1 Preface

In order to efficiently operate IT structures which are networked on a large scale, an integrated management scheme is needed which covers not only network management but also the administration of the systems and their components, and the applications.
Today, the basis for this type of management concept is provided primarily by the Internet standards for the Simple Network Management Protocol (SNMP). With the support of this "open" protocol in BS2000/OSD, BS2000 mainframes can be accessed from any management platform.

1.1 SNMP management for *open*Net Server and *inter*Net Services

The volume and structure of the information made available is described by Management Information Bases (MIBs).

The *open*Net Server delivery unit supports the following MIBs:

- MIB-II
  which contains, for example, information for TCP/IP-based communication according to RFC 1213.
- BCAM-MIB (private MIB)
  contains BCAM-specific values. The BCAM-MIB supplies, for example, information about the protocols NEA, ISO and TCP/IP and represents them on the transport system.

The *inter*Net Services delivery unit supports the FTP-MIB, which contains information for data transfer based on the FTP protocol.

SNMP-Basic-Agent BS2000

The master agent in the product SNMP-Basic-Agent BS2000 provides the basis for SNMP integration in BS2000/OSD. It is primarily responsible for communication with the SNMP manager and handles the SNMP protocol from a central site.

The master agent also assumes responsibility for the following central tasks:

– managing the management jobs
– distributing the SNMP requests on the executing subagents
– collecting and delivering the results to the manager

The master agent is also responsible for outputting and setting the values of the system group and SNMP group of the MIB-II. The subagents are subordinate to the master agent.

The subagents cannot function unless the master agent has been started. During the initialization phase the subagent logs on to the master agent and transfers its MIB to the master agent.

Subagents function event-oriented. After initialization, the subagent executes in a wait loop. It leaves the loop when an event arrives which it is to process. A event could be, for example, a request from the master agent, a timeout or the receipt of a predefined signal. Once the subagent has processed all existing events, it returns to its wait loop.

1.2 BCAM

BCAM (**B**asic **C**ommunication **A**ccess **M**ethod) forms the basis for the data communication system for BS2000/OSD hosts. It supports homogeneous and heterogeneous connections to WANs and LANs in accordance with CCITT, ISO, TCP/IP and SIEMENS recommendations and standards.
BCAM is used for generation, administration, and control of data communication to and from BS2000/OSD hosts, as well as within BS2000/OSD hosts.
More details can be found in the manuals “BCAM V16.0“, volumes 1 and 2.

BCAM is included in the *open*Net Server delivery unit.

1.3 Target groups

This manual is aimed at network planners, administrators and operators as well as system administrators who integrate the BS2000/OSD systems into an SNMP-based network, system and application management, or those who wish to operate such a system. Prior knowledge of the BS2000/OSD operating system and the basic TCP/IP terms is assumed.

1.4 Summary of contents

This manual provides a complete description of the MIB-II (RFC 1213) supplied with the *open*Net Server delivery unit for SNMP management and the proprietary BCAM-MIB, as well as the FTP-MIB (RFC 959) supplied with the *inter*Net Services delivery unit. The manual also provides detailed information on the installation and operation of the affiliated SNMP subagents.

This manual has the following structure:

- Chapter 2
  provides an introduction to the functionality and operation of the BCAM Manager. This includes how to enter the systems to be monitored, querying MIB variables, the use of search sets, and the graphic functions.
- Chapter 3
  provides information on the software requirements and installing the subagents, as well as sections on how to install and uninstall the agents and the commands for starting and stopping the subagents.
- Chapter 4
  provides a description of the window elements of the BCAM Manager together with detail on how to work in the windows.
- Chapter 5
  provides a description of the SNMP subagent for FTP.
- Chapter 6
  provides a list of the objects of the MIB-II, BCAM-MIB (private) and FTP-MIB and the information they supply.

1.5 Changes since the last version of the manual

- SNMP management for *open*Net Server and *inter*Net Services also support an SNMP subagent for the FTP server (FTP subagent) and the corresponding FTP-MIB.
- A mapping group has been added to the BCAM-MIB.

1.6 Notational conventions

The following notational conventions are used in this manual:

italic font
: indicate the names of files, programs, windows, parameters, menus titles and menu items, as well as commands and variables in running text.

<angle brackets>
: indicate variables which you must replace with concrete values.

`fixed-width font`
: indicates input for the system, system output and file names in examples.

command
: In the syntax description for commands, components (names of commands and parameters) in bold type must be entered as given.


For informative texts


for warnings.

References

References within this manual include the page concerned in the manual and the section or chapter as necessary. References to topics in other manuals include the brief title of the manual concerned. You will find the full title in the list of related publications at the end of this manual.

1.7 README file

Information on any functional changes and additions to the current product version described in this manual can be found in the product-specific README file. You will find the README file on your BS2000/OSD computer under the file name SYSRME.BCAM.160.E or SYSRME.TCP-IP-SV.040.E or SYSRME.TCP-IP-AP.040.E. The user ID under which the README file is cataloged can be obtained from your system administrator. You can view the README file using the /SHOW-FILE command or an editor, and print it out on a standard printer using the following command:

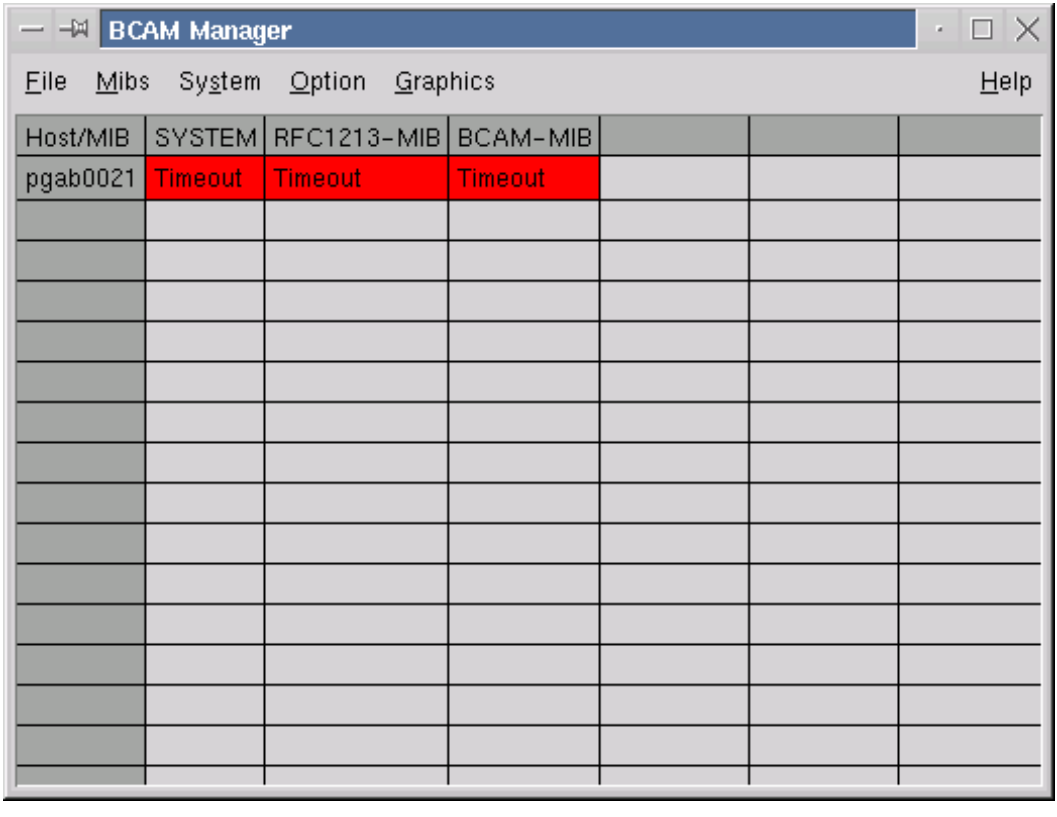
```
/PRINT-DOCUMENT <filename>, LINESPACING=*BY-EBCDIC-CONTROL
```

for SPOOL versions prior to 3.0A:

```
/PRINT-FILE FILE-NAME=filename,LAYOUT-CONTROL=
    PARAMETERS(CONTROL-CHARACTERS=EBCDIC)
```

2 Functionality of the BCAM Manager

The BCAM Manager is an SNMP management application which has been optimized for the MIB-II, the private BCAM-MIB and the FTP-MIB.



Main BCAM Manager window

The BCAM Manager provides the following functionality:

– monitoring several systems
– support of any MIBs
– definition of print names for any MIB variable
– definition of standard variables for any MIB group or table
– search for table instances using any criteria
– search for table instances using criteria from other tables for the BCAM-MIB
– definition and storage of search queries
– definition of graphic functions with bar charts and graphs (line diagrams)
– automatic search for new table instances

Introduction

This section provides you with a brief introduction to using the BCAM Manager.
Once you start the BCAM Manager, the main *BCAM Manager* window is displayed (see page 7).

You use this window to perform the following tasks, for example:

– add new MIBs and new systems
– for configuration purposes
– call the system views

Once you have started the BCAM Manager, the RFC1213-MIB (MIB-II), BCAM.MIB and FTP-MIB are available. The known MIBs are displayed in the first row of the table in the *BCAM Manager* window.

Experienced users can request further MIBs by loading these MIBs by selecting *Load MIB* from the *Mibs* menu in the main *BCAM Manager* window.

You will find a description of the menu items used to perform the following tasks below:

– Entering the systems to be monitored
– Querying MIB variables
– Using search sets
– Defining graphic functions
– Calling graphic functions

Entering the systems to be monitored

1. Select *System* → *Add/Modify System* from the menu in the *BCAM Manager* window.

   The systems are entered in the other rows in the table in the *BCAM Manager* window together with the status of the MIB agents.

2. Click on a system name to display the status of all the MIB agents of this system.

3. Select *File* → *Save Systems* from the menu in the *BCAM Manager* window to save information about the systems in a file.

Select *File* → *Load Systems* from the menu in the *BCAM Manager* window to load information about the systems again.

Querying MIB variables

1. Select *System* → *Systemview for MIB* → *Systemview* <*MIB-name*>from the menu in the *BCAM Manager* window to query variables of the MIB <MIB-name>.

   The *Systemview* <*MIB-name*> window contains a tab for each MIB group. The tab itself is divided up into two sections if the MIB group includes MIB tables. The section on the left contains the MIB group variables and the section on the right the MIB table variables.

   - You query the MIB group variables by clicking on the *Update* button.

     You can modify which MIB variables are displayed by default by clicking on the *Standard variables* button (or by selecting the menu items *Option* → *Define standard variables for*).

   - You query the MIB table variables with the *Select* <*MIB table name*> button.

     A dialog is displayed which allows you to restrict the MIB table instances displayed to those that satisfy certain criteria.

     - If you want to display all the instances, all you have to do is click on the *Start* button.

     - If you want to restrict display, use the right mouse button to display a pop-up menu which offers you additional functions including the menu item *Search function,* which allows you to perform a search using predefined search sets.

2. Select *BCAM Manager* → *System* → *Systemview for MIB* → *Systemview* <*MIB-name*>. In the selection field *System* you can select another System, whose MIB variables you want to be displayed.

Using search sets

Searching for MIB table instances which satisfy certain criteria can be done using pre-defined search sets. A search set contains a number of MIB variables and the values that these variable can have in order to fulfill the criterion.

► Select *Option → Add/Modify search set for* from the menu in the *BCAM Manager* window to define and store search sets.

Defining graphic functions

► Select *Option → Define display functions for* from the menu in the *BCAM Manager* window to define a graphic function.

Calling a graphic function

To call a graphic function, do the following:

1. Select *System → Systemview for MIB → Systemview <MIB-name>*from the menu in the *BCAM Manager* window.
2. You can activate a pop-up menu that contains the item *Graphic functions* when you press and hold down the right mouse button with the mouse cursor positioned on a tab in the *Systemview <MIB-name>* window.

   ► Activate the pop-up menu with the mouse cursor positioned on the desired tab in the *Systemview <MIB-name>* window.
3. Click on the menu item *Graphic functions* in the activated pop-up menu using the left mouse button:

   – In the case of MIB group variables, a new graphics window is immediately displayed.

   – In the case of MIB tables variables, a dialog is displayed which allows you to restrict the instances to be displayed using a search set. You can also modify any search set you select. However, this modification is not saved.

     ► Click on the *Start* button, to generate a new graphics window.
4. Select the graphic function to be displayed in the selection field *Function* in the graphics window.

   You can do the following:

   ► Click on the *Start* button to start the query and the graphic representation of the values found.

   ► Click on the *New Display* button to divide the graphics window so that several graphic functions can be displayed simultaneously.

3 SNMP for *open*Net Server and *inter*Net Services

Starting with BCAM V14.0, two subagents are supplied to support SNMP management in BS2000/OSD:

– The MIB-II subagent provides a standardized MIB-II (based on RFC1213) for network management tasks.

– The MIB of the BCAM subagent contains BCAM-specific values.

From *inter*Net Services V2.0, a subagent for FTP is provided together with a corresponding MIB (RFC 959).

A precondition for the operation of the subagents is the use of SNMP-Basic-Agent BS2000.

More details on management applications, master agents and error handling can be found in the manual "SNMP Management in BS2000/OSD".

The command programs for the BCAM, MIB-II and FTP subagents are called with /START-BCAMCMD, /START-MIB2CMD and /START-FTPCMD, respectively.

The ASN1 source files for the BCAM and MIB-II MIBs are included in the library SYSSPR.BCAM.160 as members of type S under the names BCAM.MIB and MIB-2.MIB, respectively.

The ASN1 source file for the FTP-MIB is included in the library SYSSPR.TCP-IP-AP.040 as a type-S member under the name FTP.MIB.

This chapter provides information about:

– software requirements
– installing and uninstalling the subagents
– starting and stopping the subagents
– trace files

3.1 Software requirements

The following tables list the software that is needed in order to use the subagents supplied with *open*Net Server V2.0.

BS2000/OSD as of V2.0:
TV-SBA-BS2 ≥ V2.1 or SBA-BS2 V5.0
POSIX-BC ≥ V1.0 *
SDF-P-BASYS V2.0B
FTP ≥ 4.0
optional:
JV ≥ V11.2

Software requirements for the use of *open*Net Server subagents.
The components identified by * are a part of BS2000/OSD-BC

3.2 Installing and uninstalling the subagents

The MIB-II subagent (RFC 1213), the BCAM subagent (BCAM private MIB) and the FTP subagent (FTP-MIB) are installed with the software delivery and information system SOLIS2.
The SOLIS2 installation includes the required BS2000/OSD-specific tasks such as the MSGFILE update, subsystem catalog entries, etc. In order to install the subagents, you will need to have SUBSYSTEM-MANAGEMENT privileges.

Please note that internal communication

– between the master agent and the MIB-II or BCAM subagent is handled via port number 3161.

– between the master agent and the FTP subagent is handled via port number 3237.

In particular, the dynamic assignment of port numbers by BCAM should begin with a higher value. The BCAM default value is 4096.

Deleting the SINLIB after installation leads to errors, since the agents also need the SINLIB for operation.

3.2.1 Installation

The POSIX subsystem must be started beforehand. The executable agents can be found in the SINLIB.BCAM.160, which also includes all elements that need to be installed in the NFS. The installation is performed under the SYSROOT or TSOS ID (UID=0, GID=0) with the POSIX installation tool:

```
/CALL-PROCEDURE *LIB(LIB=$TSOS.SINPRC.POSIX-BC.<posix-bc-version>,
                 ELEMENT=POSINST)
```

Function: Install POSIX program packages

- for the BCAM subagent and MIB-II subagent:

  Product name: BCAM
  Product version: 160

- for the FTP subagent

  Product name: TCP-IP-AP
  Product version: 040

3.2.2 Uninstallation

Uninstallation is also performed under the SYSROOT or TSOS ID (UID=0, GID=0) with the POSIX installation tool:

```
/CALL-PROCEDURE *LIB(LIB=$TSOS.SINPRC.POSIX-BC.<posix-bc-version>,
                ELEMENT=POSINST)
```

Function: Uninstall POSIX program packages

- for the BCAM subagent and MIB-II subagent:

  Product name: BCAM
  Product version: 160

- for the FTP subagent

  Product name: TCP-IP-AP
  Product version: 040

3.3 Starting and stopping

The subagents operate only if the master agent has been started, but can otherwise be started and stopped individually at any time.

The following prerequisites must be satisfied in order to start the agents:

– an operational TCP/IP connection between the BS2000/OSD host and the manager
– a started POSIX subsystem
– an installed SNMP subsystem
– the POSIX root authorization
– as well as the following privileges:

Command	**Privilege in BS2000/OSD ≥ V2.0**
START-SNMP-MIB-MIB2	NET-ADMINISTRATION
START-SNMP-MIB-BCAM	NET-ADMINISTRATION
START-SNMP-FTP	NET-ADMINISTRATION

Agents must be stopped under a user ID with POSIX root privileges.

The following autostart/stop scripts are copied to the POSIX file system during installation:

– */etc/rc2.d/S91snmpbcam*
– */etc/rc0.d/K11snmpbcam*

If you want an automatic start when POSIX is started up, please remove the comment characters from the appropriate places in *S91snmpbcam*.

3.3.1 MIB-II subagent

The MIB-II subagent supports read access and, to some extent, write access on the MIB-II based on RFC1213.

Starting the MIB-II subagent in BS2000/OSD:

/START-SNMP-MIB-MIB2
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54> , **CPU-LIMIT**=***JOB-REST** / <integer 1 .. 32767> , **JOB-CLASS**=***STD** / <name 1 .. 8> , **TIMER-INTERVAL**= 5 / <integer 1 .. 32767>

or in the POSIX shell with:

```
mib2agt
```

Stopping the MIB-II subagent in BS2000/OSD:

/STOP-SNMP-MIB-MIB2
VERSION=***STD** / <product-version>

or in the POSIX shell with:

```
mib2cmd T
```

Description of the operands see page 18.

