary?` on page 40

Description
Description of the instrument status that is evaluated.

For a comprehensive description of the instrument status and its effects, refer to the user manual of the instrument that is monitored.

Extended description
Additional information about the instrument status.

Value
Some instrument status depend on a certain value, for example a temperature.

The "Value" column shows the current value for the evaluated status, for example the current temperature.

Unit
Some instrument status depend on a certain value with a unit, for example a temperature.

The "Unit" column shows the unit that the evaluated status is measured in, for example "degree Celsius" for the temperature status.

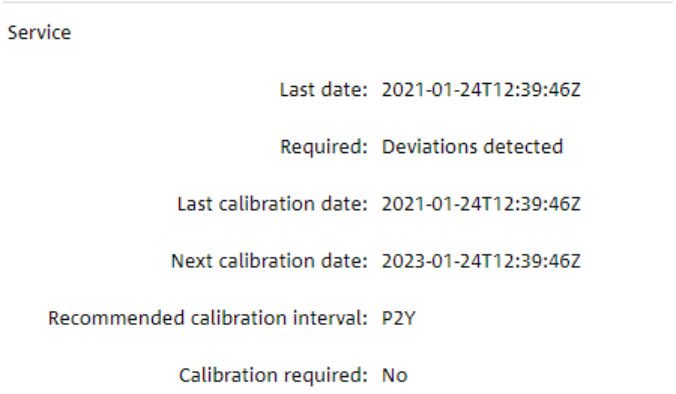
Lower and upper limit
Some instrument status depend on a certain value, for example a temperature.

The "Lower Limit" and "Upper Limit" columns indicate the limits that cause an error when they are violated.

For example, if the limits for the instrument temperature are 10 °C and 30 °C, the instrument status is OK as long as the measured temperature is between those limits.

5.7 Service

Access: "Service"



The "Service" dialog contains information about service activities.

Service information
Shows miscellaneous service information. The displayed information is usually entered via the firmware of the connected instrument.

This dialog provides service data such as:

- Last date of service
- Required service
- Last/next calibration date
- Recommended calibration date
- Recommended calibration interval in ISO801 duration format, e.g. "P2Y" = period of two years
- Required calibration

We recommend not changing the system time. Repeatedly changing the system time can mix up planned calibration dates. Rather configure the time server of the operating system properly for your organization.

Remote command:
DIAGnostic:HUMS:SERVice? on page 36

5.8 Security

Access: "Security"

The "Security" dialog contains an overview of IT security-related information about your connected instrument and hotfixes.

On instruments running on a Linux operating system, the "Security" characteristics only partially evaluated.

Security Overview
The "Overview" tab contains information about IT security featires on the monitored instrument. This information includes status of anti-malware software (name, state and status) and the firewall status.

Hotfixes
The "Hotfixes" tab lists all Windows-specific security updates installed on your instrument by their Microsoft ID and installation date.

Remote command:
DIAGnostic:HUMS:SECurity? on page 35

5.9 Storage

Access: "Storage"

Disks | S.M.A.R.T. | Installed packages

	/dev/sda
Model	SanDisk SD9SB8W128G
Serial number	182004801161
Device ID	5 001b44 8b67d0a7d
Firmware version	X6107000
Capacity	128035676160
Sector size	512
Rotation rate	0
Form factor	2.5 inches

The "Storage" dialog lists the properties of the built-in hard disks such as information about the disks, S.M.A.R.T (Self-Monitoring, Analysis and Reporting Technology) or installed software packages.

Disks
This tab lists all disk-related information such as hard disk model, serial number, device ID or firmware version, depending on the detected storage devices.

S.M.A.R.T.
S.M.A.R.T. is a common technology that analyzes and outputs the state of HDDs and SSDs. It is supported by most hard disk manufacturers. The return takes place via attributes that the manufacturer defines itself.

This S.M.A.R.T tab indicates disk-specific health data such as temperature or free/ used space based on the selected disk. Select the preferred disk in the "Disk Information" field.

Remote command:
`DIAGnostic:HUMS:STORage?` on page 37

Installed Packages
This tab lists the installed software packages on your instrument.

On instruments running on a Linux operating system, the "Installed Packages" are only partially evaluated.

Remote command:
`DIAGnostic:HUMS:SW?` on page 38

5.10 Utilizations

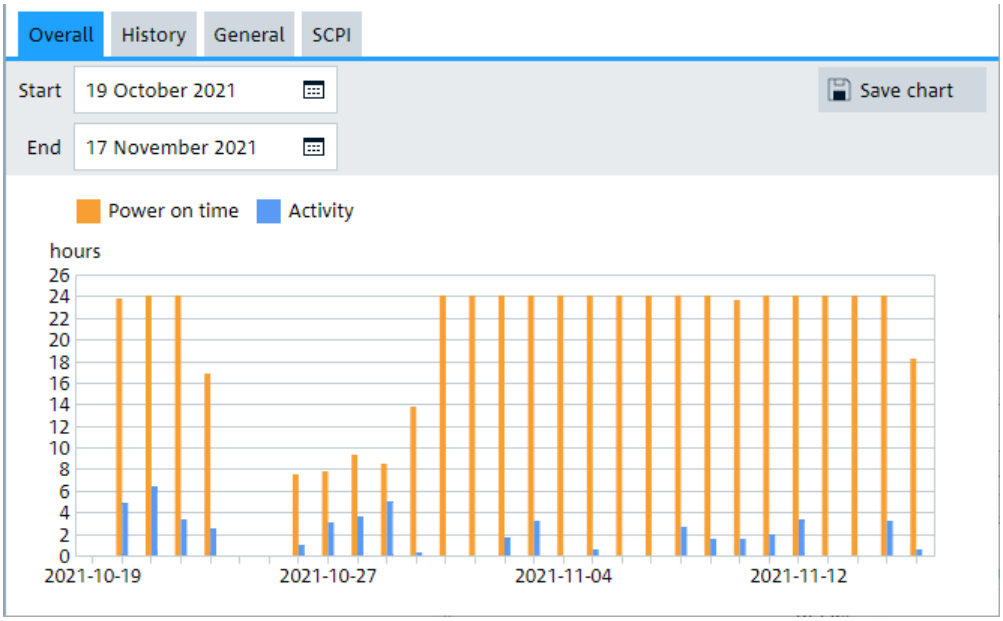
Access: "Utilization"

The "Utilization" dialog contains functionality to monitor instrument usage and define the data that are tracked. This tracking includes tracking of data related to remote control (SCPI) usage.

5.10.1 Utilizations overall

Access: "Utilization" > "Overall"

The "Overall" tab shows the usage data of the connected instrument during a certain period of time in a diagram.



The y-axis represents the usage time in hours. The x-axis represents the time period over which the evaluation has occurred.

- The orange bar indicates the power on time (= amount of time the instrument has been running).
- The blue bar indicates the total activity of the instrument (= amount of time of all tracked activities/utilizations).

The displayed time period depends on your selection.

Start / End
Start and end date of the time period you want to evaluate.

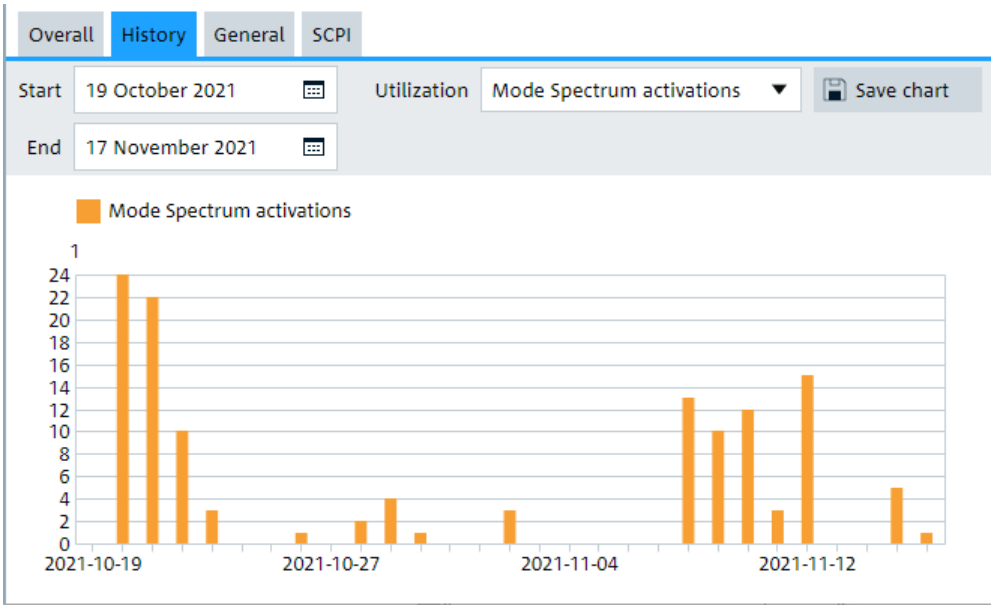
Save chart
Downloads the diagram for later evaluation.

Remote command:
`DIAGnostic:HUMS:SAVE` on page 35

5.10.2 Utilizations history

Access: "Utilization" > "History"

The "History" tab shows the usage data of a certain utilization during a certain period of time in a diagram.



The y-axis represents the usage time in hours. The x-axis represents the time period over which the evaluation has occurred.

- The orange bar indicates the selected utilization, e.g. in this case "Software starts".

Start / End
Start and end date of the time period you want to evaluate.

Displayed activity
Activity or utilization you want to evaluate. The available utilizations depend on the connected instrument.

Remote command:
`DIAGnostic:HUMS:UTILization:HISTory?` on page 44

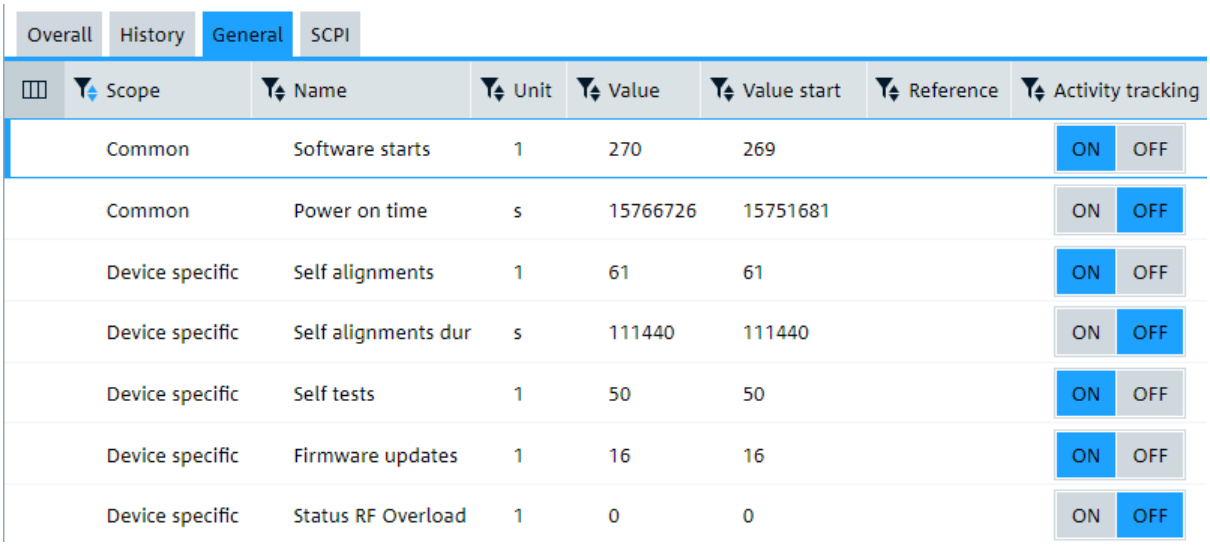
Save chart
Downloads the diagram for later evaluation.

Remote command:
`DIAGnostic:HUMS:SAVE` on page 35

5.10.3 Utilizations general

Access: "Utilization" > "General"

The "General" tab shows all utilizations available on the instrument. Utilizations are the type of data that can be tracked.



Scope	Name	Unit	Value	Value start	Reference	Activity tracking
Common	Software starts	1	270	269		ON OFF
Common	Power on time	s	15766726	15751681		ON OFF
Device specific	Self alignments	1	61	61		ON OFF
Device specific	Self alignments dur	s	111440	111440		ON OFF
Device specific	Self tests	1	50	50		ON OFF
Device specific	Firmware updates	1	16	16		ON OFF
Device specific	Status RF Overload	1	0	0		ON OFF

Some utilizations are generic and available on all instruments while others are specific to certain instrument types, like spectrum analyzers. For a list of available utilizations, see Chapter 10, "Overview of utilizations", on page 54.

The utilization data is collected and evaluated continuously. The cycle can differ depending on the instrument.

Scope
The "Scope" column shows the data category. You can find different scopes inside the table depending on your connected instrument:

Common	Tracking of general instrument usage.
Device specific	Tracking of instrument usage and status.
License	Tracking of application usage and usage of hardware components.
Remote	Tracking of remote control usage.
Local	Tracking of local instrument control.
Custom	Tracking of custom instrument statistics. For more information, see Chapter 5.10.4, "Custom utilizations", on page 26.

Name
The "Name" column shows the name of the utilization and indicates the type of data that is tracked.

For a list of available utilizations and on which instruments they are available, see Chapter 10, "Overview of utilizations", on page 54.

Unit
The "Unit" column shows the unit of the corresponding utilization.

For example, the unit for the "Power on time" is s(econds).

Value
The "Value" column shows the current value of the corresponding utilization.