3.3.2 BCAM subagent

Starting the BCAM subagent in BS2000/OSD:

/START-SNMP-MIB-BCAM
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54> , **CPU-LIMIT**=***JOB-REST** / <integer 1 .. 32767> , **JOB-CLASS**=***STD** / <name 1 .. 8> , **TIMER-INTERVAL**= 5 / <integer 1 .. 32767>

or in the POSIX shell with:

```
bcamagt
```

Stopping the BCAM subagent in BS2000/OSD:

/STOP-SNMP-MIB-BCAM
VERSION=***STD** / <product-version>

or in the POSIX shell with:

```
bcamcmd T
```

Description of the operands see page 18.

3.3.3 FTP subagent

Starting the FTP subagent in BS2000/OSD:

/START-SNMP-FTP
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54> , **CPU-LIMIT**=***JOB-REST** / <integer 1 .. 32767>

or in the POSIX shell with:

```
ftpagt
```

Stopping the FTP subagent in BS2000/OSD:

/STOP-SNMP-FTP
VERSION=***STD** / <product-version>

or in the POSIX shell with:

```
ftpcmd T
```

Description of the operands see page 18.

3.3.4 Description of operands

VERSION=<product-version>
Defines the version of the agent to be started or stopped. This entry is currently not evaluated.

MONJV=...
Name of the job variable with which the agent is to be monitored.

MONJV=*NONE
The agent is not monitored using job variables.

MONJV=<filename 1 .. 54>
The agent is monitored by the specified job variable.

CPU-LIMIT=...
Specifies, in seconds, the CPU time that may be used by the agent.

CPU-LIMIT=*JOB-REST
The generated default value is used.

CPU-LIMIT=<integer 1 .. 32767>
Specifies the maximum CPU time in seconds.

JOB-CLASS=...
Specifies the job class with which the agent is started.

JOB-CLASS=*STD
The generated default job class is used.

JOB-CLASS=<name 1 .. 8>
Defines the job class to be used.

TIMER-INTERVAL=...

Interval at which the agent checks for changes from the command program. In addition, trap monitoring is performed every 30 seconds. The supervisor subagent checks all the subagents from which no messages has been received during the last five minutes by sending a request.

TIMER-INTERVAL=5 / < 1 .. 32767>

The timer interval is set to 5 seconds by default.

3.3.5 Trace files

When the agents are running, trace files are created under the user ID under which the agent was started. These files contain startup and error information. The trace files are created with the name *SYSTRC.SNMP.*<*agent*>.<*date*>.<*time*> and can, if not otherwise required, be deleted on stopping the corresponding agent. The messages of the agent are stored by default in a trace file in the BS2000 file system.

The name of the trace file *SYSTRC.SNMP.*<*agent*>.<*date*>.<*time*> is created as follows:

<agent> is the name of the agent program.

<date> is the current date in the form: yyyy-mm-dd

<time> is the current time in the form: ssmmss

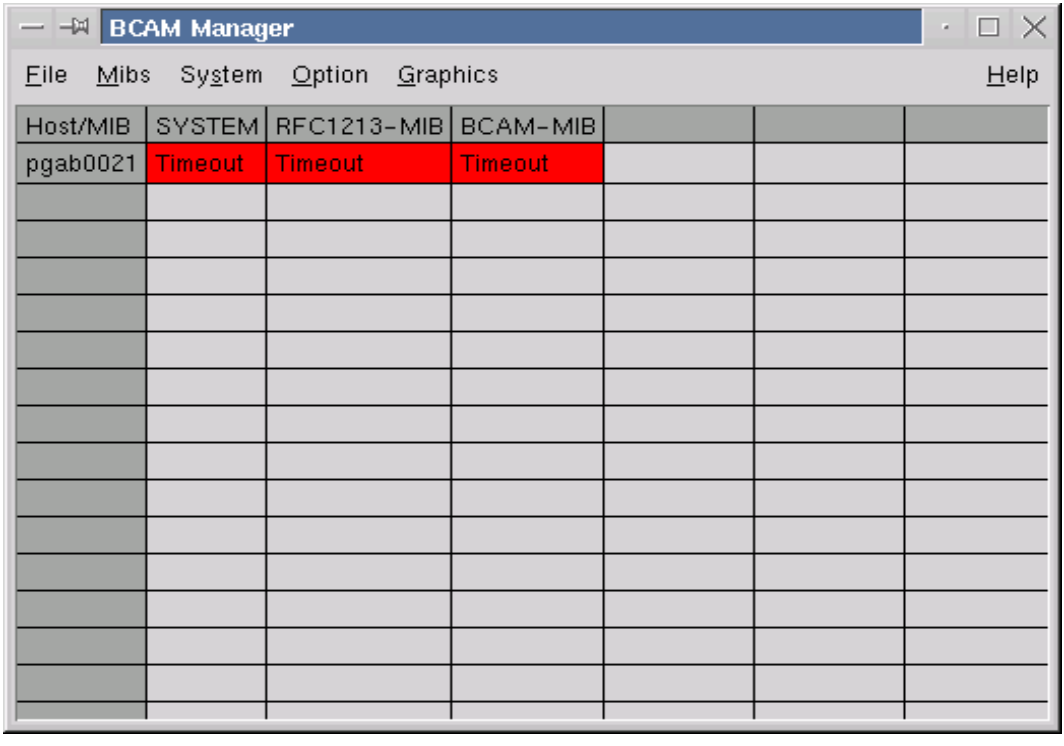
More information on trace files can be found in the manual "SNMP Management in BS2000/OSD".

4 Windows of the BCAM Manager

This chapter introduces you to the windows of the BCAM Manager and describes how to work in these windows. The windows for processing the FTP-MIB are described on page 124ff in the chapter “SNMP subagent for FTP”.

4.1 Main BCAM Manager window

This window provides you with access to the functions of the BCAM network management program.



Main BCAM Manager window

You access these functions using the menu bar. The table provides information about the systems currently being monitored and about the current status of the respective MIB agent of the MIBs which have been loaded.

Window elements and working in the window

The menu bar includes the following menus:

- The *File* menu
  contains commands which influence operation of the management program.
  - The *Exit* command
    terminates the program. All open windows are closed.
  - The *Save Systems* command
    allows you to save the currently monitored systems together with their SNMP options in a file.
  - The *Load Systems* command
    allows you to read in systems and their SNMP options from a file.
- The *Mibs* menu
  contains commands relating to the management program's work with certain management data.
  - *Load MIB* command
    When you select this command, a new dialog window is opened which you use to select the MIB you want to load. The name of the selected MIB is displayed as the name of a column in the table in the window.
- The *System* menu
  contains commands relating to the management program's work with certain systems. Please refer to page 26ff for more information.
  - *Add/Modify System* command
    When you select this command, a new dialog window is opened which you use to make new systems known to the management program or to modify the operating parameters of a known system. A system known to the management program is displayed in the table; the name of this system is displayed as the name of a row.
  - *Show MIB* command
    When you select this command, a submenu is displayed from which you select the MIB for which a view is to be shown. When you select a MIB, a new dialog window is opened.

- The *Option* menu
  contains commands for defining or modifying the work modes of the management program. Options you require often can be predefined for the duration of the session and saved for later sessions. Predefined options can be modified. When you click on a command in this menu, a submenu for selecting a MIB is displayed. A new dialog window is opened when you select an MIB. Please refer to page 94ff for more information.

  – *Define reference variables for* command
    The values of the reference variables are buffered for each system during query operations so that searches can be performed locally. Reference variables can only be defined for MIB tables.

  – *Define display functions for* command
    Graphic display functions can be defined for MIB groups and MIB tables. The management program provides mechanisms for periodically updating the values of MIB variables and displaying them graphically. When defining a display function make sure you select only MIB variables with numeric values.

  – *Define standard variables for* command
    You will encounter standard variables in various windows of the management program. Variables of particular importance or interest can be predefined with this command.

  – *Define print names for* command
    Each MIB variable can be linked with a print name. The print names are freely selectable but must be unique within the MIB.

  – *Add/modify search set for* command
    A search set describes a set of MIB table instances using their common characteristics. Search sets can be used for a number of purposes. They provide the basis for search operations within the MIB tables. If you want to be able to perform complex search operations in different tables, the MIB must have a special structure. It must, for example, be possible to reference a table in the MIB from another table in the MIB.

- The *Help* menu
  - *MIB-Variable Help* command
    When you select this command a new dialog is started in which you can query the meaning of individual MIB variables.
  - *Main Window Help* command
    The help text describing the window is displayed in a text box.
  - *About* command
    provides information on the version of the management program.

Table

The table is used to display information about the status of the systems being monitored and the MIB agents. The systems known to the management program are listed in the first column of the table, and the MIBs which have been loaded are listed in the first row. The fields in the table contain the current status of the system and the MIB agents available in the system.

The meaning of the status:

None
: No test messages have yet been sent.

Testing
: A test message is on its way; no response has yet been received.

Available
: The test message was answered correctly.

Not available
: The test variable could not be queried. The MIB is not supported.

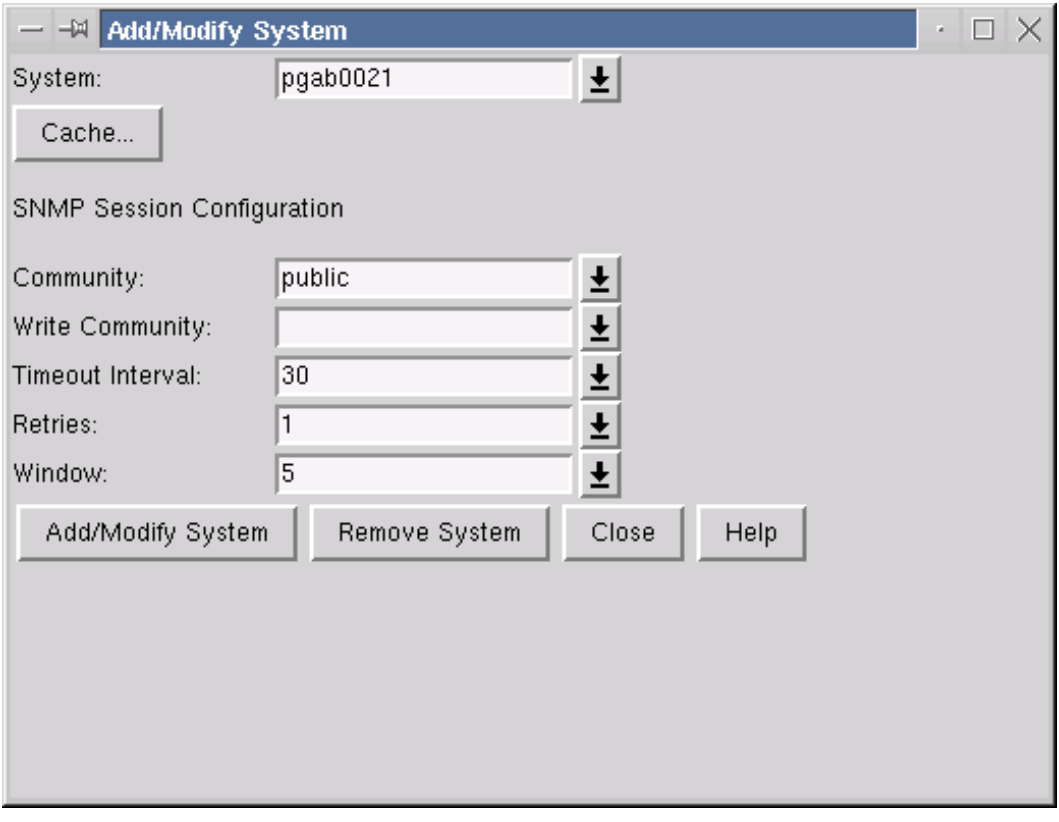
Timeout
: The tested system did not respond within the define timeout interval. It is a good idea to try again.

Failed
: The test variable query failed due to a reason not listed above.

You update the status field by clicking the mouse button within the table. You initiate the testing of all the systems and agents listed in the table by clicking on the upper left field in the table.
You test a single system by clicking on the name of the system.
You test a single MIB by clicking on the name of the MIB.
You can test an individual agent within a certain system by clicking on the appropriate field in the table.

4.2 The System menu

4.2.1 Adding and modifying systems to be monitored



Main BCAM Manager window → System menu → Add/Modify System

This window allows you to add or modify a system to be monitored. You can also influence the SNMP protocol options and the local storage of MIB variables.

Window elements and working in the window

Selection fields

System
indicates which system is to be added or modified.

Community
defines the password to be used for read operations.

Write Community
defines the password to be used for write operations.

Timeout Interval
contains the maximum timeout (in seconds) for processing an SNMP package.

Retries
contains the number of attempts made to send an SNMP package within the interval specified in the field *Timeout Interval* if no response is received.

Window
contains the maximum number of SNMP packages which can be sent simultaneously.

Buttons

Cache
opens a new dialog window which contains the number of entries for each table with cached MIB table instances and a button for deleting the cache.

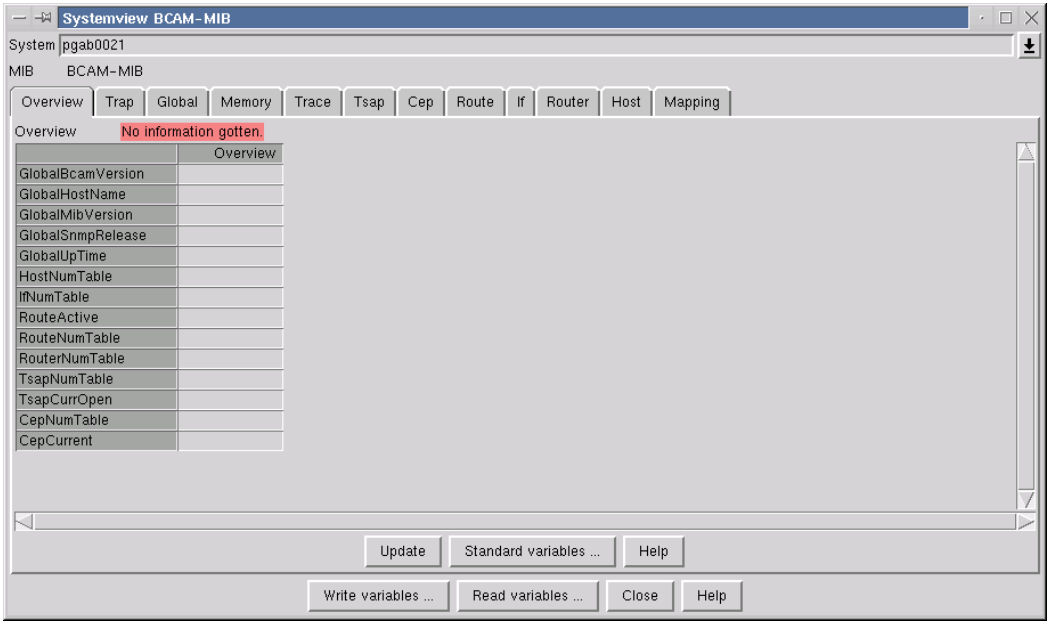
Add System
adds the system displayed in the selection field *System*. In the case of a system already known to the management program, the modified parameters are taken over.

Remove System
deletes the system displayed in the selection field *System*.

Close
closes the window.

Help
displays the appropriate help text.

4.2.2 Start window for querying a certain MIB



Main BCAM Manager window → System menu → Systemview for MIB

This window is the start window for querying a specific MIB. It contains a tab for each group of the MIB being viewed. The tab is divided in two parts if the MIB group contains MIB tables. The MIB group variables are displayed on the left and the MIB table variables on the right. You can modify the selection of the MIB variables using the *Standard variables* button. You can query and display the values of the MIB variables again using the *Update* button. The *Select...* button in the table area allows you to search for MIB table instances using reference variables as the search criteria. Every part of a tab includes a pop-up menu which you display using the right mouse button. The items on the pop-up menu are described in the help text for the relevant tab.

Window elements and working in the window

The selection field *System*
displays the name or address of the current system and allows you to change system. The selection list includes all the systems that have been added.

The *MIB* box
displays the name of the current MIB.

The tab area
includes a tab for each group in the current MIB.

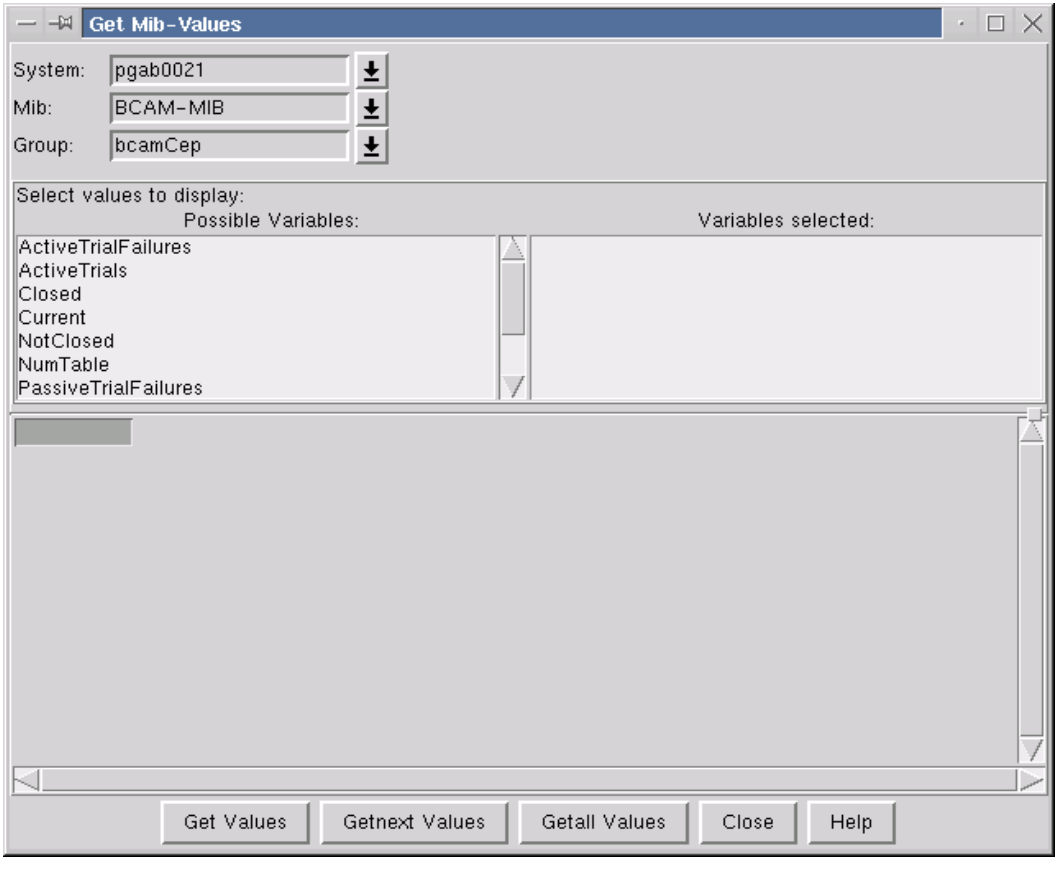
The *Write variables* button
initiates a dialog for writing individual MIB variables from all the MIBs which have been loaded.

The *Read variables* button
initiates a dialog for querying individual MIB variables from all the MIBs which have been loaded.

The *Close* button
closes the window

The *Help* button
displays the appropriate help text.

Querying the MIB variables of a certain system



Main BCAM Manager window → System menu → Systemview for MIB → Read variables button

This dialog window provides you with functions for querying the MIB variables of a certain system.

Window elements and working in the window

Before you can start a query, you must enter the appropriate data in the fields provided for this purpose. You must also select the MIB variables to the queried.

Selection fields

System
: contains the name of the system to be queried.

Mib
: contains the name of the MIB to be queried.

Group
: contains the name of the MIB group or MIB table to be queried.

Indices (optional)
: indicates which instances of the selected MIB table are to be queried.
You may also specify a list of indices separated with blanks.

Selection lists

Possible Variables
: lists all the variables of the selected MIB group or MIB table. You select a variable by clicking on it with the left mouse button. The variable is moved from the list on the left to the selection list on the right.

Variables selected
: contains the variables selected for the query. You deselect a variable by clicking on it with the left mouse button.

Table

The results of a query are displayed in table form.
The first column contains the names of the queried MIB variables. The other columns contain the corresponding values.
The first row contains the indices of the MIB table instances. If a MIB group was queried, the index is 0.

Special error or status indicators are displayed in the upper left-hand table field.

Buttons

Get Values
: starts a query whose response is processed asynchronously. If no variable has been selected for the query, the result is *No Information available*.

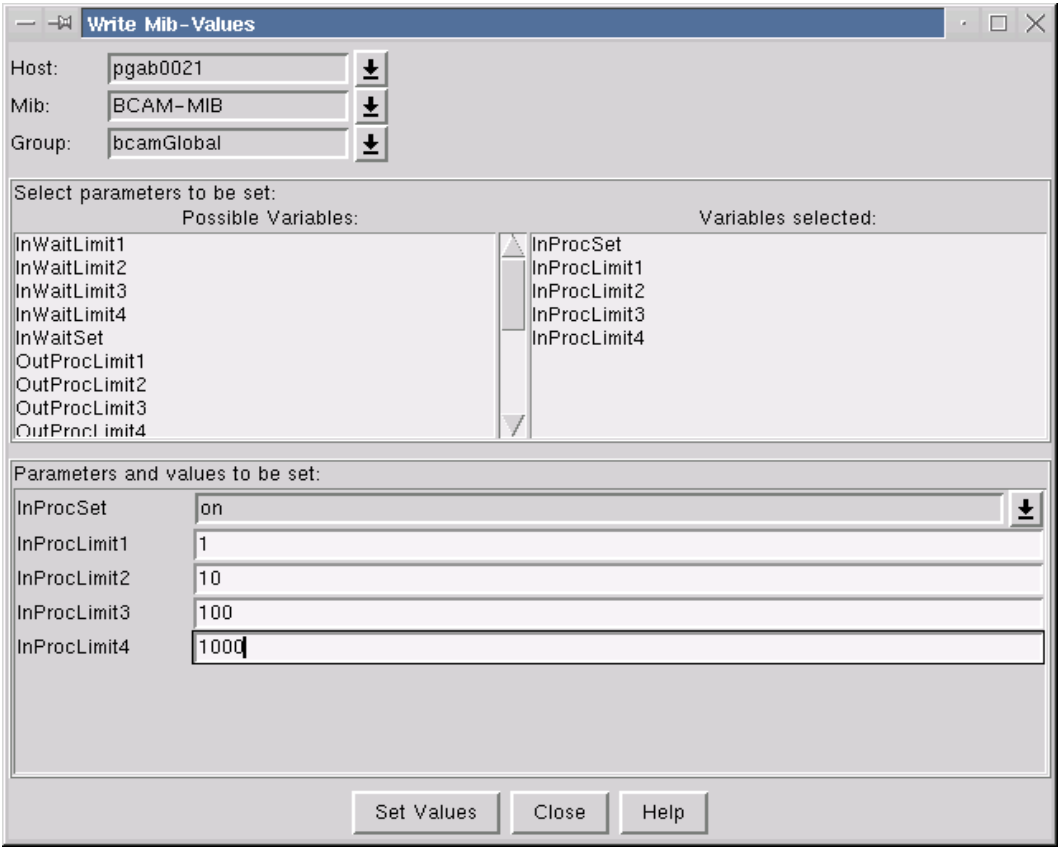
Getnext Values
: starts a query whose response is processed asynchronously. The values of the MIB table instance with the next higher table index are displayed. If no variable has been selected for the query, the result is *No Information available*.
The function is not defined for MIB groups. If this function is called for an MIB group, the result returned is *Function not supported for a group*.
If no table index higher than the one specified exists, *End of Table* is output.

Getall Values
: starts a series of queries whose responses are processed asynchronously. The values of all the instances contained in the MIB table are displayed, starting with the instance with the next higher instance than that specified in the *Indices* field.
If no variable has been selected for the query, no query is performed and no results are displayed.
The function is not defined for MIB groups. If this function is called for an MIB group, the result returned is *Function not supported for a group*.
If no table index higher than the one specified exists, *End of Table* is output.

Close
: closes the window

Help
: displays the appropriate help text.

Performing write operations on MIB variables of a certain system



Main BCAM Manager window → System menu → Systemview for MIB → Write variables button

This dialog window provides you with functions for performing write operations on MIB variables of a certain system.
An attempt to perform a write operation is only made if a write password was specified when the system was added. An invalid write password normally causes a timeout and an error message which is displayed in the result window.
Any successful write operation is valid until another write operation by an authorized instance is performed.

Window elements and working in the window

Before a write operation can be performed, you must enter various data in the fields provided for this purpose. You must also select the MIB variables to be written.

Selection fields

Host
: contains the name of the system to be modified.

Mib
: contains the name of the MIB to be modified.

Group
: contains the name of the MIB group or MIB table to be modified.

Indices (optional)
: indicates which instances of the selected MIB table are to be modified. You may also specify a list of indices separated with blanks.

List boxes

Possible Variables
: lists all the variables of the selected MIB group or MIB table which can be written. You select a variable by clicking on it with the left mouse button. The variable is moved from the list on the left to the selection list on the right.

Variables selected
: contains the variables selected for modification. You deselect a variable by clicking on it with the left mouse button.

The input list *Parameters and values to be set*
: contains the selected variables and an associated input field in which you enter the value to be written.

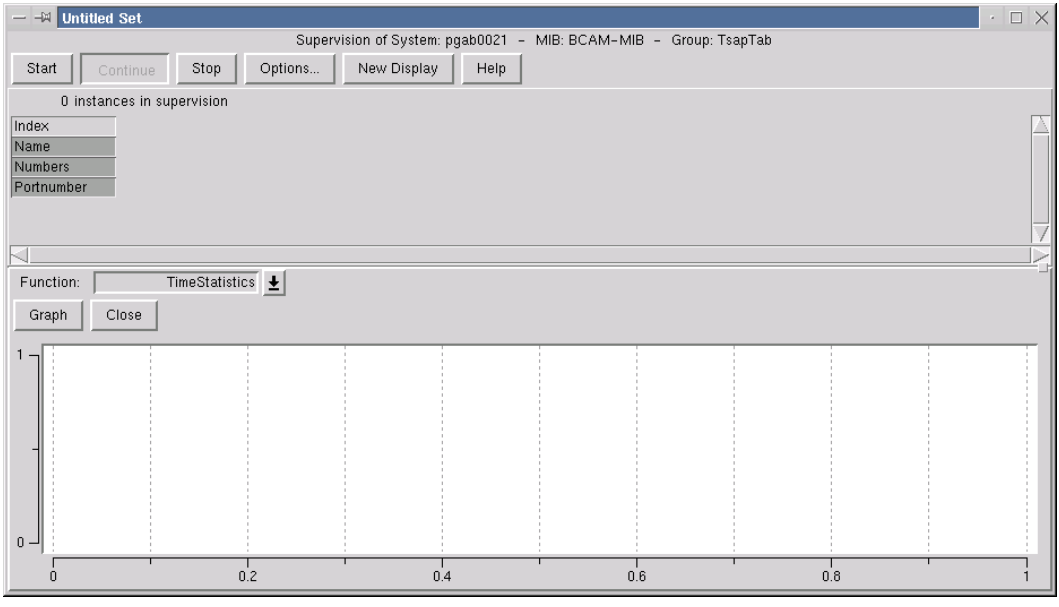
Buttons

Set values
: starts the write operation, the result of which is processed asynchronously.

Close
: closes the window

Help
: displays the appropriate help text.

Displaying the results of functions defined for a MIB group or MIB table



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → right mouse button → Graphic functions

This window is used to display the results of the functions defined for a MIB group or MIB table. The function values and their graphic display are updated periodically. The window is divided into a control section and a display section. In the case of functions for MIB tables, a table is inserted between these two sections and is used to display the selected table instances.

Window elements and working in the window

Control section

The top line provides you with information on the monitored system, the MIB and the MIB group or MIB table. It is recommended that you check the settings of the operating options before you collect the values.

Buttons

Start
: starts the search for appropriate MIB table instances (this step is omitted for MIB groups). The first collection of the values is then performed.

Stop
: stops collection of the values. A search or query currently being executed is completed but no follow-up operation initiated. The search or query interval can be modified in the stop state.

Continue
: continues with the collection. A new search step is initiated starting with the highest table index previously found.
The table instances found up to now are retained.
If the query interval is changed, values previously collected are deleted as soon as collection of the values is continued.
The diagrams in the graphics windows are deleted.

Options
: allows you to check and modify the settings for the operating options.

New Display
: generates a new graphics window.

Help
: displays a description of this window.

Table

The first column of the table contains the name of the MIB table variables to be displayed. The other columns contain the corresponding values (or blanks if no value yet exists). The first row contains the indices of the MIB table instances.

The background color provides additional information about the MIB table instance:

– Yellow indicates that no information was obtained for the relevant MIB table instance.

– Orange indicates that the relevant instance probably no longer exists.

– White indicates that the values for this table instance cannot be displayed graphically.

Selection of the variables is determined by a table-specific standard set.

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

You select a MIB table instance by clicking on the relevant column in the table with the left mouse button. You may select more than one instance. If you click on your selection again, this selection is cancelled.

Pop-up menu

Menu items:

Delete selected columns
: deletes the selected columns in the table irrevocably. The values collected for the deleted instances are discarded. New graphics are displayed. It is recommended that you delete invalid table instances (which are indicated by an orange background color) so that the table is compact and the graphic easy to read.

Get variables of selected columns
: updates the values of the variables in the selected columns.

Hide graphic of selected columns
: hides the graphic representation of the selected columns in all the graphics windows. Values which have already been collected are retained, polling of the instances is continued so that display at a later time is possible.

Reveal graphic of selected columns
: displays the values of the selected instances graphically. This function is only effective if you select an instance which was not previously displayed.

Hide graphic of all instances
: graphic representation is suppressed in all the graphics windows. Values which have already been collected are retained, polling of the instances is continued so that display at a later time is possible.

Reveal graphic of all instances
: The values for all the instances are displayed graphically.

Graphics section

The graphics section contains a graphics window.
A selection field allows you to select which function is displayed. The collected function values for the MIB group or table are displayed as a bar chart or graph.

Selection field

Function
The function displayed in this box determines the context in which the values collected are to be represented.

Buttons

Graph
switches the display to a graph (line chart). If no values exist or no appropriate representation has been defined for the function, the diagram remains empty.

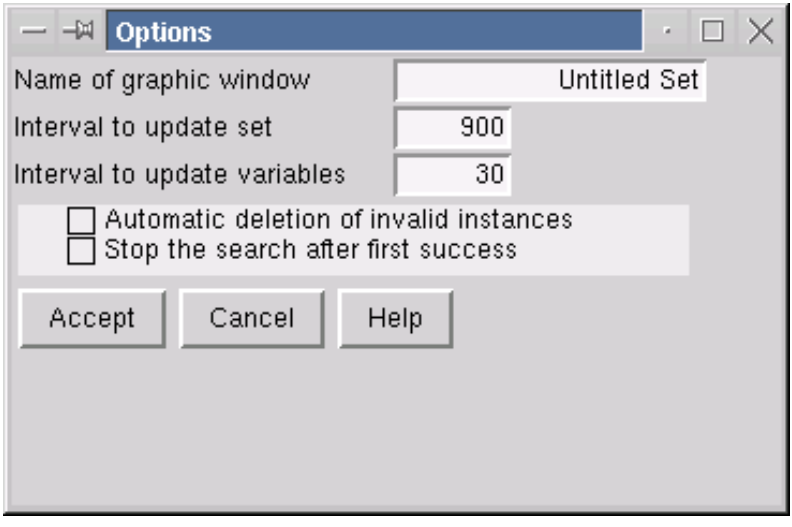
Bar
switch the display to a bar chart. If no values exist or no appropriate representation has been defined for the function, the diagram remains empty.

Close
makes the graphic disappear. The collected values are retained and can be displayed at any time in a new graphics window.

Diagram

When you position the mouse cursor on the diagram and press and hold down the right mouse button, a pop-up menu for selecting a function is displayed.

Controlling how the functions in the graphics window work



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → right mouse button → Graphic functions → Options

This window provides you with the resources you need to control how the functions in the graphics window work. Several options can only be selected if a query has been interrupted.

Window elements and working in the window

Input fields

Name of graphic window
: allows you to change the name of the graphics window. Graphics windows should be given a new name if the graphic functions are not called using a predefined search set (in which case the window name is *Untitled Set*) or if the functions for a search set have been called more than once. This option allows you to keep the windows unique within a management function.

Interval to update set
: defines the interval at which a search for new MIB table instances which satisfy the criteria of a search set is performed. This option is therefore only of significance for functions involving MIB tables.

Interval to update variables
: defines the interval at which the MIB variables required for the graphic functions are updated.

Check boxes

Automatic deletion of invalid instances
: keeps the set of instances as small as possible. An instance is removed from the function if the periodic query function considers it invalid due to certain acknowledgments from the SNMP agents in the monitored system. If you do not select this option, an invalid instance is indicated by an orange color in the table heading.

Stop the search after first success
: prevents further search steps from being executed once a search run has returned at least one suitable instance. You can use this option if you know that the set of instances to be monitored will remain the same.

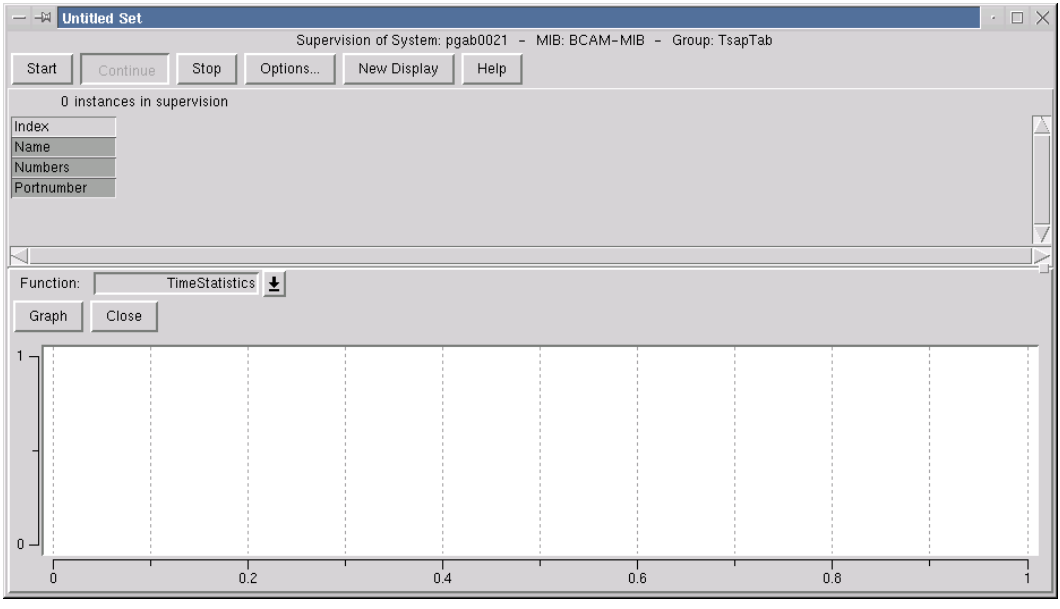
Buttons

Accept
: applies the selected options to the graphic functions.
The option window is closed.