Example:

"Power on time" = 500 000

Means that the instrument has been running for 500 000 seconds since the last reset of the counter.

Value start
The "Value Start" column shows the initial value for the corresponding utilization when the instrument was turned on.

Reference
The "Reference" column shows additional information about the utilization.

Activity Tracking
The "Activity Tracking" column defines if the utilization contributes to the graph shown in the "Overall" utilization tab.

Turn on any utilizations that you want to contribute to the overall utilization graph. This contribution to the overall utilization graph also works retroactively.

Remote command:
`DIAGnostic:HUMS:UTILization?` on page 42
`DIAGnostic:HUMS:UTILization:ACTivity:TRACking:STATe` on page 43

5.10.4 Custom utilizations

Access: "Utilization" > "General"

Definition of custom utilizations: REST and SCPI only.

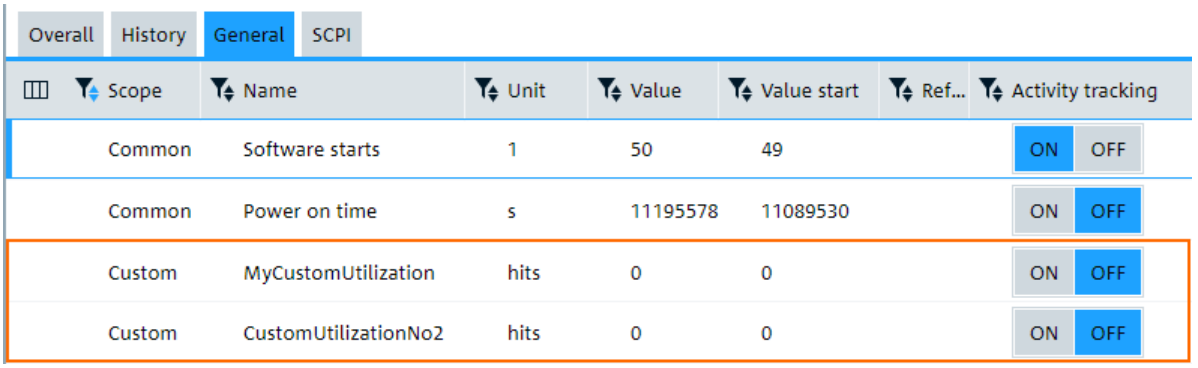
In addition to the predefined generic and instrument specific utilizations, you can define custom utilizations that can track instrument statistics not yet covered by one of the predefined utilizations. Custom utilizations can be managed and controlled using SCPI commands.

The basic workflow for custom utilizations goes like this:

1. Create a custom utilization using a SCPI command.
2. Wait until an event occurs that should trigger a change in the custom utilization.
3. Update the value of the custom utilization using a SCPI command.
4. Wait until the event occurs again.
5. Update the value of the custom utilization using a SCPI command.
6. etc.

In the user interface of the device web, custom utilizations are added to the utilizations list. The category (or scope) for custom utilizations is "Custom".

You can also view the custom utilizations in the utilizations history diagram.



For details about managing and controlling custom utilizations with REST, see the REST API specification. See Chapter 4.3, "Access via REST (HTTP)", on page 11 for details on how to access the specification.

The remote commands required to manage and control custom utilizations are described in Chapter 7.4, "Custom utilizations commands", on page 47.

For more information about the information about the utilizations list in general, see Chapter 5.10.3, "Utilizations general", on page 24.

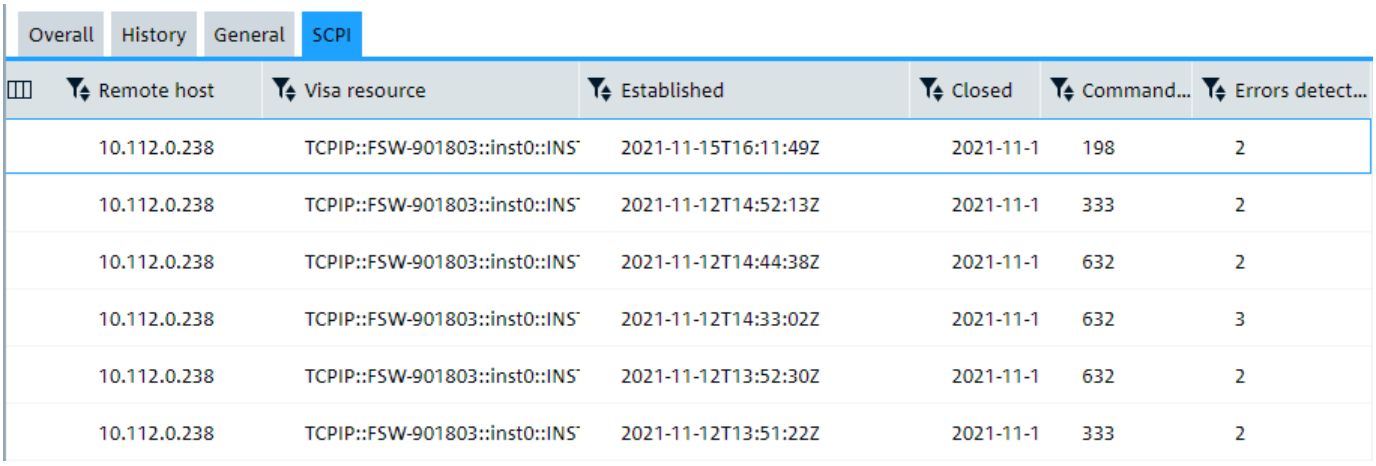
5.10.5 Utilizations SCPI

Access: "Utilization" > "SCPI"

The "ACPI" tab shows all data regarding SCPI connections during a certain period of time.

The SCPI data includes the following information.

- Remote host
- Establish or close date
- How many commands have been executed
- Detected SCPI errors.



6 Instrument specific monitoring parameters

The R&S HUMS application has several instrument specific functions and evaluations. In particular, different instruments support different status monitorings and different utilizations.

The availability of the features descrobed here therefore depends on the instrument that you are monitoring.

Checking the current instrument status

The current instrument status is displayed in the "Status" tab of the R&S HUMS application. While the evaluation parameters are the same for all instruments that support status monitoring, the possible status messages depend on the instrument type.

Useful links:

- For more information about instrument status monitoring, see Chapter 5.6, "Status", on page 18.
- For a detailed list of status information supported by certain instrument groups, see Chapter 9, "Overview of status messages", on page 53.
  For more information about the meaning of status messages, refer to the user manual of the analyzer.

On some instruments, you can also track how often certain instrument status have occured on your instrument using utlizations. Check the list of utilizations for your instrument to see if the instrument supports this feature.

Tracking instrument status over time

In addition to the current instrument status, you can also track how often a certain instrument status has occured and how long it was active. This information is available in the "Utilizations" tab of the HUMS application. If the instrument status is tracked over time depends on the instrument.