Cancel
: cancels the specifications and selections made. The old options remain valid.
The option window is closed.

Help
: displays the appropriate help text.

Displaying values in a bar chart



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → right mouse button → Graphic functions

This window is used to display the values in a bar chart.
One axis indicates the values collected and calculated by the functions. The other axis is divided up into as many equidistant parts as there are bars in the diagram.
The bars are provided with a label.
A legend is used to group the bars together. The legends are displayed in a separate window. All the bars in a group have the same color.

Window elements and working in the window

There are two ways of modifying a bar chart:

– Operations performed on the actual bar chart (diagram operations)
(the mouse cursor is located in the graphics window)

– Operations provided by a pop-up menu (menu operations)
(press and hold down the right mouse button with the mouse cursor located in the window containing the chart, move the cursor to the desired item on the pop-up menu and release the button).

Diagram operations

Click on a bar with the left mouse button:
The coordinates of the bar are displayed.

Menu operations

Invert Axis:
inverts both axes.

ModeAligned:
displays the bars next to each other. This is how the bars are displayed initially.

ModeStacked:
The first bar represents the first value in all the groups. The second bar represents the second value in all the groups, etc. The legends are used as the labels for the bars. The legend window is closed.

ScalePercent:
The values in a group are represented as a percentage of the sum of all the values of this group.

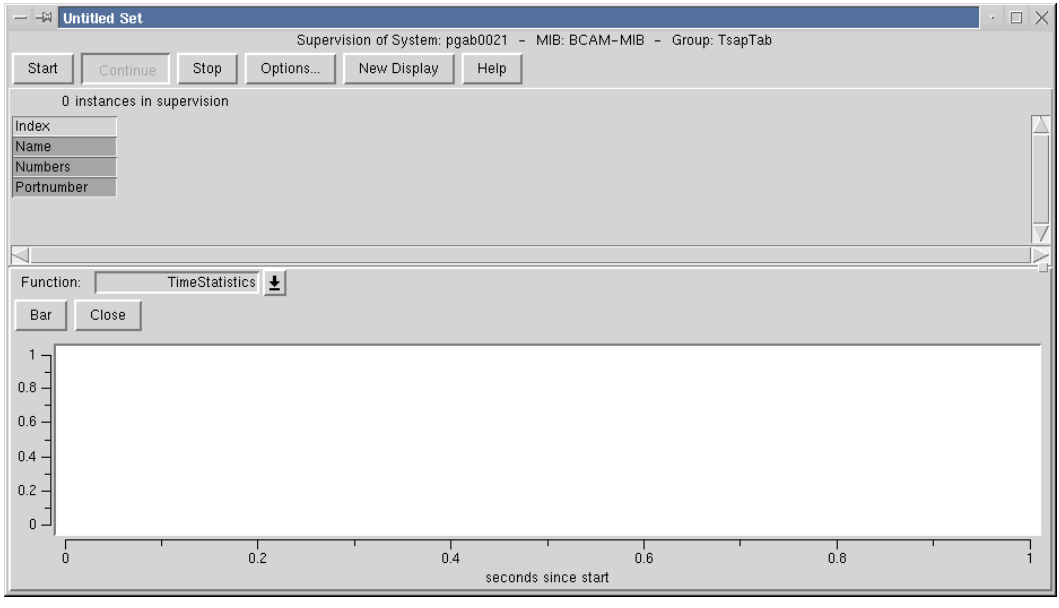
ScaleNormal:
The length of a bar is proportional to its value.

LegendOnOff:
displays and hides the legend window. This command is only effective in the mode *ModeAligned*.

Save Diagram:
saves the graphic in a postscript file.

Help:
displays the appropriate help text.

Displaying vectors in a line diagram (graph)



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → right mouse button → Display functions

This window is used to show vectors in a line diagram.
The values on the horizontal axis indicate the time in seconds since the start.
The values on the vertical axis indicate the results of the selected function at the corresponding time.

The representation of a vector in the chart is referred to a an *element* and is identified by a name, the *legend*. The legends are displayed in a separate window.

The elements in a line diagram are displayed in different colors and the values compiled are indicated by different symbols, which makes it easier to distinguish between the elements in the diagram. The same colors and symbols are assigned to the legends in the legend window.

An element can be displayed in three ways:

normal:	as described
hidden:	the element is displayed in the background color (white) and is therefore invisible
active:	the element is displayed in blue.

Window elements and working in the window

There are three ways of modifying the graph (line diagram):

– operations performed in the legend window (legend operations)
(the mouse cursor is located in the legend window)

– operations performed on the actual line diagram (diagram operations)
(the mouse cursor is located in the graphics window)

– operations provided by a pop-up menu (menu operations)
(press and hold down the right mouse button with the mouse cursor located in the window containing the diagram, move the cursor to the desired item on the pop-up menu and release the button).

Legend operations

– Move the mouse cursor onto the legend:
The corresponding element becomes active (blue).

– Click on the legend with the left mouse button:
The status of the element changes from normal to hidden or vice versa.

Diagram operations

Zoom:
Press and hold down the left mouse button in the upper left corner of the rectangular area that you want to enlarge. Drag the mouse cursor to the lower right corner of the rectangle and release the button. The contents of the selected rectangle now fill the whole graphics window. While you are holding the mouse button down, an outline of the rectangle is displayed in the graphics window and the zoom level is displayed in the upper left corner of the graphics window. The reverse operation is a menu operation.

Output coordinates:
Move the mouse cursor into the vicinity of a symbol. The symbol turns blue and its coordinates are displayed.

Menu operations

Zoom Back:
The last zoom operation is canceled. The new zoom level is displayed in the upper left corner of the graphics window.

Scale MinMax:
The elements are drawn in such a way that their lowest point is on the bottom edge of the graphic and their highest point on the upper edge. This means that each element fills the entire vertical area.

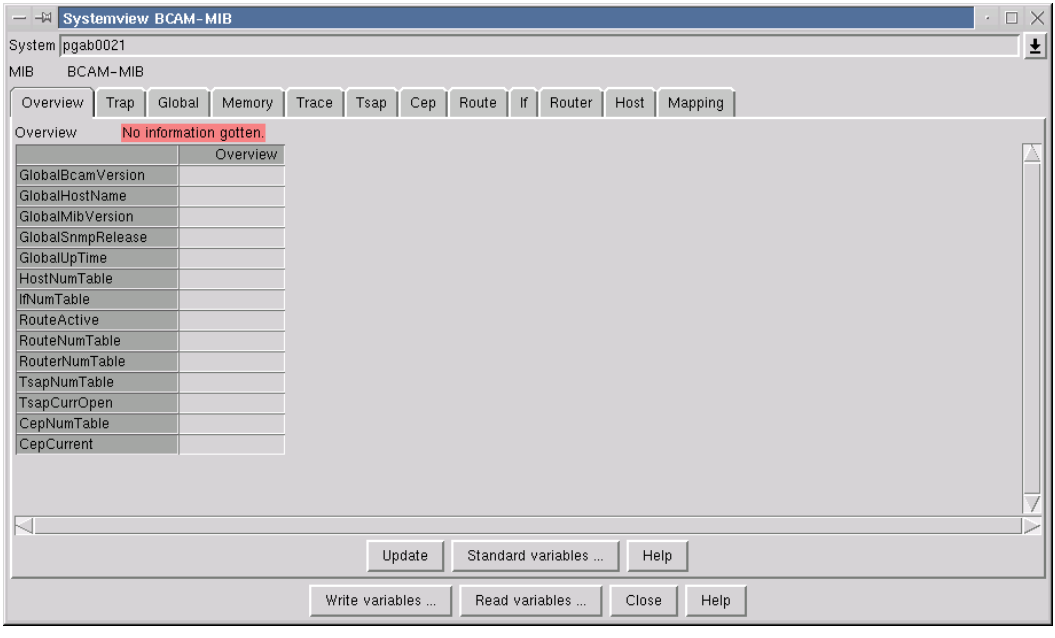
Scale ZeroMax:
As above but the coordinate of the bottom edge is zero.

LegendOnOff:
hides and displays the legend window.

Save Diagram:
saves the graphic in a postscript file.

Help:
displays the appropriate help text.

Obtaining information about the MIB of the monitored system



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Overview

This window provides you with general information about the BCAM-MIB of the system. It is not assigned to any particular MIB group.

Window elements and working in the window

Table

The first column contains the names of the MIB table variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a standard set which can be changed if necessary (see below).

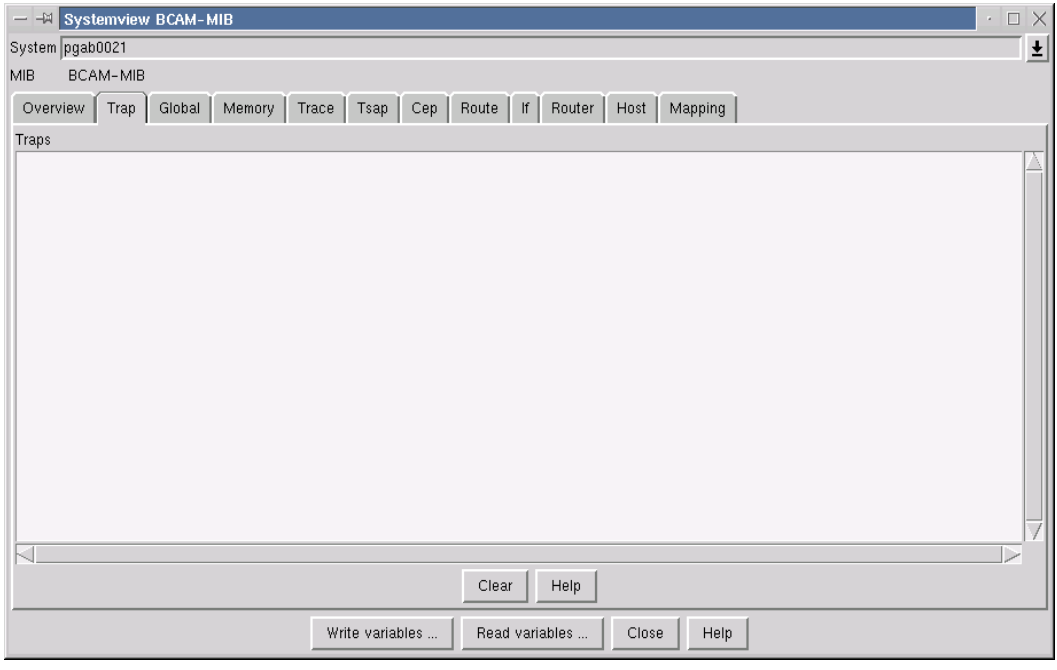
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Display all the traps received for the MIB and system being viewed



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Trap

This window is used to display all of the traps received for the MIB and system being viewed from the time the window was generated.

Window elements

The text field *Traps*
shows the traps received using the following format:

"Trap received from SYMBNAME (XXX.XXX.XXX.XXX) YY.YY.YYYY ZZ:ZZ:ZZ"

SYMBNAME .	The symbolic name of the system or empty.
XXX.XXX.XXX.XXX.	The IP address from which the trap was received.
YY.YY.YYYY .	The date with the format day, month, year.
ZZ:ZZ:ZZ .	The time with the format hours, minutes, seconds.

This is followed by the MIB variables received with the trap in the following format:

"VVVVV : WWWWWW".

VVVVV: The symbolic MIB variable name if known.

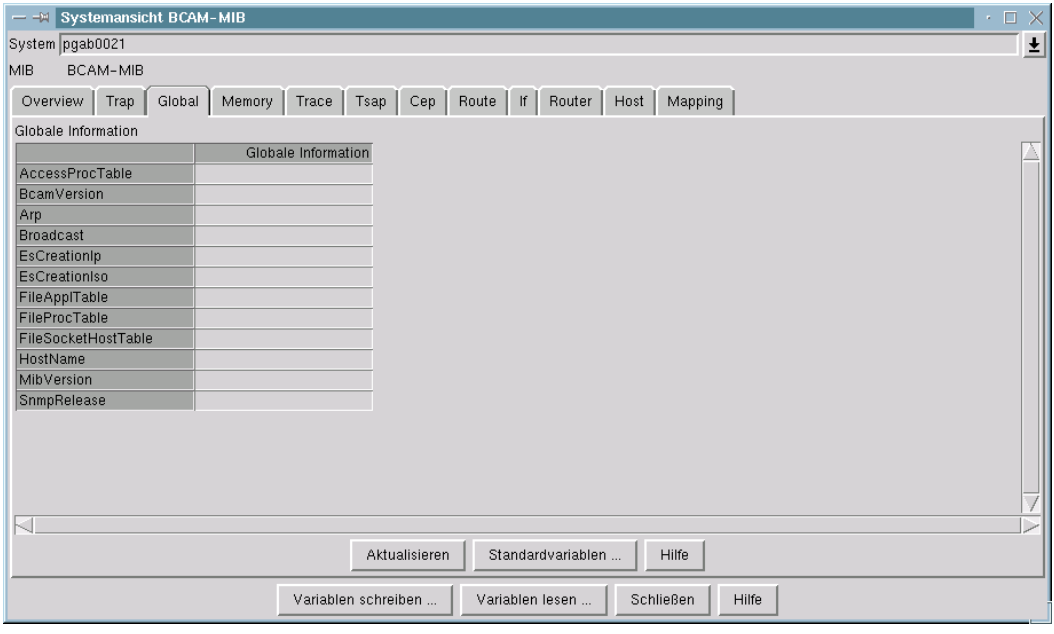
WWWWW: The value of the MIB variable.

Buttons

Clear
deletes all the traps entered in the text field.

Help
displays the appropriate help text.

Obtaining information about the global group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Global

This window provides information about the *global* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

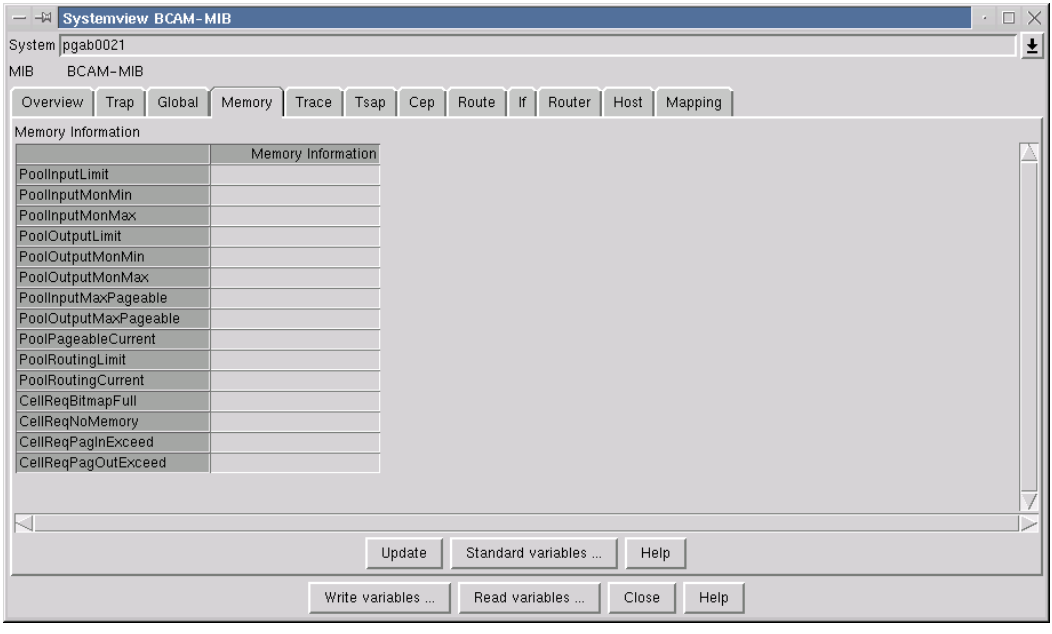
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the memory group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Memory

This window provides information about the *memory* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

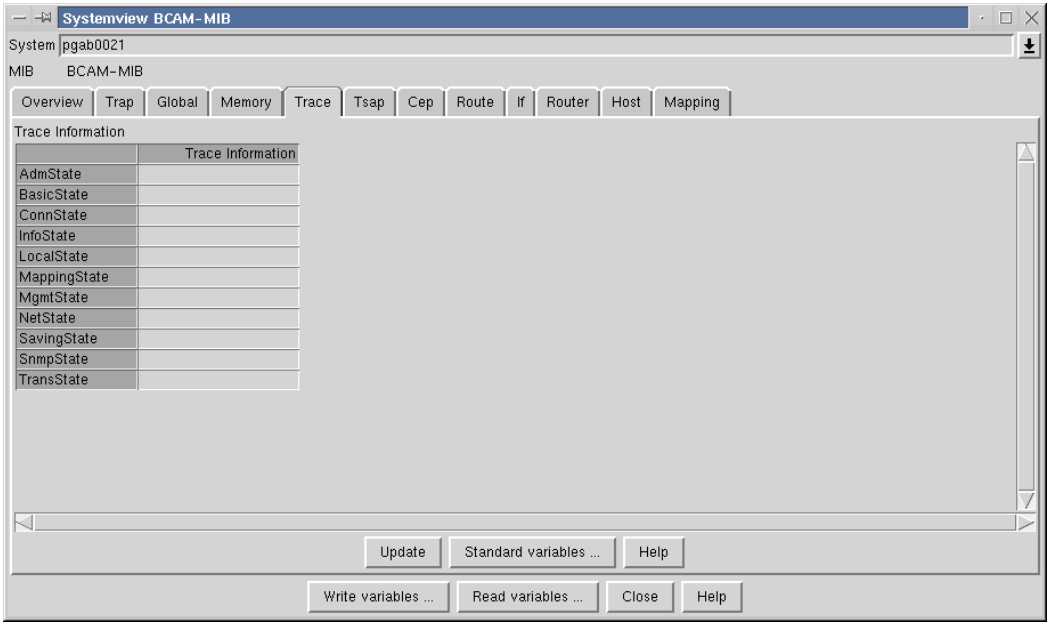
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the trace group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Trace

This window provides information about the *trace* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

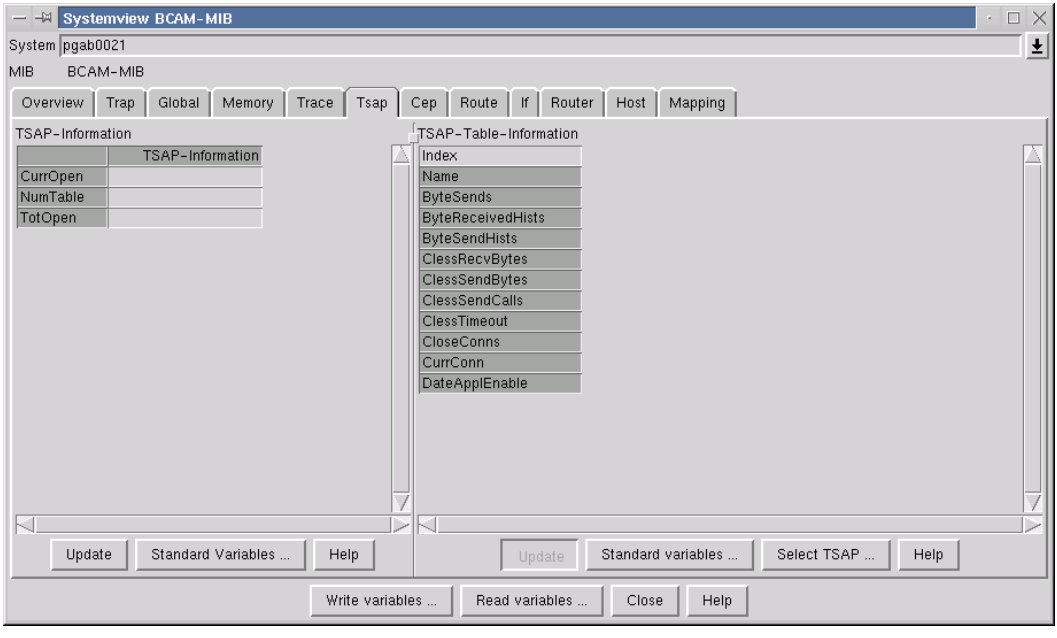
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the application group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Tsap

This window provides information about the *application* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

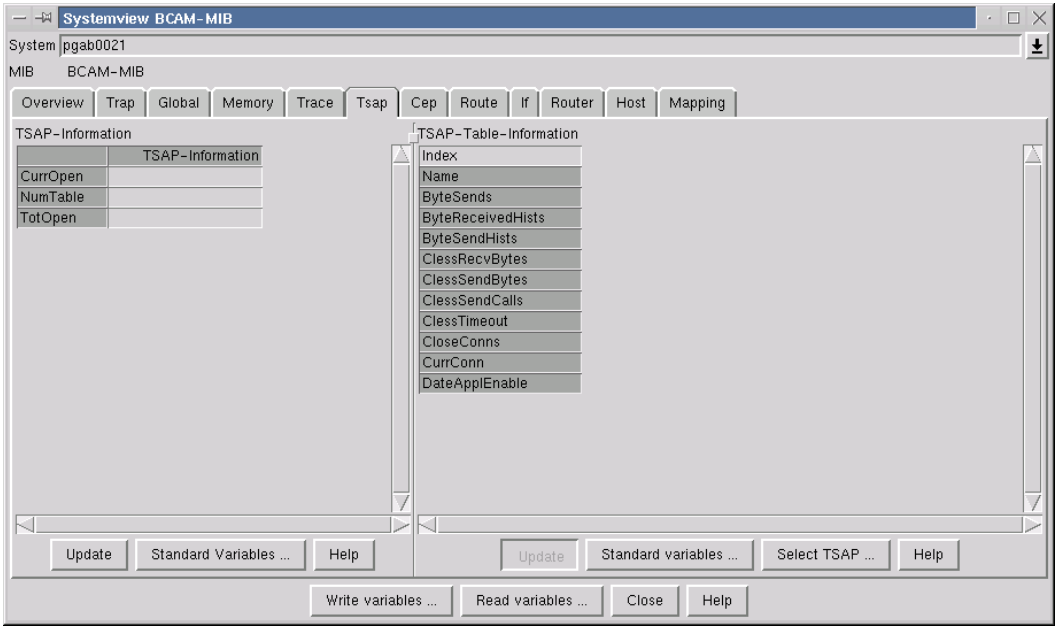
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the application table



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Tsap

This window provides information about the *application table* of the BCAM-MIB.
Each MIB table instance describes a communications application (transport service access point, TSAP).

A TSAP is uniquely identified by its table index. Table instances with the same name but different indices indicate that a communications application has been opened more than once. Only one of these TSAPs can have an “open” status.

Information about selected MIB table instances is displayed in the table in this window.

Window elements and working in the window

Table

The first column contains the names of the MIB table variables to be displayed.
The other columns contain the corresponding values (or blanks if no value yet exists). The first row contains the indices of the MIB table instances.

The background color indicates whether the table information was received during the last update:

– Yellow indicates that no information was received for the relevant MIB table instance.

– Orange indicates that the relevant instance probably no longer exists.

Selection of the variables is determined by a table-specific standard set which can be modified if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

You select a MIB table instance by clicking on the relevant column in the table with the left mouse button. You may select more than one instance. If you click on your selection again, this selection is cancelled.

Pop-up menu

Menu items:

Delete selected columns
: deletes the selected columns in the table.

Get variables of selected columns
: updates the selected columns.

Show connections
: allows you to search for the connection end points assigned to the selected applications. The results are displayed in a second table. A separate selection window allows you to modify the selection of the connections by specifying additional search criteria (which can also originate from other MIB tables).

Show routes
: allows you to search for the routes assigned to the selected applications. The results are displayed in a second table. A separate selection window allows you to modify the route selection by specifying additional search criteria (which can also originate from other MIB tables).

Show interfaces
: allows you to search for the interfaces used by the routes used by connections assigned to the selected applications. A separate selection window allows you to modify the interface selection by specifying additional search criteria (which can also originate from other MIB tables).

Search function
: allows you to search for MIB table instances using search criteria from different MIB tables or by selecting a previously defined search set. The results are displayed in the table area.

Graphic functions of selected columns
: Graphic functions for the selected instances are provided in a new graphics window.

Graphic functions
: allows you to define criteria for the search for MIB tables instances from different MIB tables or to select a previously defined search set. Graphic functions for the appropriate instances are provided in a new graphics window.

Buttons

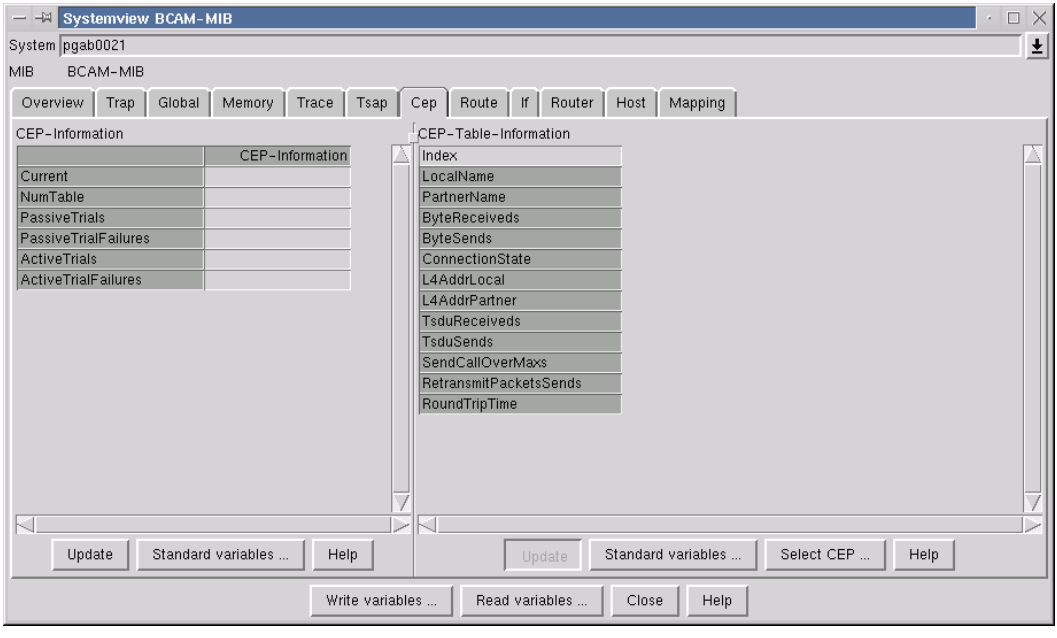
Update
: updates the values of all the columns in the table. If no MIB table instance has been selected, this button is inactive.

Standard variables
: allows you to modify the MIB variables to be displayed for this MIB table.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the other columns.

Select TSAP
: allows you to search for instances of the TSAP-MIB table using criteria from the TSAP-MIB table. Applications found are displayed in the table.

Help
: displays the appropriate help text.

Obtaining information about the connection group



Main BCAM Manager window → System menu → Systemview BCAM-MIB → BCAM-MIB → Cep

This window provides information about the *connection* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

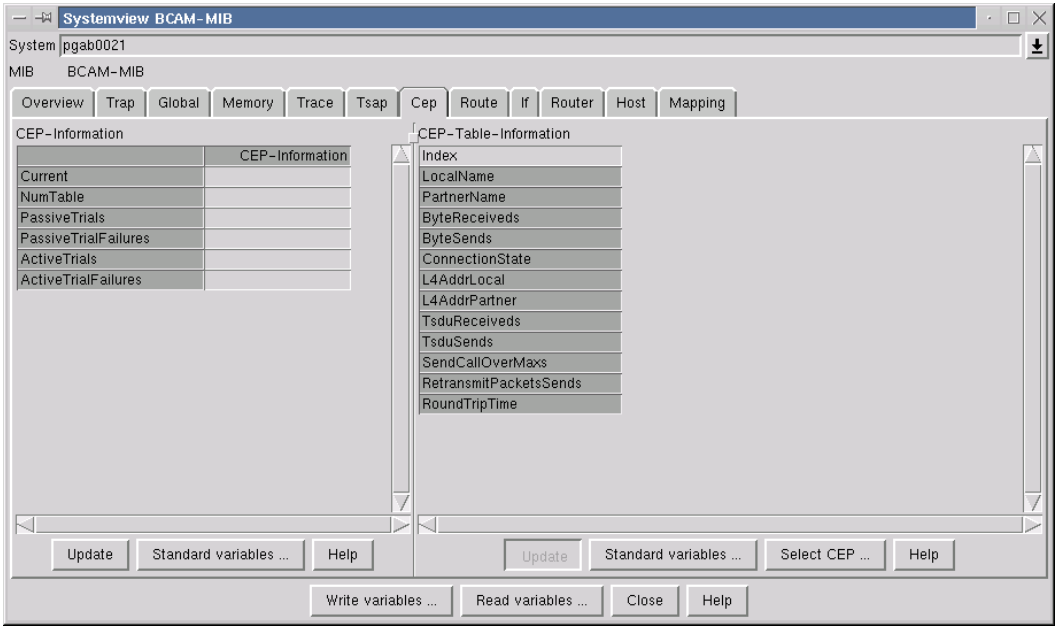
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the connection end points table



Main BCAM Manager window → System menu → Systemview BCAM-MIB → BCAM-MIB → Cep

This window provides information about the *connection endpoint table* of the BCAM-MIB. Each MIB table instance describes a connection endpoint (CEP), e.g. the representative of a connection-oriented communication relationship.

A CEP can be uniquely identified by its table index. Table instances with the same name but different indices indicate that a communication relationship has been set up more than once. If these are not parallel connections, only one CEP can have an “open” status.

Information about the selected MIB table instances are displayed in the table area of this window.

Window elements and working in the window

Table

The first column contains the names of the MIB table variables to be displayed.
The other columns contain the corresponding values (or blanks if no value yet exists). The first row contains the indices of the MIB table instances.

The background color indicates whether the table information was received during the last update:

– Yellow indicates that no information was received for the relevant MIB table instance.

– Orange indicates that the relevant instance probably no longer exists.

Selection of the variables is determined by a table-specific standard set which can be modified if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

You select a MIB table instance by clicking on the relevant column in the table with the left mouse button. You may select more than one instance. If you click on your selection again, this selection is cancelled.

Pop-up menu

Menu items:

Delete selected columns
: deletes the selected columns in the table.

Get variables of selected columns
: updates the selected columns.

Show application
: allows you to search for the application associated with each of the selected CEPs. The results are displayed in a second table window. A separate selection window allows you to modify the selection of the applications by specifying additional search criteria (which can also originate from other MIB tables).

Show routes
: allows you to search for the routes used for each of the selected CEPs. The results are displayed in a second table window. A separate selection window allows you to modify the route selection by specifying additional search criteria (which can also originate from other MIB tables).

Show interfaces
: allows you to search for each of the selected CEPs to locate the interface used by the route. A separate selection window allows you to modify the interface selection by specifying additional search criteria (which can also originate from other MIB tables).

Search function
: allows you to search for MIB table instances using search criteria from different MIB tables or by selecting a previously defined search set.
The results are displayed in the table.

Graphic functions of selected columns
: Graphic functions for the selected instances are provided in a new graphics window.

Graphic functions
: allows you to define criteria for the search for MIB tables instances from different MIB tables or to select a previously defined search set. Graphic functions for the appropriate instances are provided in a new graphics window.

Buttons

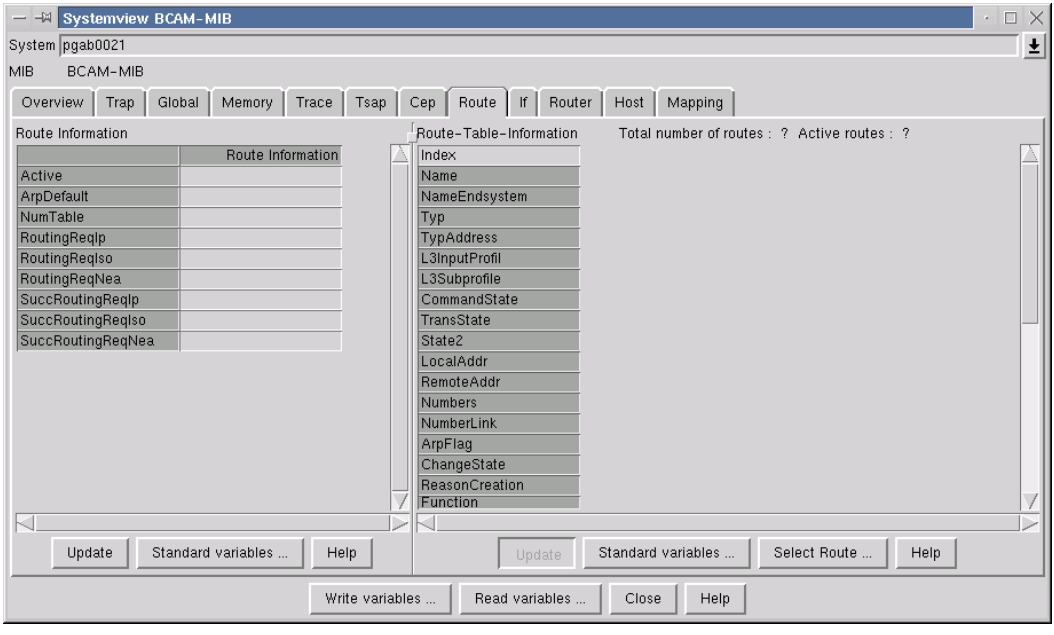
Update
: updates the values of all the columns in the table. If no MIB table instance has been selected, this button is inactive.

Standard variables
: allows you to modify the MIB variables to be displayed for this MIB table.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the other columns.

Select CEP
: allows you to search for instances of the CEP-MIB table using criteria from the CEP-MIB table. CEPs found are displayed in the table.

Help
: displays the appropriate help text.