Useful links:

- For more information about general status functionality, see Chapter 5.6, "Status", on page 18.
- For a detailed list of instrument status that are tracked by certain instrument groups, see Chapter 9, "Overview of status messages", on page 53.

Instruments that support tracking the instrument status over time track the following information.

- Number of activations:
  Shows how often the corresponding status occurred.
- Active time:
  Shows the amount of time the status has been active.

Example:

Signal and spectrum analyzers track the occurence of "RF Overload" status. For this status the HUMS would track, for example:

- Number of activations = 5, means that an RF overload has occurred five times since the last reset.
- Active time = 350 s, means that the RF overload has was active for 350 s over these five occurrences.

Tracking instrument usage over time

Most instruments that support HUMS track various information about instrument usage over time. This information is available in the "Utilizations" tab of the HUMS application. The kind of data that is tracked over time depends on the instrument.

Useful links:

- For more information about general utilization functionality, see Chapter 5.10, "Utilizations", on page 22.
- For a detailed list of utilizations available for certain instrument groups, see Chapter 10, "Overview of utilizations", on page 54.

Instruments that track the instrument usage over time track the various information.

Typical information that is collected regarding the instrument usage is, for example:

- Number of activations (typical for all instrument types):
  Shows how often the corresponding instrument function was activated.
- Active time (typical for all instrument types):
  Shows the amount of time the corresponding instrument function was active.
- Measurement time (typical for analyzers):
  Shows how long the instrument has been measuring in the corresponding application.

Example:

Example for signal and spectrum analyzers. Signal and spectrum analyzers track the usage of the firmware applications, for example the usage of the spectrum application. You can read the resulting information as, for example:

- Number of activations = 8, means that the spectrum application has been started eight times since the last reset.
- Active time = 1528451 s, means that the spectrum application has been active 1528451 seconds over these eight occurrences.
- Active measurement time = 1528437 s, means that the spectrum application has been measuring for 1528437 seconds.

Note that these numbers can aggregate the statistics for several instrument functions. For example, for spectrum analyzers, the application usage for R&S FSx-K18, -K18D and -K18F is grouped as usage for "Mode Amplifier".

7 Remote control

In addition to the REST API and SNMP, you can also use SCPI commands to control the health and utilization monitoring service and read out the tracked data.

For basic information of network and remote operations and the general structure of SCPI commands, read the appropriate section in the main user manual of your instrument.

For a comprehensive description of using SCPI commands, refer to the following documents:

- General introduction on remote control: Getting started: Remote control via SCPI
- The main user manual available for your instrument online.

7.1 System information commands

DIAGnostic:HUMS[:ALL]?

Queries the REST endpoints for the HUMS application in a single query. This allows you to read all HUMS data stored on the instrument via REST API.

The data can be displayed either in JSON or XML format. For more information about setting the format, see DIAGnostic:HUMS:FORMat on page 34.

Return values:

<Endpoints> <block_data>

String returns REST endpoints as block data in a comma-separated list.
Binary block data with the following syntax:
#<Digits><Length><Binarydata>
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the <Length> value
<Length>
Decimal value
Number of bytes the follow in the <Binary data> part
<Binary data>
Binary data in ASCII format

Example: //Return complete HUMS data
DIAG:HUMS?

Usage: Query only

DIAGnostic:HUMS:BIOS?

Queries the BIOS information from the instrument. Depending on the set data format, the queried data is either displayed in XML or JSON format. For more information about setting the data format, see DIAGnostic:HUMS:FORMat on page 34.

Return values:

<BIOSInfo> <block_data>

BIOS information of the connected instrument as block data in a comma-separated list.
#<manufacturer>,<serialNumber>,<version>,
<releaseDate>,<caption>,<isPresent>
Binary block data with the following syntax:
#<Digits><Length><Binarydata>
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the <Length> value
<Length>
Decimal value
Number of bytes the follow in the <Binary data> part
<Binary data>

Binary data in ASCII format

Example: //Return BIOS information
`DIAG:HUMS:BIOS?`

Usage: Query only

Manual operation: See "BIOS" on page 14

DIAGnostic:HUMS:DELete:ALL

Deletes the complete HUMS data. This includes device history, device tags, SCPI connections, utilization history and utilizations.

Example: //Delete HUMS data
`DIAG:HUMS:DEL:ALL`

Usage: Event

DIAGnostic:HUMS:DEVice:HISTory?

Queries the device history information of the connected instrument. Depending on the set data format, the queried data is either displayed in `XML` or `JSON` format. For more information about setting the data format, see `DIAGnostic:HUMS:FORMat` on page 34.

Return values:

<HistoryInfo> <block_data>

Device history information of the connected instrument as block data in a comma-separated list:
`#blockdata[{event1},{event2},{event3}...]`
With the following parameters:
`<eventID>,<eventTimestamp>,<eventMessage>, <eventDetails>,<eventSeverity>`
Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value
<Length>
Decimal value
Number of bytes the follow in the `<Binary data>` part
<Binary data>
Binary data in ASCII format

Example: //Return device history
```
DIAG:HUMS:DEV:HIST?
```
Returns for example:
```
#44715[{"eventId":32,"eventTimestamp":
"2021-02-02T17:25:39Z","eventMessage":
"Deviation from Self Alignment Temperature",
"eventDetails":
"Deviations resolved","eventSeverity":0}
```

Usage: Query only

Manual operation: See "Severity" on page 18

DIAGnostic:HUMS:DEVice:HISTory:DELete:ALL

Deletes the complete device history information of the connected instrument.

Example: //Delete complete device history
```
DIAG:HUMS:DEV:HIST:DEL:ALL
```

Usage: Event

Manual operation: See "Severity" on page 18

DIAGnostic:HUMS:DEVice:HISTory:EVENt:ADD <Severity>, <Message>[, <Detail>]

Adds an event to the event history.

Note that the event ID, time stamp and event source are automatically created when you add the event.

Setting parameters:

<Severity> INFO | WARNing | ERRor
Severity of the event.

<Message> Short description of the event.

<Detail> More comprehensive description of the event.

Example: //Add an event ot the event history
```
DIAG:HUMS:DEV:HIST:EVEN:ADD
INFO,"InfoEvent","An event that has occurred on
the instrument"
```

Usage: Setting only

Manual operation: See "Source" on page 18

DIAGnostic:HUMS:EQUipment?

Queries the equipment information (device footprint) of the instrument. Depending on the set data format, the queried data is either displayed in XML or JSON format. For more information about setting the data format, see DIAGnostic:HUMS:FORMat on page 34.

Return values:

<EquipmentInfo> <block_data>

Device footprint of the connected instrument as block data in a comma-separated list.
Binary block data with the following syntax:
#<Digits><Length><Binarydata>
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the <Length> value
<Length>
Decimal value
Number of bytes the follow in the <Binary data> part
<Binary data>
Binary data in ASCII format

Example: //Return device footprint
DIAG:HUMS:EQU?

Usage: Query only

Manual operation: See "Software" on page 17

DIAGnostic:HUMS:FORMat <DataFormat>

Selects the format for the queried HUMS data. You can query the HUMS data either in JSON format or XML format.

The defined format affects all other commands that return block data.

Parameters:

<DataFormat> JSON | XML

JSON
Returns the HUMS data in JSON format.

XML
Returns the HUMS data in XML format.

*RST: JSON

Example: //Return data in JSON format
DIAG:HUMS:FORM JSON

DIAGnostic:HUMS:SAVE <path>

Saves the HUMS history as a `ZIP` file to your preferred path.

Setting parameters:
<path>

Example: //Save HUMS history data
```
DIAG:HUMS:SAVE 'C:\HUMS\hums_2021.zip'
```

Usage: Setting only

Manual operation: See "Save chart" on page 23
See "Save chart" on page 24

DIAGnostic:HUMS:SECurity?

Queries the security information of the instrument. Depending on the set data format, the queried data is either displayed in `XML` or `JSON` format. For more information about setting the data format, see `DIAGnostic:HUMS:FORMat` on page 34.

Return values:

<SecurityInfo> <block_data>

Security information of the connected instrument as block data in a comma-separated list.
```
#blockdata{"antimalware":
{<name>,<enabled>,<upToDate>,<timestamp>},
"firewallEnabled"}
```
Binary block data with the following syntax:
```
#<Digits><Length><Binarydata>
```
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value
<Length>
Decimal value
Number of bytes the follow in the `<Binary data>` part
<Binary data>
Binary data in ASCII format

Example: //Return security information
```
DIAG:HUMS:SEC?
#3133{"antimalware":{"name":
"Windows Defender","enabled":false,"upToDate":
true,"timestamp":
"2018-02-08T10:09:22Z"},"firewallEnabled":true}
```

Usage: Query only

Manual operation: See "Hotfixes" on page 21

DIAGnostic:HUMS:SERVice?

Queries the service information of the instrument. Depending on the set data format, the queried data is either displayed in `XML` or `JSON` format. For more information about setting the data format, see `DIAGnostic:HUMS:FORMat` on page 34.

Return values:

<ServiceInfo> <block_data>

Service information of the connected instrument as block data in a comma-separated list:
`#blockdata{<lastdate>,<requiredservice>,<calibration>:{<lastcalibration>,<nextDue>,<nextDueExpired>,<recommendedCalibrationInterval>},<uptime>}`
Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value
<Length>
Decimal value
Number of bytes the follow in the `<Binary data>` part
<Binary data>
Binary data in ASCII format

Example:

```
//Return service information
DIAG:HUMS:SERV?
Returns for example:
#3196{"last":"2021-01-19T23:00:00Z","required":2,"calibration":{"last":"2021-01-19T23:00:00Z","nextDue":"2022-01-19T23:00:00Z","nextDueExpired":false,"recommendedCalibrationInterval":"P1Y"},"uptime":0}
```

Usage: Query only

Manual operation: See "Service information" on page 20

DIAGnostic:HUMS:STATe <State>

Turns the HUMS service and data collection on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: ON

Example:

```
//Turn on HUMS service
DIAG:HUMS:STAT ON
```

DIAGnostic:HUMS:STORage?

Queries the storage information of the instrument. Depending on the set data format, the queried data is either displayed in `XML` or `JSON` format. For more information about setting the data format, see `DIAGnostic:HUMS:FORMat` on page 34.

Return values:

<StorageInfo> <block_data>

Returns storage information for each storage device of the connected instrument as block data.
`#blockdata[{storagedevice1},{storagedevice2},...]`
Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value
<Length>
Decimal value
Number of bytes the follow in the `<Binary data>` part
<Binary data>
Binary data in ASCII format
For each storage device it returns:
`<diskname>,<temperature>,<type>,<modelfamily>,<model>,<serialnumber>,<id>,<firmwareversion>,<capacity>,<sectorsize>,<rotationrate>,<formfactor>,<ataversion>,<sataversion>,<smartdatatable>.`
For each value of the smart data table it returns:
`<id>,<attributename>,<dataflag>,<value>,<worst>,<threshold>,<type>,<update>,<whenfailed>,<rawvalue>`

Example: //Return storage information
DIAG:HUMS:STOR?
Returns for example:
#44620[{"name":"/dev/sda","temperature":
305,"type":"ata","modelFamily":null,"model":
"SanDisk SD9SB8W128G","serialNumber":
"000000000000","id":
"5 001b44 8b67d0a7d","firmwareVersion":
"A1234000","userCapacity":
128035676160,"sectorSizeLogical":
512,"sectorSizePhysical":512,"rotationRate":
"0","formFactor":"2.5 inches","ataVersion":
"ACS-4 T13/BSR INCITS 529 revision 5","sataVersion":
"SATA 3.3","smartDataTable":[{"id":
5,"attributeName":
"Reallocated_Sector_Ct","dataFlag":50,"value":
100,"worst":100,"threshold":0,"type":
"Old_age","updated":
"Always","whenFailed":null,"rawValue":0},{"id":
9,"attributeName":"Power_On_Hours","dataFlag":
50,"value":100,"worst":100,"threshold":
0,"type":"Old_age","updated":
"Always","whenFailed":null,"rawValue":16856}]

Usage: Query only

Manual operation: See "S.M.A.R.T." on page 22

DIAGnostic:HUMS:SW?

Queries information about the installed software on the instrument. Depending on the set data format, the queried data is either displayed in XML or JSON format. For more information about setting the data format, see DIAGnostic:HUMS:FORMat on page 34.

Return values:

<SoftwareInfo> <block_data>

For each installed software package it returns the following information as block data in a comma-separated list:
#blockdata[{<index>,<name>,<softwareType>,
<installDate>}]
Binary block data with the following syntax:
#<Digits><Length><Binarydata>
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the <Length> value
<Length>
Decimal value

Number of bytes the follow in the <Binary data> part
<Binary data>
Binary data in ASCII format

Example: //Return installed software information
DIAG:HUMS:SW?

Usage: Query only

Manual operation: See "Installed Packages" on page 22

DIAGnostic:HUMS:SYSTem:INFO?

Queries the system information of the instrument. Depending on the set data format, the queried data is either displayed in XML or JSON format. For more information about setting the data format, see DIAGnostic:HUMS:FORMat on page 34.

Return values:

<SystemInfo> <block_data>
Returns system information of the connected instrument as block data.
Binary block data with the following syntax:
#<Digits><Length><Binarydata>
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the <Length> value
<Length>
Decimal value
Number of bytes the follow in the <Binary data> part
<Binary data>
Binary data in ASCII format

Example: //Return system information
DIAG:HUMS:SYST:INFO?

Usage: Query only

Manual operation: See "System" on page 14

DIAGnostic:HUMS:SYSTem:STATus?