Obtaining information about the route group



Main BCAM Manager Window → System menu → Systemview for MIB → BCAM-MIB → Route

This window provides you with information about the *route* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

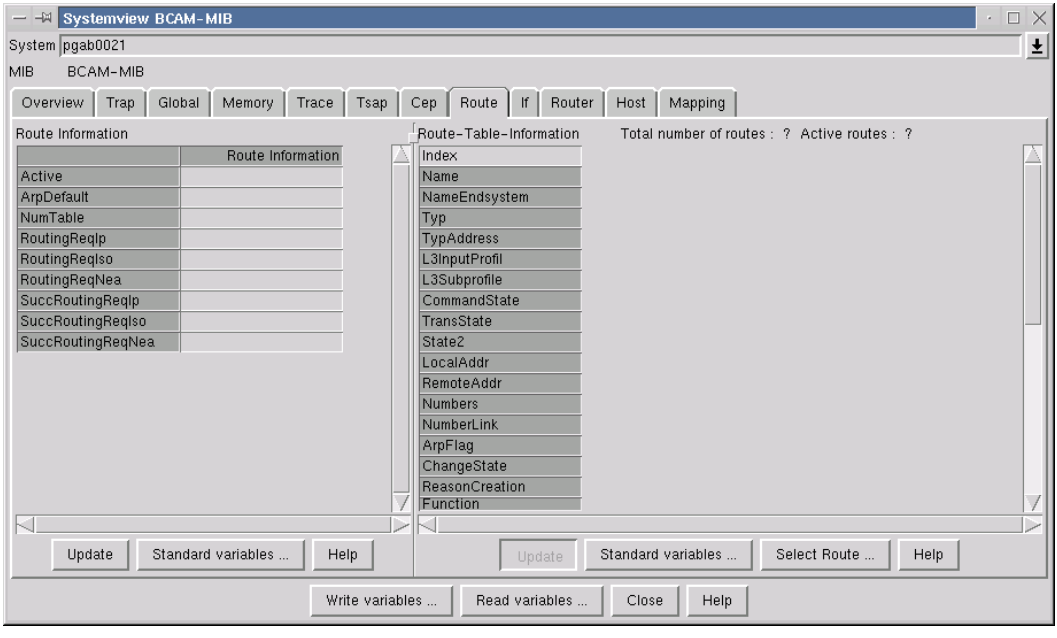
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the route table



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Route

This window provides information about the *route* table of the BCAM-MIB. Each MIB table instance describes a route, which basically represents the path from one system to another. To be more exact, a route is defined by a pair of network addresses, the local and the remote address, which are used as either the source address or target address (depending on the direction) for datagrams exchanged via the network.

Each route is assigned an interface, which provides access to the network.
A route can be assigned auxiliary routes such as, for example, the route to the *first-hop* router. A route is uniquely identified by its MIB table index or its name. It is also possible to have several routes to another system.

Information on the selected MIB table instances is displayed in the table area of the window.

Window elements and working in the window

Table

The first column contains the names of the MIB table variables to be displayed.
The other columns contain the corresponding values (or blanks if no value yet exists). The first row contains the indices of the MIB table instances.

The background color indicates whether the table information was received during the last update:

– Yellow indicates that no information was received for the relevant MIB table instance.

– Orange indicates that the relevant instance probably no longer exists.

Selection of the variables is determined by a table-specific standard set which can be modified if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

You select a MIB table instance by clicking on the relevant column in the table with the left mouse button. You may select more than one instance. If you click on your selection again, this selection is cancelled.

Pop-up menu

Menu items:

Delete selected columns
: deletes the selected columns in the table.

Get variables of selected columns
: updates the selected columns.

Auxiliary routes and interface
: Only one route can be selected. The selected route, any auxiliary routes and the interfaces are displayed in a second window.

Show connections
: allows you to search for the CEPs assigned to the selected routes. The results are displayed in a second table window. A separate selection window allows you to modify the selection by specifying additional search criteria (which can also originate from other MIB tables).

Show applications
: allows you to search for applications which possess connections assigned to one of the selected routes. The applications found are displayed in a second table window. A separate selection window allows you to modify the selection of the applications by specifying additional search criteria (which can also originate from other MIB tables).

Search function
: allows you to search for MIB table instances using search criteria from different MIB tables or by selecting a previously defined search set.
The results are displayed in the table.

Graphic functions of selected columns
: Graphic functions for the selected instances are provided in a new graphics window.

Graphic functions
: allows you to define criteria for the search for MIB tables instances from different MIB tables or to select a previously defined search set. Graphic functions for the appropriate instances are provided in a new graphics window.

Buttons

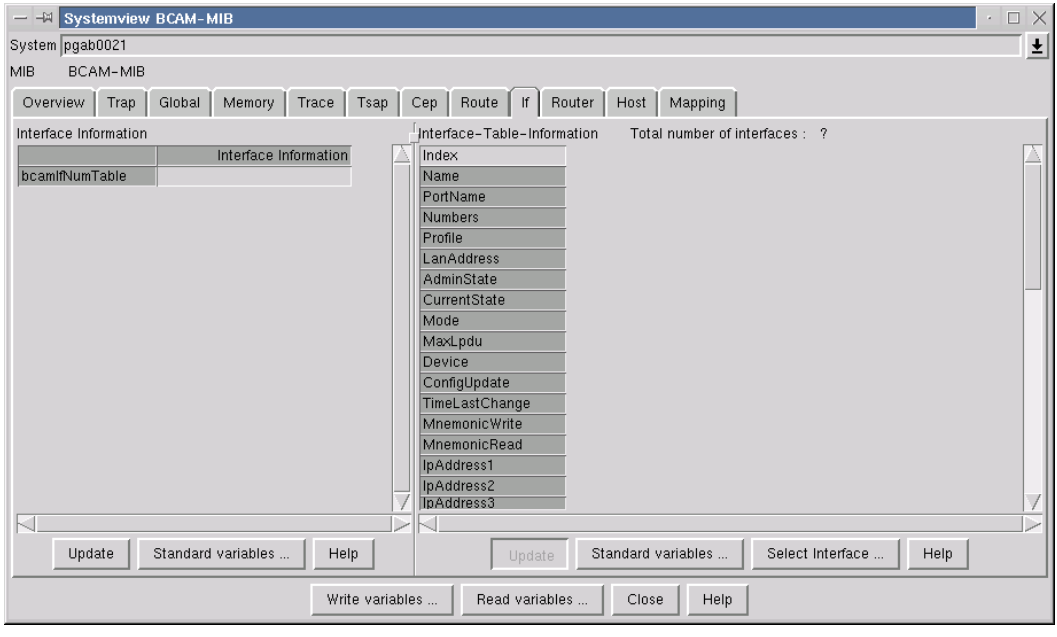
Update
: updates the values of all the columns in the table.
If no MIB table instance has been selected, this button is inactive.

Standard variables
: allows you to modify the MIB variables to be displayed for this MIB table.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the other columns.

Select Route
: allows you to search for instances of the route MIB table using criteria from the route MIB table. Routes found are displayed in the table area.

Help
: displays the appropriate help text.

Obtaining information about the interface group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → If

This window provides information about the *interface* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

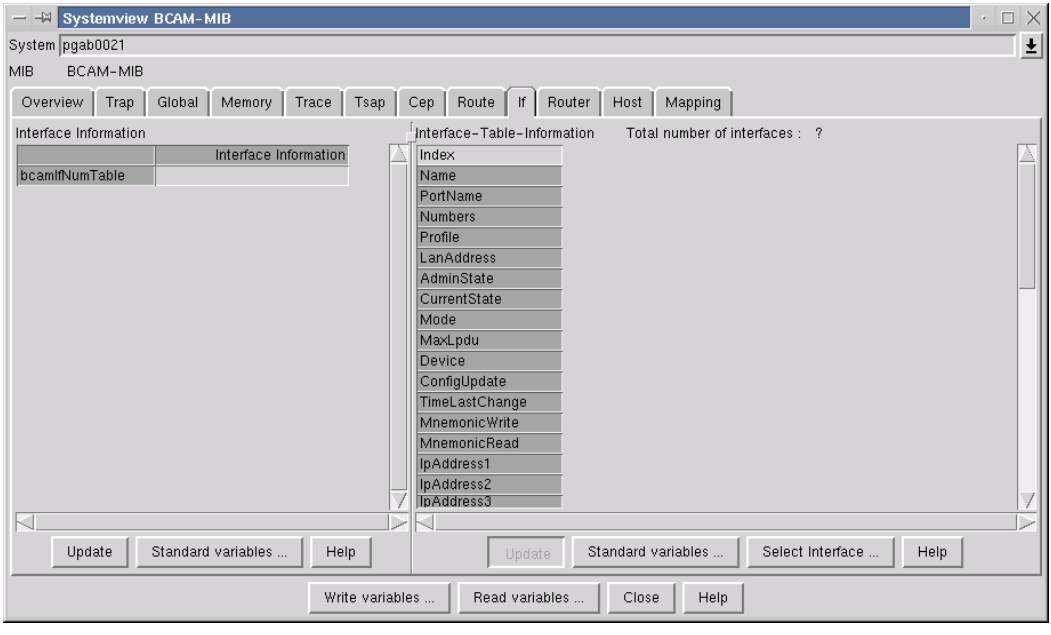
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the interface table



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → If

This window provides information about the *interface* table of the BCAM-MIB. Each MIB table instance describes an interface which provides physical access to the network.

An interface is uniquely identified by its MIB table index or its name. More than one interfaces can be connected to the same LAN subnetwork. In this case, they all have the same port name.

Information about the selected MIB table instances are displayed in the table area of the window.

Window elements and working in the window

Table

The first column contains the names of the MIB table variables to be displayed. The other columns contain the corresponding values (or blanks if no value yet exists). The first row contains the indices of the MIB table instances.

The background color indicates whether the table information was received during the last update:

– Yellow indicates that no information was received for the relevant MIB table instance.

– Orange indicates that the relevant instance probably no longer exists.

Selection of the variables is determined by a table-specific standard set which can be modified if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

You select a MIB table instance by clicking on the relevant column in the table with the left mouse button. You may select more than one instance. If you click on your selection again, this selection is cancelled.

Pop-up menu

Menu items:

Delete selected columns
: deletes the selected columns in the table.

Get variables of selected columns
: updates the selected columns.

Show routes
: allows you to search for the routes assigned to the selected interfaces. The results are displayed in a second table window. A separate selection window allows you to modify the route selection by specifying additional search criteria (which can also originate from other MIB tables).

Show connections
: allows you to search for the CEPs assigned to the selected interfaces. The results are displayed in a second table window. A separate selection window allows you to modify the selection by specifying additional search criteria (which can also originate from other MIB tables).

Show applications
: allows you to search for applications which possess connections assigned to one of the selected interfaces. The applications found are displayed in a second table window. A separate selection window allows you to modify the selection of the applications by specifying additional search criteria (which can also originate from other MIB tables).

Search function
: allows you to search for MIB table instances using search criteria from different MIB tables or by selecting a previously defined search set. The results are displayed in the table.

Graphic functions of selected columns
: Graphic functions for the selected instances are provided in a new graphics window.

Graphic functions
: allows you to define search criteria from different MIB tables to be used to search for MIB table instances or to select a previously defined search set. Graphic functions for the appropriate instances are provided in a new graphics window.

Buttons

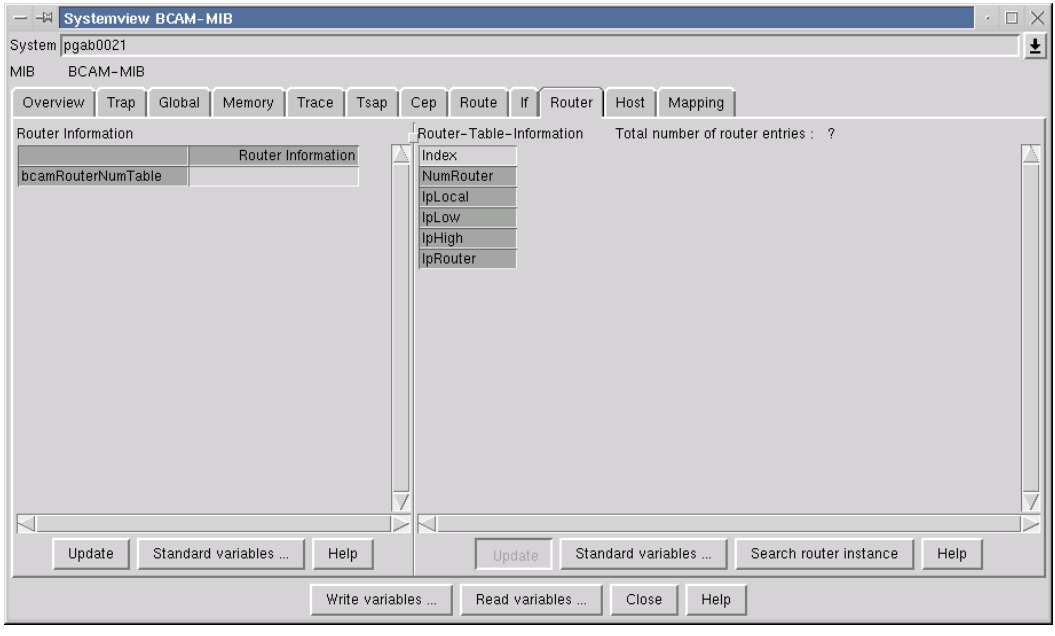
Update
: updates the values of all the columns in the table.
If no MIB table instance has been selected, this button is inactive.

Standard variables
: allows you to modify the MIB variables to be displayed for this MIB table.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the other columns.

Select Interface
: allows you to search for instances of the interface MIB table using criteria from the interface MIB table. Interfaces found are displayed in the table area.

Help
: displays the appropriate help text.

Obtaining information about the router group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Router

This window provides information about the *router* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

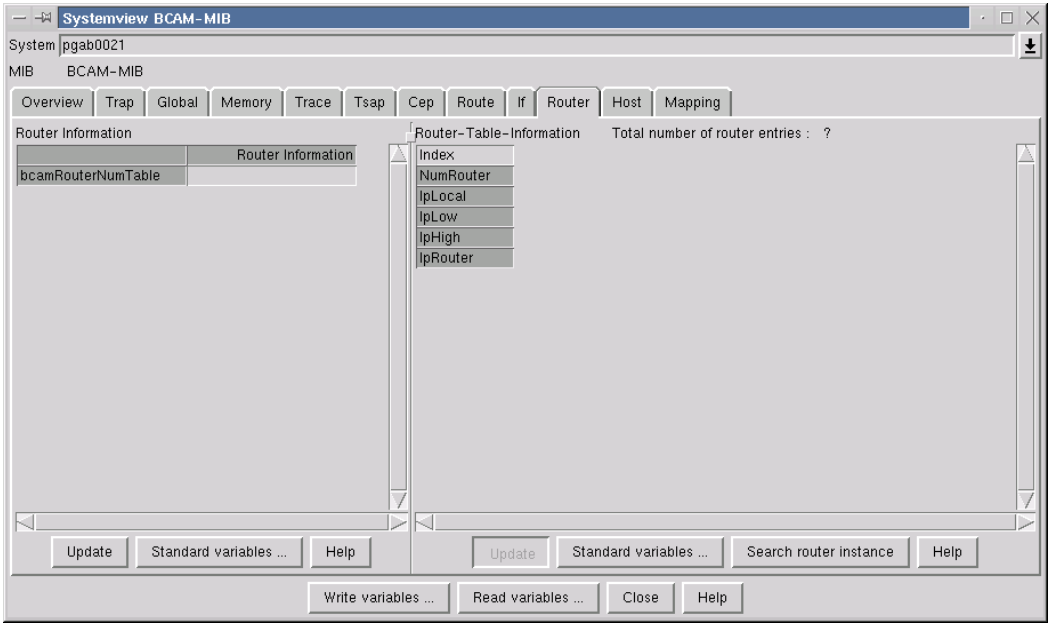
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the router table



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Router

This window provides information about the *router* table of the BCAM-MIB.
Each MIB table instance describes either the main router or a virtual router.

A router is uniquely identified by its MIB table index or its name.

Information about the selected MIB table instances are displayed in the table area of the window.

Window elements and working in the window

Table

The first column contains the names of the MIB table variables to be displayed. The other columns contain the corresponding values (or blanks if no value yet exists). The first row contains the indices of the MIB table instances.

The background color indicates whether the table information was received during the last update:

– Yellow indicates that no information was received for the relevant MIB table instance.

– Orange indicates that the relevant instance probably no longer exists.

Selection of the variables is determined by a table-specific standard set which can be modified if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

You select a MIB table instance by clicking on the relevant column in the table with the left mouse button. You may select more than one instance. If you click on your selection again, this selection is cancelled.

Pop-up menu

Menu items:

Delete selected columns
: deletes the selected columns in the table.

Get variables of selected columns
: updates the selected columns.

Show router route
: Only one instance can be selected. The corresponding route to the route is displayed in a second window.

Search function
: allows you to search for MIB table instances using search criteria from different MIB tables or by selecting a previously defined search set. The results are displayed in the table.

Graphic functions
: allows you to define search criteria from different MIB tables to be used to search for MIB table instances or to select a previously defined search set. Graphic functions for the appropriate instances are provided in a new graphics window.

Buttons

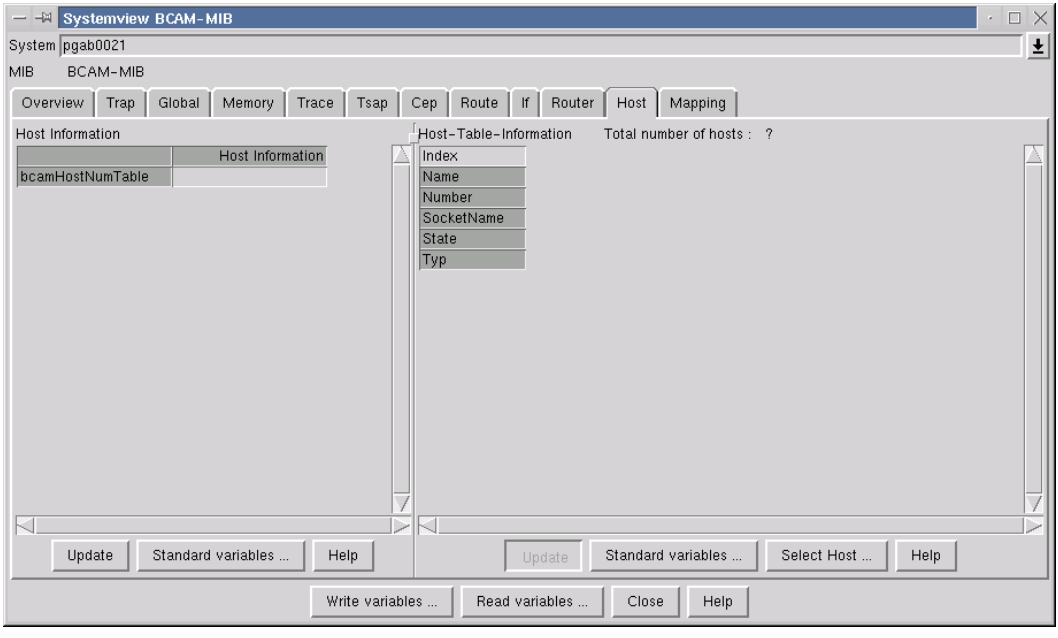
Update
: updates the values of all the columns in the table.
If no MIB table instance has been selected, this button is inactive.