Queries the complete system status information of the instrument. Depending on the set data format, the queried data is either displayed in XML or JSON format. For more information about setting the data format, see DIAGnostic:HUMS:FORMat on page 34.

Return values:

<SystemStatus> <block_data>

Returns complete system status information of the connected instrument as block data in a comma-separated list:
`#blockdata{<globalStatus>,<tablevalues>:[{tablevalue1},{tablevalue2},...]}`
Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value
<Length>
Decimal value
Number of bytes the follow in the `<Binary data>` part
<Binary data>
Binary data in ASCII format
For each table value it returns the following values:
`<id>,<description>,<descriptionExtended>,<value>,<unit>,<upperLimit>,<lowerLimit>,<reference>,<severity>`

Example: //Return system status information

```
DIAG:HUMS:SYST:STAT?
#41874{"globalStatus":3,"values":[{"id":
31522816,"description":
"RF Overload","descriptionExtended":
null,"value":null,"unit":null,"upperLimit":
null,"lowerLimit":null,"reference":
null,"severity":1},...}]}
```

Usage: Query only

Manual operation: See "Status" on page 19

DIAGnostic:HUMS:SYSTem:STATus:SUMMary?

Queries the status summary of the complete system.

Return values:

<StatusSummary>

OK
The system is OK.

WARNing
There are some issues with the system which might be critical later.

ERRor
Problems with the system occurred and the status is critical.

Example: //Return status summery of the system

```
DIAG:HUMS:SYST:STAT:SUMM?
```

Usage: Query only

Manual operation: See "Status" on page 19

7.2 Device tags commands

DIAGnostic:HUMS:TAGS:ALL?

Queries all key-value tags that you have assigend to the instrument. Depending on the set data format, the queried data is either displayed in `XML` or `JSON` format. For more information about setting the data format, see `DIAGnostic:HUMS:FORMat` on page 34.

Return values:

<ID> ID number of the defined tag.

<Key> String containing key name of the defined tag.

<Value> String containing value of the defined tag.

Example:
```
//Return all tags
DIAG:HUMS:TAGS:ALL?
1,"location","building_11",2,"time zone","CET"
```

Usage: Query only

Manual operation: See "Value" on page 15

DIAGnostic:HUMS:TAGS[:VALue] <ID>, <Key>, <Value>
DIAGnostic:HUMS:TAGS[:VALue]? <ID>

Adds or modifies a key-value pair (device tag).

The query returns the key-value pair for a given ID or an empty string if the ID is unknown.

Parameters:

<Key> String containing key name of the queried tag.

<Value> String containing value of the queried tag.

Parameters for setting and query:

<ID> 0 - 31
ID number of the tag you want to modify or query.
To identify the ID number, query all device tags from the system first. For more information, read here `DIAGnostic:HUMS:TAGS:ALL?` on page 41.

Example:
```
//Add or modify a tag (tag 1)
DIAG:HUMS:TAGS 1,'location','building_11'
```

Manual operation: See "Value" on page 15

DIAGnostic:HUMS:TAGS:DELete <ID>

Deletes a certain tag you assigned to your instrument, including its key and value.

Setting parameters:

<ID> ID number of the tag you want to delete.
To identify the ID number, query all device tags from the system first. For more information, see `DIAGnostic:HUMS:TAGS:ALL?` on page 41.

Example: //Delete tag
`DIAG:HUMS:TAGS:DEL 0`

Usage: Setting only

Manual operation: See "Value" on page 15

DIAGnostic:HUMS:TAGS:DELete:ALL

Deletes all key-value tags you have assigned to the instrument.

Example: //Delete all tags
`DIAG:HUMS:TAGS:DEL:ALL`

Usage: Event

7.3 Utilization commands

DIAGnostic:HUMS:UTILization?

Queries the current utilization data of the instrument.

Depending on the set data format, the queried utilization data is either displayed in `XML` or `JSON` format. For more information about setting the data format, see `DIAGnostic:HUMS:FORMat` on page 34.

Return values:

<UtilizationData> <block_data>
Returns the current utilization data of the connected instrument as block data.
`#blockdata[{utilization1},{utilization2},...]`

Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value
<Length>
Decimal value
Number of bytes the follow in the `<Binary data>` part
<Binary data>
Binary data in ASCII format
For each utilization it returns the following values:
`<id>,<scope>,<reference>,<unit>,<description>,<value>,<startvalue>,<state>`

Example: //Return utilization data
`DIAG:HUMS:UTIL?`
Returns for example:
```
#123456[{"id":31719936,"scope":
"DEVICE_SPECIFIC","reference":null,"name":
"Self alignments","unit":
"counter","description":
"Report number of self alignments","value":
0,"startupValue":
0,"activityTracking":true},...]
```

Usage: Query only

Manual operation: See "Activity Tracking" on page 26

DIAGnostic:HUMS:UTILization:ACTivity:TRACking:STATe <ID>, <State>
DIAGnostic:HUMS:UTILization:ACTivity:TRACking:STATe? <ID>

Turns the activity tracking of utilizations on and off.

Parameters:
<State> ON | OFF | 1 | 0

ON | 1
Tracking activity = on

OFF | 0
Tracking activity = off

Parameters for setting and query:
<ID> ID number of the utilization activity.
To find out the ID of the utilization, you have to query the complete utilization data. For more information, see `DIAGnostic:HUMS:UTILization?` on page 42.

Example: //Set activity tracking state of ID 31719936
`DIAG:HUMS:UTIL:ACT:TRAC:STAT 31719936,'1'`

Manual operation: See "Activity Tracking" on page 26

DIAGnostic:HUMS:UTILization:HISTory? [<StartDate>[, <EndDate>[, <TimeResolution>]]]

Queries the history of utilization activity on the connected instrument.

The data format is either `XML` or `JSON`, depending on `DIAGnostic:HUMS:FORMat`.

Timestamps follow the Unix time format.

Query parameters:

<StartDate> String containing the start date of the evaluation period (ISO8601 format).
If you do not define a start date, the command returns data starting with the data acquisition one month before the end date.

<EndDate> String containing the end date of the evaluation period (ISO8601 format).
If you do not define a end date, the command returns data for a month starting on the start date.

<Granularity> Resolution of the evaluation period (default = usage over a period of 1 day or 86 400 seconds).
You can define the resolution directly in terms of seconds as a number or indirectly in terms of SEC, MIN, HOUR or DAY (for example `7DAY` results in a resolution of one week). The base unit is seconds.