Standard variables
: allows you to modify the MIB variables to be displayed for this MIB table.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the other columns.

Search router instance
: allows you to search for instances of the router MIB table using criteria from the router MIB table. Router instances found are displayed in the table area.

Help
: displays the appropriate help text.

Obtaining information about the host group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Host

This window provides information about the *host* group of the BCAM-MIB.

Window elements and working in the window

Table

The first column contains the names of the MIB group variables to be displayed.
The second column contains the corresponding values (or blanks if no value yet exists).

Selection of the variables is determined by a group-specific standard set which can be changed if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

Pop-up menu

Menu item *Graphic functions*
: Graphic functions for the group are provided in a new graphics window.

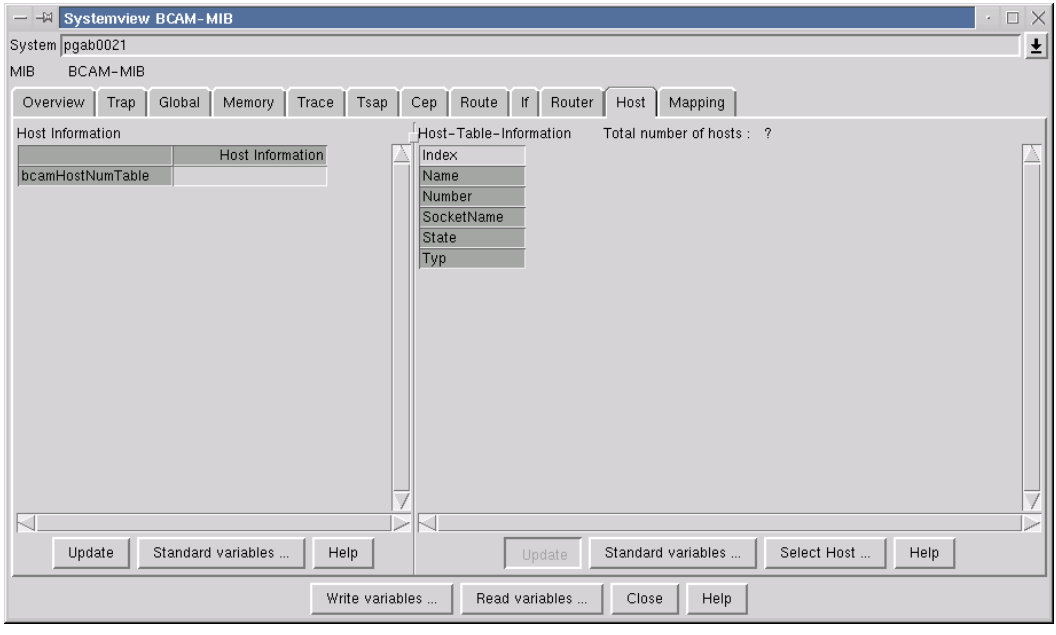
Buttons

Update
: updates the values in the second column of the table.

Standard variables
: allows you to change the MIB variables to be displayed for this MIB group.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the second column.

Help
: displays the appropriate help text.

Obtaining information about the host table



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Host

This window provides information about the *host* table of the BCAM-MIB. Each MIB table instance describes either the main host or a virtual host.

A host is uniquely identified by its MIB table index or its name.

Information about the selected MIB table instances are displayed in the table area of the window.

Window elements and working in the window

Table

The first column contains the names of the MIB table variables to be displayed. The other columns contain the corresponding values (or blanks if no value yet exists). The first row contains the indices of the MIB table instances.

The background color indicates whether the table information was received during the last update:

– Yellow indicates that no information was received for the relevant MIB table instance.

– Orange indicates that the relevant instance probably no longer exists.

Selection of the variables is determined by a table-specific standard set which can be modified if necessary (see below).

A pop-up menu from which you can select various functions is displayed if you position the mouse cursor on the table and press and hold down the right mouse button.

You select a MIB table instance by clicking on the relevant column in the table with the left mouse button. You may select more than one instance. If you click on your selection again, this selection is cancelled.

Pop-up menu

Menu items:

Delete selected columns
: deletes the selected columns in the table.

Get variables of selected columns
: updates the selected columns.

Show connections
: allows you to search for the CEPs assigned to the selected hosts. The results are displayed in a second table window. A separate selection window allows you to modify the selection by specifying additional search criteria (which can also originate from other MIB tables).

Show applications
: allows you to search for the applications (TSAPs) assigned to the selected hosts. The results are displayed in a second table window. A separate selection window allows you to modify the selection of the applications by specifying additional search criteria (which can also originate from other MIB tables).

Search function
: allows you to search for MIB table instances using search criteria from different MIB tables or by selecting a previously defined search set. The results are displayed in the table.

Graphic functions of selected columns
: Graphic functions for the selected instances are provided in a new graphics window.

Graphic functions
: allows you to define search criteria from different MIB tables to be used to search for MIB table instances or to select a previously defined search set. Graphic functions for the appropriate instances are provided in a new graphics window.

Buttons

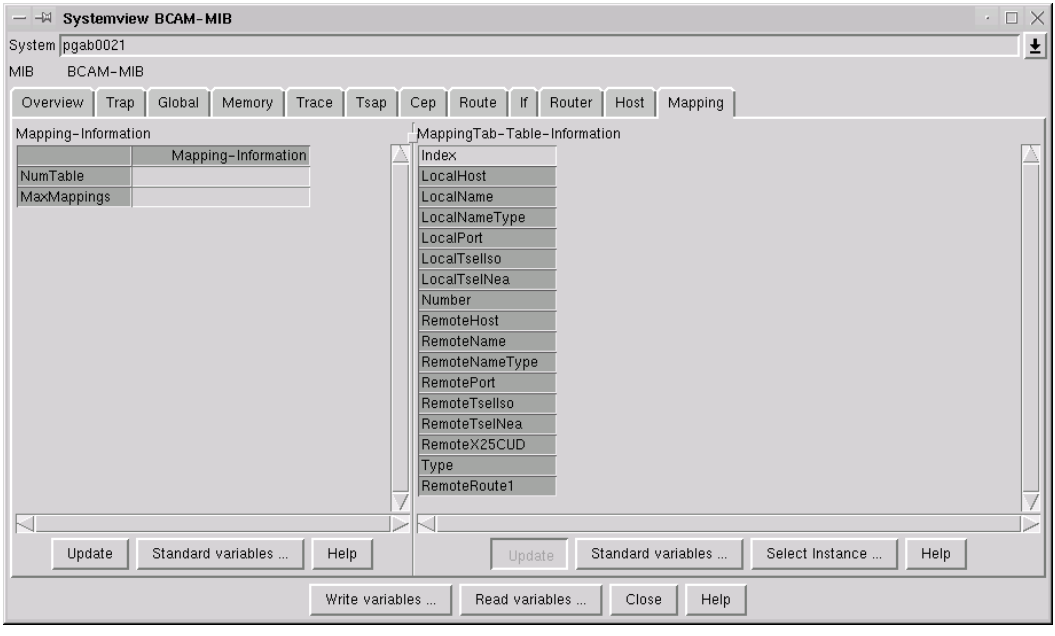
Update
: updates the values of all the columns in the table.
If no MIB table instance has been selected, this button is inactive.

Standard variables
: allows you to modify the MIB variables to be displayed for this MIB table.
If you make a change, the names of the selected variables are entered in the first column of the table, and the values of the variables are preset to blanks in the other columns.

Select Host
: allows you to search for instances of the host MIB table using criteria from the host MIB table. Hosts found are displayed in the table area.

Help
: displays the appropriate help text.

Obtaining information about the mapping group



Main BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Mapping

This window provides information about the *mapping* group of the BCAM-MIB.

Working in the window

A pop-up menu is displayed when you press and hold down the right mouse button.

Window elements

Table

The first column of the table contains the print names of the MIB group variables to be displayed. The values from the last query are shown in the second column.

Pop-up menu

The pop-up menu contains the menu item *Graphic functions* which opens the window for the graphic display functions defined for this MIB group.

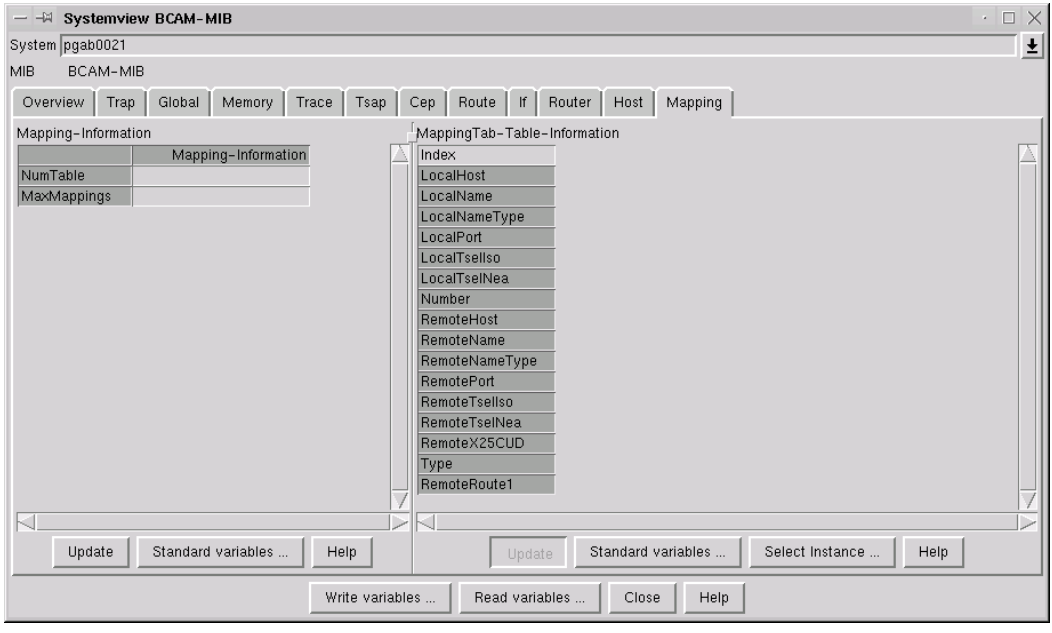
Buttons

Update
: queries the MIB variables in the table and displays their current values.

Standard variables
: opens a dialog for defining the MIB group variables to be displayed in the table.

Help
: displays the appropriate help text.

Obtaining information about the mapping table



BCAM Manager window → System menu → Systemview for MIB → BCAM-MIB → Mapping

This window provides information about the *mapping* table of the BCAM-MIB.

Working in the window

A pop-up menu is displayed when you press and hold down the right mouse button.
You select or deselect columns in the table by clicking on the relevant column using the left mouse button.

Window elements

Table

The first column in the table contains the print names of the MIB table variables to be displayed. The other columns contain the table instances which were queried last.

Pop-up menu

You activate the pop-up menu which contains the following menu items when you press and hold down the right mouse button:

Delete selected columns
: deletes the selected MIB table instances.

Get variables of selected columns
: queries the MIB variables that are included in the table for the selected MIB tables instances and displays their current values.

Graphic functions of selected columns
: opens the window for displaying the graphic functions defined for this MIB table using the marked columns.

Graphic functions
: opens the dialog used to search for and display the graphic functions defined for this MIB table.

Buttons

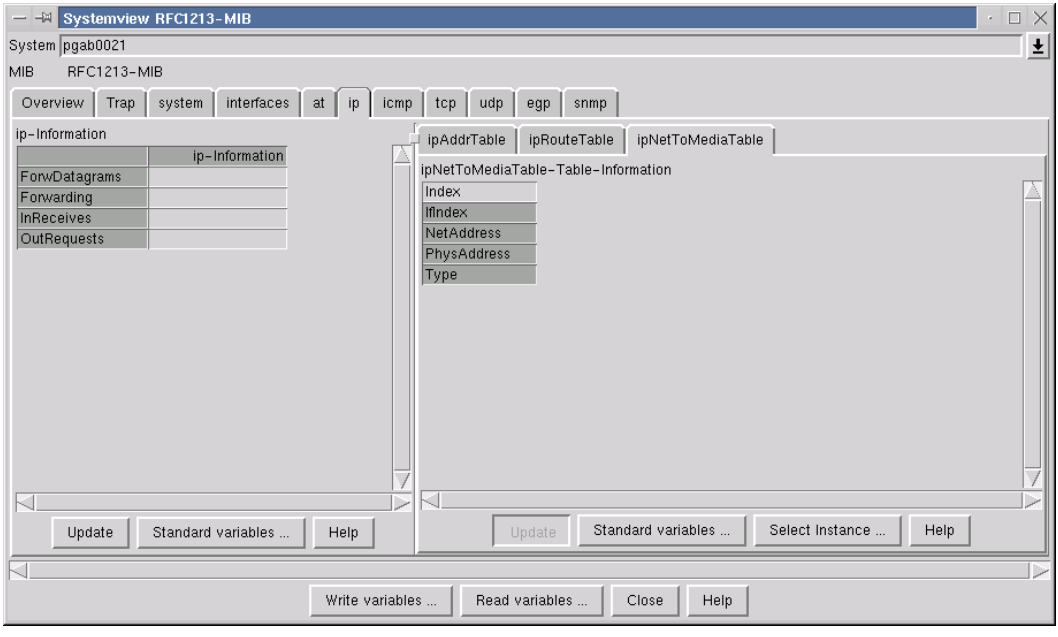
Update
: queries the MIB variables in the table for all MIB table instances and displays their current values.

Standard variables
: opens a dialog for defining the MIB group variables to be displayed in the table.

Select Instance
: opens a dialog for searching for table instances.

Help
: displays the appropriate help text.

4.2.3 Systemview of another MIB: displaying the standard variables of the MIB group being viewed



Main BCAM Manager window ⟶ System menu ⟶ Systemview for MIB ⟶ RFC1213-MIB

The standard variables of the MIB group being viewed are displayed in the table on the left.

Working in the window

A pop-up menu is displayed when you press and hold down the right mouse button.

Window elements

Table

The first column of the table contains the print names of the MIB group variables to be displayed. The values from the last query are shown in the second column.

Pop-up menu

The pop-up menu contains the menu item *Graphic functions* which opens the window for the graphic display functions defined for this MIB group.

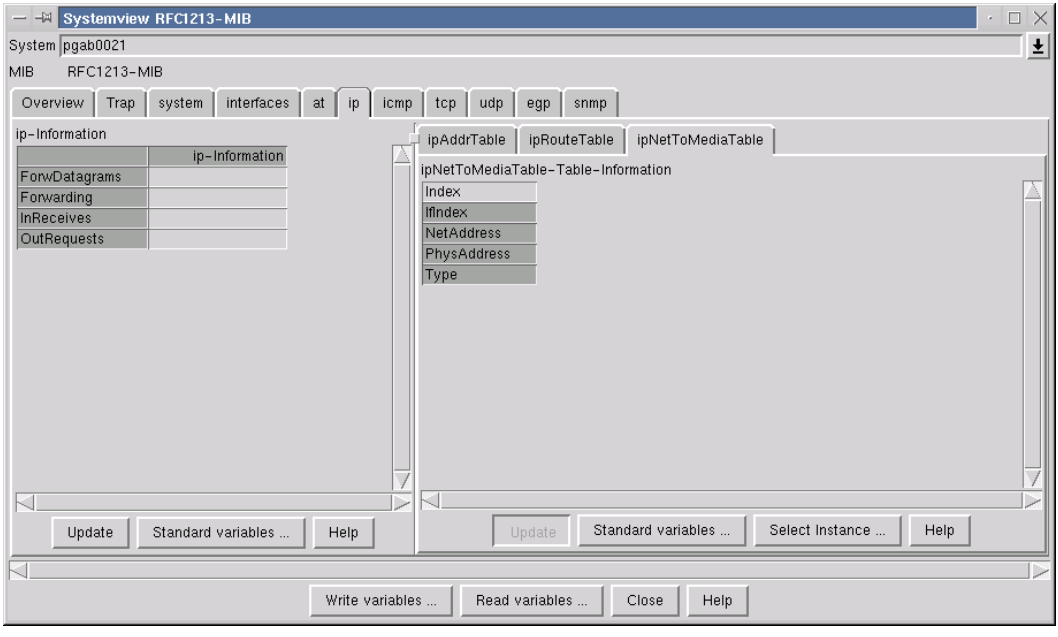
Buttons

Update
: queries the MIB variables in the table and displays their current values.

Standard variables
: opens a dialog for defining the MIB group variables to be displayed in the table.

Help
: displays the appropriate help text.

4.2.4 Systemview of another MIB: displaying the standard variables of the MIB table being viewed



Main BCAM Manager window → System menu → Systemview for MIB → RFC1213-MIB

The standard variables of the MIB table being viewed are displayed in the table on the right.

Working in the window

A pop-up menu is displayed when you press and hold down the right mouse button.
You select or deselect columns in the table by clicking on the relevant column using the left mouse button.

Window elements

Table

The first column in the table contains the print names of the MIB table variables to be displayed. The other columns contain the table instances which were queried last.

Pop-up menu

You activate the pop-up menu which contains the following menu items when you press and hold down the right mouse button:

Delete selected columns
: deletes the selected MIB table instances.