Default unit: s

Return values:

<History> <block_data>

Utilization usage of the connected instrument during the time period you have defined. The number of return values depends on the selected time period and time resolution. For example, if you evaluate over a time period of one week with a resolution of one day, the command returns 7 values.
Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block.
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value.
<Length>
Decimal value
Number of bytes that follow in the `<Binary data>` part.
<Binary data>
Binary data in ASCII format

The binary data represents the utilization activity. Each utilization is assigned a timestamp for when the utilization has been active. Together they form a pair of values.
`#blockdata{"timestamps":[<timestamp1>,<timestamp2>,...],"activity":[<activity1>,<activity2>,...]}`
Each timestamp therefore has a corresponding activity:
`<timestamp1> + <activity1>, <timestamp2> + <activity2>`

Example: //Query utilization history for all utilizations
`DIAG:HUMS:UTIL:HIST?`
Returns for example:
`#3440{"timestamps":[1612259226,1612345626,1612432026,1612518426],"activity":[0,0,66000,81000]}`

Usage: Query only

Manual operation: See "Displayed activity" on page 24

DIAGnostic:HUMS:UTILization:HISTory:DELete:ALL

Deletes the complete utilization history information from the instrument.

Alternatively, you can delete:

- device history only with `DIAGnostic:HUMS:DEVice:HISTory:DELete:ALL`
- all data with `DIAGnostic:HUMS:DELete:ALL`

Example: //Delete complete utilization history
`DIAG:HUMS:UTIL:HIST:DEL:ALL`

Usage: Event

DIAGnostic:HUMS:UTILization:HISTory:DETailed? <ID>[, <StartDate>[, <EndDate>[, <TimeResolution>]]]

Query parameters:

<ID> ID of the utilization you want to get information about.

<StartDate> String containing the start date of the evaluation period (ISO8601 format).
If you do not define a start date, the command returns data starting with the data acquisition one month before the end date.

<EndDate> String containing the end date of the evaluation period (ISO8601 format).
If you do not define a end date, the command returns data for a month starting on the start date.

<Granularity>
Resolution of the evaluation period (default = usage over a period of 1 day or 86 400 seconds).
You can define the resolution directly in terms of seconds as a number or indirectly in terms of SEC, MIN, HOUR or DAY (for example `7DAY` results in a resolution of one week). The base unit is seconds.

Default unit: s

Return values:

<History>
<block_data>

Utilization usage of the connected instrument during the time period you have defined. The number of return values depends on the selected time period and time resolution. For example, if you evaluate over a time period of one week with a resolution of one day, the command returns 7 values.
Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block.
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value.
<Length>
Decimal value
Number of bytes that follow in the `<Binary data>` part.
<Binary data>
Binary data in ASCII format
The binary data represents the utilization activity. Each utilization is assigned a timestamp for when the utilization has been active. Together they form a pair of values.
`#blockdata{"timestamps":`
`[<timestamp1>,<timestamp2>,...],"activity":`
`[<activity1>,<activity2>,...]}`
Each timestamp therefore has a corresponding activity:
`<timestamp1> + <activity1>, <timestamp2> + <activity2>`

Example:
//Return utilization history for a specific utilization
`DIAG:HUMS:UTIL:HIST? 1`
Returns for example:
`#3440{"timestamps":`
`[1612259226,1612345626,1612432026,1612518426],"activity":`
`[0,0,6000,18000]}`

Usage:
Query only

7.4 Custom utilizations commands

DIAGnostic:HUMS:UTILization:CUSTom:ADD <ID>, <Name>, <Description>, <Unit>, <State>

Creates a custom utilization.

Setting parameters:

<ID> Identifier of the custom utilization.
Range: 1 to 99

<Name> Name of the custom utilization that appears in the user interface.

<Description> Description of the custom utilization.

<Unit> Unit of the custom utilization that appears in the user interface.

<State> ON | OFF | 1 | 0
Initial activity state of custom utilization.

Example:

```
//Create a custom utilization
DIAG:HUMS:UTIL:CUST:ADD
1,"TrackingSomething","This utilization tracks
something","Hz",1
//Set an initial absolute value for the utilization with ID = 1; initial
start value = 0
DIAG:HUMS:UTIL:CUST:UPD 1,25,abs
//Upon an event, the utilization changes its value each time the
event occurs
DIAG:HUMS:UTIL:CUST:UPD 1,5,incr
//New value = 30, new start value = 25
//Another event
DIAG:HUMS:UTIL:CUST:UPD 1,5,incr
//New value = 35, new start value = 30 etc.
//Query all available custom utilizations
DIAG:HUMS:UTIL:CUST:ALL
//Delete the custom utilization with ID = 1
DIAG:HUMS:UTIL:CUST:DEL 1
//Delete all custom utilizations
DIAG:HUMS:UTIL:CUST:DEL:ALL
```

Usage: Setting only

DIAGnostic:HUMS:UTILization:CUSTom:ALL?

Queries all custom utilizations currently in use.

Return values:

<block_data> String returns REST endpoints as block data in a comma-separated list.
Binary block data with the following syntax:
`#<Digits><Length><Binarydata>`
#
Indicates the start of the binary block
<Digits>
Decimal value
Gives the number of decimal digits used for the `<Length>` value
<Length>
Decimal value
Number of bytes the follow in the `<Binary data>` part
<Binary data>
Binary data in ASCII format

Example: See `DIAGnostic:HUMS:UTILization:CUSTom:ADD`

Usage: Query only

DIAGnostic:HUMS:UTILization:CUSTom:DELete <ID>

Deletes a single custom utilization.

Setting parameters:

<ID> ID of the utilization you want to delete.

Example: See `DIAGnostic:HUMS:UTILization:CUSTom:ADD`

Usage: Setting only

DIAGnostic:HUMS:UTILization:CUSTom:DELete:ALL

Deletes all custom utilizations.

Example: See `DIAGnostic:HUMS:UTILization:CUSTom:ADD`

Usage: Event

DIAGnostic:HUMS:UTILization:CUSTom:UPDate <ID>, <Value>, <Mode>[, <State>]

Changes the current values of the custom utilization (and indirectly also the start value).

Setting parameters:

<ID> Identifier of the custom utilization.

<Value> Defines the value to be changed.

<Mode> **ABSolute**
Defines an absolute value for the utilization.
For example: Currently, the utilization value = 5. If you define an absolute value of 1, the new value is 1.

INCRement
Increases or decreases the current value by a certain amount.
For example: Currently, the utilization value = 5. If you increment it by 1, the new value is 6.
If you you specify a negative number, it decreases the current value.

In both cases, the previous value becomes the new start value.

<State> ON | OFF | 1 | 0

Example: See `DIAGnostic:HUMS:UTILization:CUSTom:ADD`

Usage: Setting only

7.5 Service data commands

DIAGnostic:SERVice:CALibration:DATE <CalibrationDate>

Defines last date and time the instrument was calibrated in ISO 8601 format.

Parameters:
<CalibrationDate> String containing calibration date of the instrument.

Example: //Set calibration date
```
DIAG:SERV:CAL:DATE "2019-05-05T00:00:00Z"
```

DIAGnostic:SERVice:CALibration:DUE:DATE <DueDate>

Defines next date and time the instrument needs calibration to be done in ISO 8601 format. The response may be empty in case of no fixed next calibration due.

Parameters:
<DueDate> String containing next calibration due date.
An empty string resets the date (= no due date).

Example: //Set calibration due date
```
DIAG:SERV:CAL:DUE:DATE "2020-05-12T00:00:00Z"
```

DIAGnostic:SERVice:CALibration:DUE:STATe?

Queries the state if the calibration is OK or the instrument requires calibration.

Return values:

<State>

NAN
Not a number, e.g. if no due date is set.

OK
Calibration is OK since due date has not expired yet.

EXPired
Calibration due date has expired. Calibration is needed.

Example:

```
//Return calibration due state
DIAG:SERV:CAL:DUE:STAT?
```

Usage: Query only

DIAGnostic:SERVice:CALibration:INTerval <Duration>

Defines the recommended calibration interval (ISO 8601 duration).

Parameters:

<Duration> String containing duration of the calibration interval.

Example:

```
//Set calibration interval to a period of 4 years (=P4Y)
DIAG:SERV:CAL:INT P4Y
```

DIAGnostic:SERVice:DATE <ServiceDate>

Defines the last date and time the instrument was serviced (ISO 8601 format).

Parameters:

<ServiceDate> String containing last service date.

Example:

```
//Return last service date
DIAG:SERV:DATE?
```

DIAGnostic:SERVice:STATe?

Queries the state if the instrument requires service.

Return values:

<ServiceState>

NAN
Not a number, e.g. if no service date is set.

OK
Instrument is OK. No service is needed.

DEViations
Deviations have been detected. Service might be required soon.

REQuired
Instrument requires service.

Example: //Return service state
`DIAG:SERV:STAT?`

Usage: Query only

8 Supported products

The HUMS application is available on the following Rohde & Schwarz products.

Table 8-1: Signal and spectrum analyzers

Instrument
R&S FSW
R&S FSV3000
R&S FSWP
R&S ESW
R&S FSMR3000

Table 8-2: Vector network analyzers

Instrument
R&S ZNA
R&S ZNB
R&S ZNBT
R&S ZND

Table 8-3: Signal generators

Instrument
R&S SMW200A
R&S SMM100A

Table 8-4: Radio communication tester

Instrument
R&S CMP180
R&S CMP200
R&S CMX500

9 Overview of status messages

The HUMS application tracks various device status messages.

For a comprehensive description of the information that is tracked for each status message, see Chapter 5.6, "Status", on page 18.

The type of status message that is tracked depends on the instrument group (for example analyzers or signal generators). Note that within a product group, specific firmware releases might not (yet) support all listed utilizations.

Status messages that track optional hard- or software only appear if the optional hard- or software is actually installed on the device.

For a detailed description of each status message, refer to the user manual of the corresponding instrument.

Table 9-1: Status messages tracked by signal and spectrum analyzers and receivers

Status message
RF overload
IF overload
Input overload
UNCAL
Oven cold
LO unlocked
External reference missing
Calibrating
Deviation from self alignment temperature
Last self test state
Last self alignment state
System messages

Table 9-2: Status messages tracked by vector network analyzers

Status message
Last selftest
Selftest
System messages

10 Overview of utilizations

The HUMS application supports various utilizations.

For a comprehensive description of the information that is tracked for each utilization, see Chapter 5.10, "Utilizations", on page 22.

- Basic utilizations supported by all instruments.
- Utilizations specific to a certain instrument group (for example analyzers or signal generators).
  Note that within a product group, specific firmware releases might not (yet) support all listed utilizations.
  Utilizations that track optional hard- or software only appear if the optional hard- or software is actually installed on the device.

Utilizations are available for the following instrument groups.

- Basic utilizations
- Utilizations for signal and spectrum analyzers
- Utilizations for vector network analyzers
- Utilizations for radio communication testers
- Utilizations for signal generators

Table 10-1: Basic utilizations supported by all instruments

Utilization name
Software starts
Power on time
Local keyboard mouse events[1]
SCPI Tx
SCPI commands
SCPI remote
SCPI Rx
SCPI connections
HUMS REST request
HUMS SNMP request

[1]Note that the service does not track which functions you are using, only the key strokes in general.

Table 10-2: Utilizations supported by signal and spectrum analyzers and receivers

Utilization name
Active measurement time[1]
Self alignments
Self alignments duration
Self tests

Utilization name
Firmware updates
Instrument status[2]
<InstrumentStatus> activations
<InstrumentStatus> active time
General hardware usage
Internal reference active time
External reference active time
Sync trigger active time
Mechanical attenuation <x> dB activations
Preamplifier activations
Calibration source activations
AC/DC coupling activations
Analog baseband: mechanical attenuation <x> dB activations
Analog baseband: calibration source activations
Analog baseband: input configuration activations
Analog baseband: calibration signal (AC/DC) activations
Signal path baseband input I to RF activations
Preselector <usage>[3]
Frequency usage[4]
<FrequencyRange> activations
<FrequencyRange> active time
<FrequencyRange> active measurement time
Bandwidth usage[5]
<BandwidthRange> activations
<BandwidthRange> active time
<BandwidthRange> active measurement time
Application usage[6]
Mode <applicationName> activations
Mode <applicationName> active time
Mode <applicationName> active measurement time

[1]Accumulated measurement time over all applications

[2] For a detailed description of possible states (for example RF overload), refer to the user manual of the analyzer

[3]Information about preselector usage (receiver only)

Utilization name

[4] The frequency ranges correspond to the available instrument models. They are, for example, as follows: ≤ 8 GHz - 13.6 GHz - 26.5 GHz - 43 GHz ...

The "FrequencyMax + Allowance" entry accumulates frequencies that do not fit into one of the ranges.

[5] The bandwidth ranges correspond to the bandwidth options installed on the instrument. They are, for example, as follows: ≤ 10 MHz - 28 MHz - 40 MHz - 80 MHz - 160 MHz ...

[6]Available applications depend on the analyzer. For a list of available applications, refer to the datasheet of the analyzer.

These numbers aggregate the statistics for all options that belong to an application. For example, "Mode Amplifier" aggregates the statistics for R&S FSx-K18, -K18D and -K18F.

Table 10-3: Utilizations supported by vector network analyzers

Utilization name
Mode <applicationName> sweep count[1,2]

[1]Number of measurement sweeps that have occured in the corresponding application.

[2]Available applications depend on the analyzer. For a list of available applications, refer to the datasheet of the analyzer.

Table 10-4: Utilizations supported by radio communication tester

Utilization name
Active signaling time
Active measurement time
Active generator time

Table 10-5: Utilizations supported by signal generators

Utilization name
Hardware usage[1]
RF <x> activations
RF <x> active time
Application usage[2]
Kxx #1 activations
Kxx #1 active time

[1]Available RF paths depend on the signal generator and its configuration.

[2]Available applications depend on the analyzer. For a list of available applications, refer to the datasheet of the analyzer. Note that only installed applications are displayed, and only after it has been used at least once.

List of commands