Get variables of the selected columns
: queries the MIB variables that are included in the table for the selected MIB tables instances and displays their current values.

Graphic functions of selected columns
: opens the window for displaying the graphic functions defined for this MIB table using the marked columns.

Graphic functions
: opens the dialog used to search for and display the graphic functions defined for this MIB table.

Buttons

Update
: queries the MIB variables in the table for all MIB table instances and displays their current values.

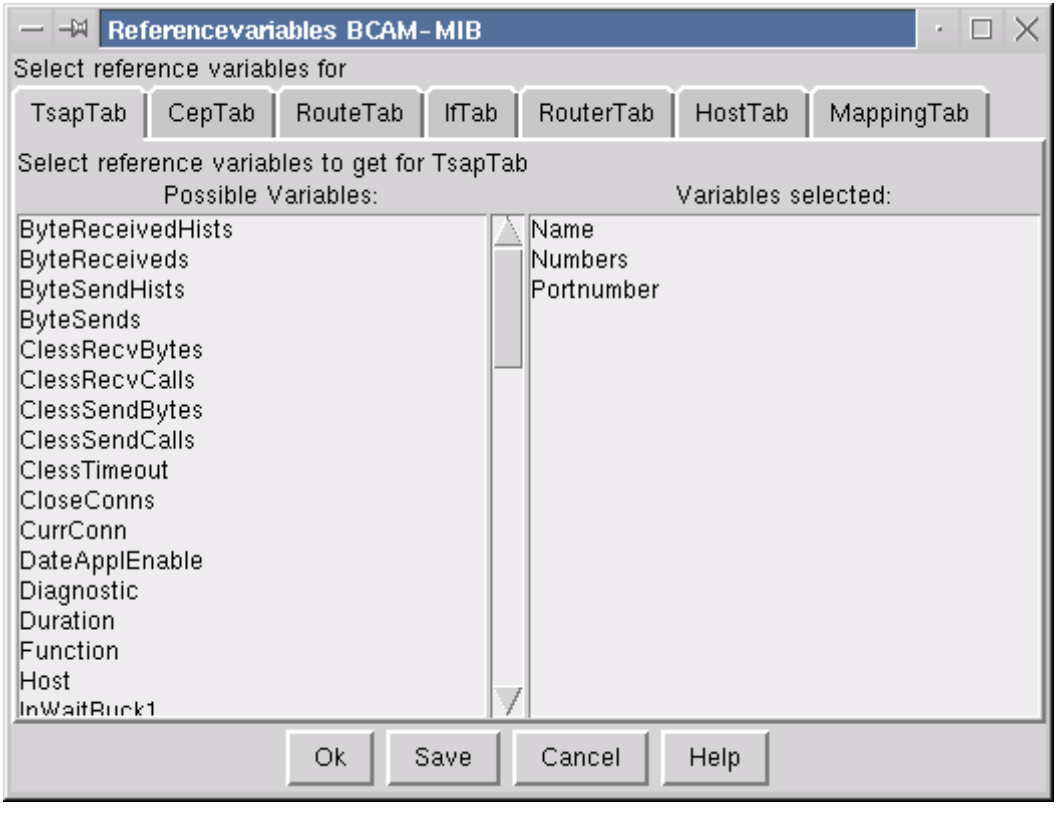
Standard variables
: opens a dialog for defining the MIB group variables to be displayed in the table.

Select Instance
: opens a dialog for searching for table instances.

Help
: displays the appropriate help text.

4.3 The Option menu

4.3.1 Defining reference variables for a MIB table



Main BCAM Manager window → Option menu → Define reference variables for

This window allows you to define the reference variables for a MIB table. The values of the reference variables of a MIB table instance are stored locally during a search operation via the network. If another search is performed which uses one of these reference variables as a search criteria, local storage is searched first. If there is a hit, the values of all the MIB variables specified as search criteria are fetched via the network and compared.

Once the search in the local storage has been completed, a search operation is performed via the network starting with the highest table index in the storage. A search operation via the network is only performed if the search criteria do not include any of the reference variables.

Variables which remain the same for the entire life of an MIB table instance, e.g. name, address, table index, references to table instances of other tables, are suitable for use as reference variables.

Window elements and working in the window

There is a separate tab for each MIB table; the print name of the table is displayed on the appropriate tab. If you select a tab by clicking on it with the left mouse button, it is brought to the front. Each tab contains two lists. The list on the left is used to display the variables from the MIB table which are available for selection, and the list on the right is used to display the variables selected as reference variables. If you click on a variable name with the left mouse button it is moved from one list to the other.

Buttons

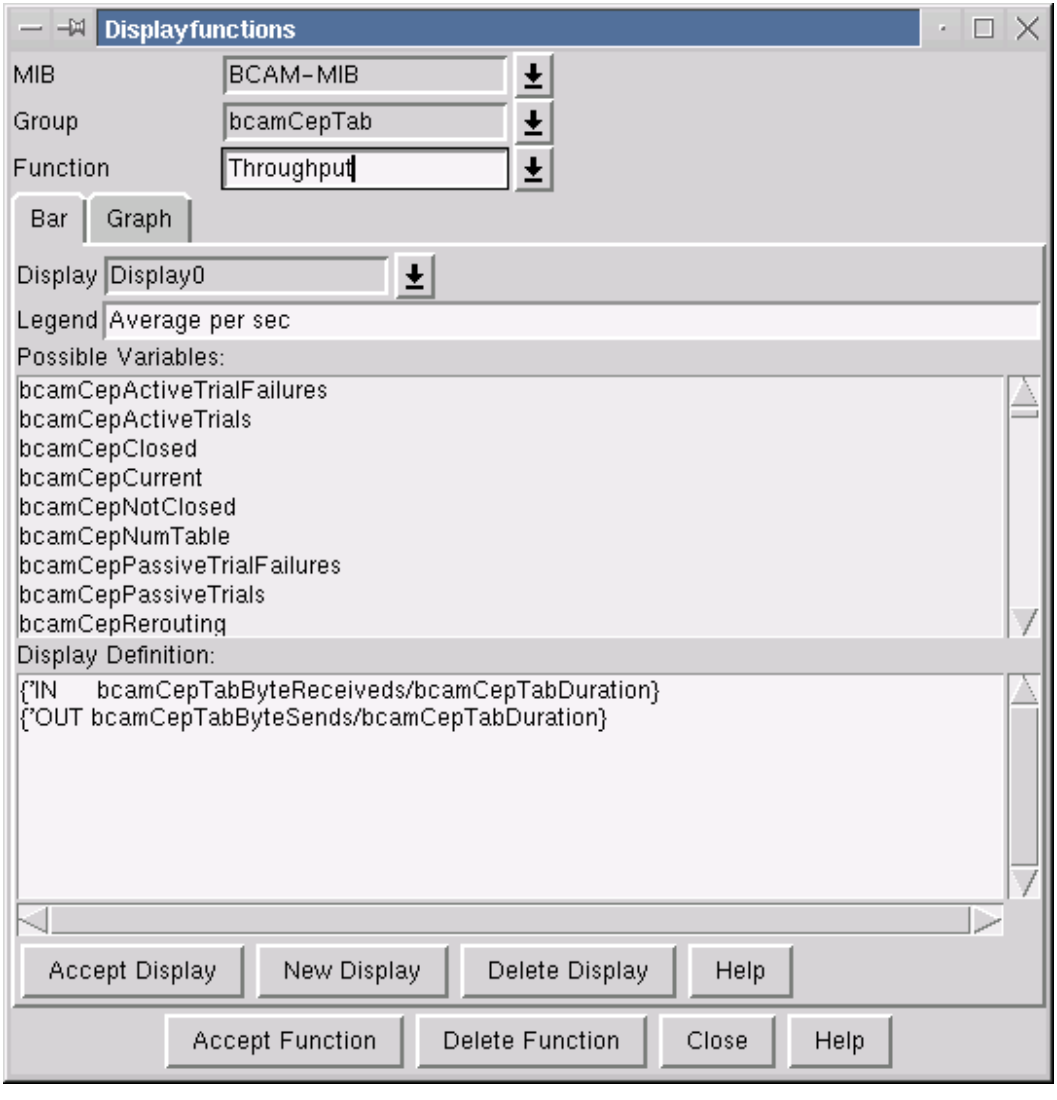
Ok
: accepts the variables currently displayed in the list on the right as temporary reference variables, which are valid until the program is terminated, and closes the window

Save
: saves the variables currently displayed in the list on the right as permanent reference variables.

Cancel
: aborts the selection process and uses the previously valid variables as reference variables.

Help
: displays the appropriate help text.

4.3.2 Defining display functions



Main BCAM Manager window → Option menu → Define display functions for

This window is used to define display functions.
Display functions are used to display periodically queried values as a graphic.

A display function has a name and can contain a line diagram (graph) or bar chart. It is possible to display several curve or bar definitions simultaneously in both graphs and bar charts.

A curve definition comprises a function expression involving one or more MIB variables.

The bar definition comprises one or more pairs of x- and y-coordinates. A bar is displayed for each pair of coordinates. You can also define groups of bars (bars which belong together from a logical point of view).
The x-coordinate can be either a fixed string or the current value of a MIB variable.
The y-coordinate is a function expression involving one or more MIB variables.

If several MIB table instances exist, a curve or bar definition is displayed for each instance.

A function expression can include any of the following operations:

+	addition
-	subtraction
*	multiplication
/	division
(or)	expressions in parentheses

The following functions, which include the name of exactly one MIB variable as an argument, are also available:

DIFF(x) .	Difference between the current value and the previous value of the variable.
GRADIENT(x) .	Difference between the current value and the previous value of the variable divided by the time interval (in seconds) between the two queries.

The following functions return exactly one value, even for more than one instance:

SUM(x).	Sum of the MIB variables over all instances.
GRADIENTSUM(x) .	Sum of GRADIENT(x) over all instances.

Working in the window

You must do the following if you want to create or modify a display function:

1. Select the MIB for which the function is to be defined; the name of this MIB is displayed in the selection field *MIB*.
2. Select the MIB group or MIB table for which the function is to be defined; the name of this MIB group or table is displayed in the selection field *Group*.
3. Enter the name of the function in the selection field *Function* (and press the <Return>-key) or select a function from the selection list.
4. Enter or modify the bar or curve definition (description in the help text for the tabs *Bar* and *Graph*).
5. Click on the button *Accept Function* to confirm.

You must click on the button *Accept Function* once you have defined the bars and curves because otherwise the function definition is deleted.

Window elements

Selection fields

MIB
: contains the current MIB. The corresponding selection list contains all the MIBs that have been loaded.

Group
: contains the current MIB group or MIB table. The corresponding selection list contains all the MIB groups and tables of the current MIB.

Function
: contains the name of the current function for the selected MIB group or MIB table. The corresponding selection list contains all the known functions for the current MIB group or MIB table. If you want to define a new function, you enter the function name in the input field. You must complete your entry by pressing the <Return> key. You can also select an existing function from the selection list and modify it.

Tabs

Bar
: is used to define or modify the individual bar definitions for the diagram. The help text for the tab provides you with a detailed description of how to do this.

Graph
: is used to define or modify the individual curves on the diagram.
The help text for the tab provides you with a detailed description of how to do this.

Buttons

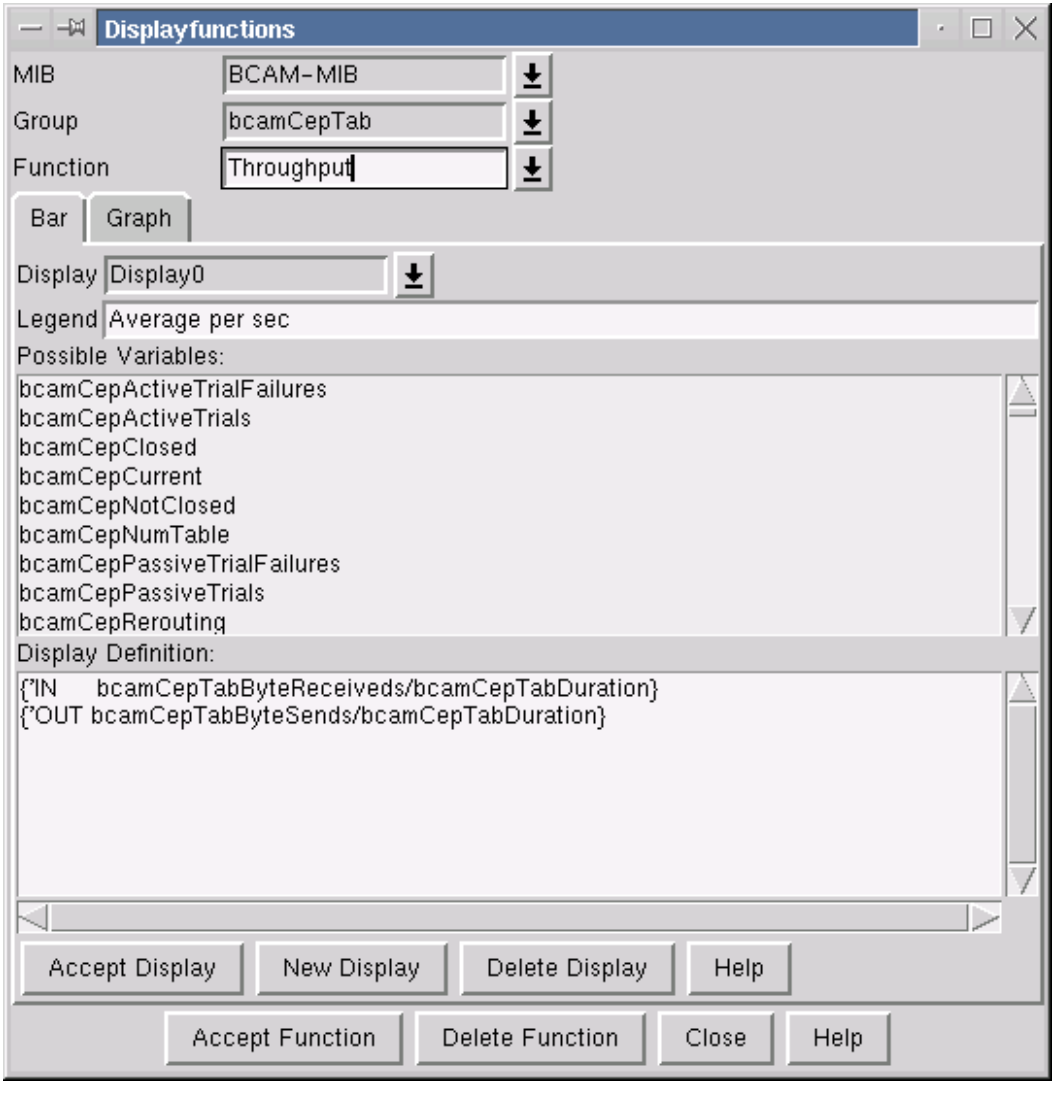
Accept Function
stores all the curves and bar definitions of the current display function.

Delete Function
deletes the current display function.

Close
closes the window.

Help
displays the appropriate help text.

Defining bar display functions for the current function



Main BCAM Manager window → Option menu → Define display functions for

This tab allows you to define bar display functions for the current functions. A *bar display function* comprises one or more displays, each of which represent a bar or a group of bars.

A *bar group* is a set of bars which are defined as a single display and which normally belong together from a logical point of view.

A *bar* represents an x-value and a function expression involving one or more MIB variables graphically.

If more than one MIB table instance exists, a bar is displayed for each instance.

A function expression can include any of the following operations:

+	addition
-	subtraction
*	multiplication
/	division
(or)	expressions in parentheses

The following functions, which include the name of exactly one MIB variable as an argument, are also available:

DIFF(x) .	Difference between the current value and the previous value of the variable.
GRADIENT(x) .	Difference between the current value and the previous value of the variable divided by the time interval (in seconds) between the two queries.

The following functions return exactly one value, even for more than one instance:

SUM(x).	Sum of the MIB variables over all instances.
GRADIENTSUM(x) .	Sum of GRADIENT(x) over all instances.

Working in the window

You must do the following if you want to create or modify a display function:

1. Generate a new display by clicking on the *New Display* button or select a display from the selection field *Display*. The name of the display is shown in the selection field *Display*.
2. Enter the legend in the input field *Legend*.
3. Enter the bar definition(s) in the text field *Display Definition*.
4. Confirm your definition by clicking on the *Accept Display* button, otherwise the bar definition is lost when you switch to a different display.

Window elements

The selection field *Display*
contains the name of the current bar definition. This name is automatically assigned when a new bar is created (Display0 ... Display<n>). The names of existing bar definitions are listed in the corresponding selection list.

The input field *Legend*
allows you to define the legend to be displayed for the bar.

The list *Possible Variables:*
contains a list of all the MIB variables available for use in the bar definition. If you double-click on a variable, this variable is inserted into the text field *Display Definition* at the current cursor position.

The text field *Display Definition:*
contains the bar definition. The definition of a bar consists of an expression or several expressions separated by blanks with the following format: {xdef ydef}

xdef stands for the name of a MIB variable or a fixed string without blanks and which starts with the character " ' "

ydef stands for a function expression which can include the following:

- numeric constants (e.g.: 5 2.87 1.34e3)
- names of MIB variables from the displayed list of variables
- the operators + - * /
- the parentheses (and)
- the functions DIFF(x), GRADIENT(x), SUM(x), GRADIENTSUM(x)

Example: (assume that the MIB variables are a, b, c, d, e, f)

```
{'Text  b}
{a      b}
{a      b + c * SUM(d)}
{'Text  a * (b + SUM(c)) / (GRADIENT(d) - GRADIENTSUM(e)) + DIFF(f)}
```

Buttons

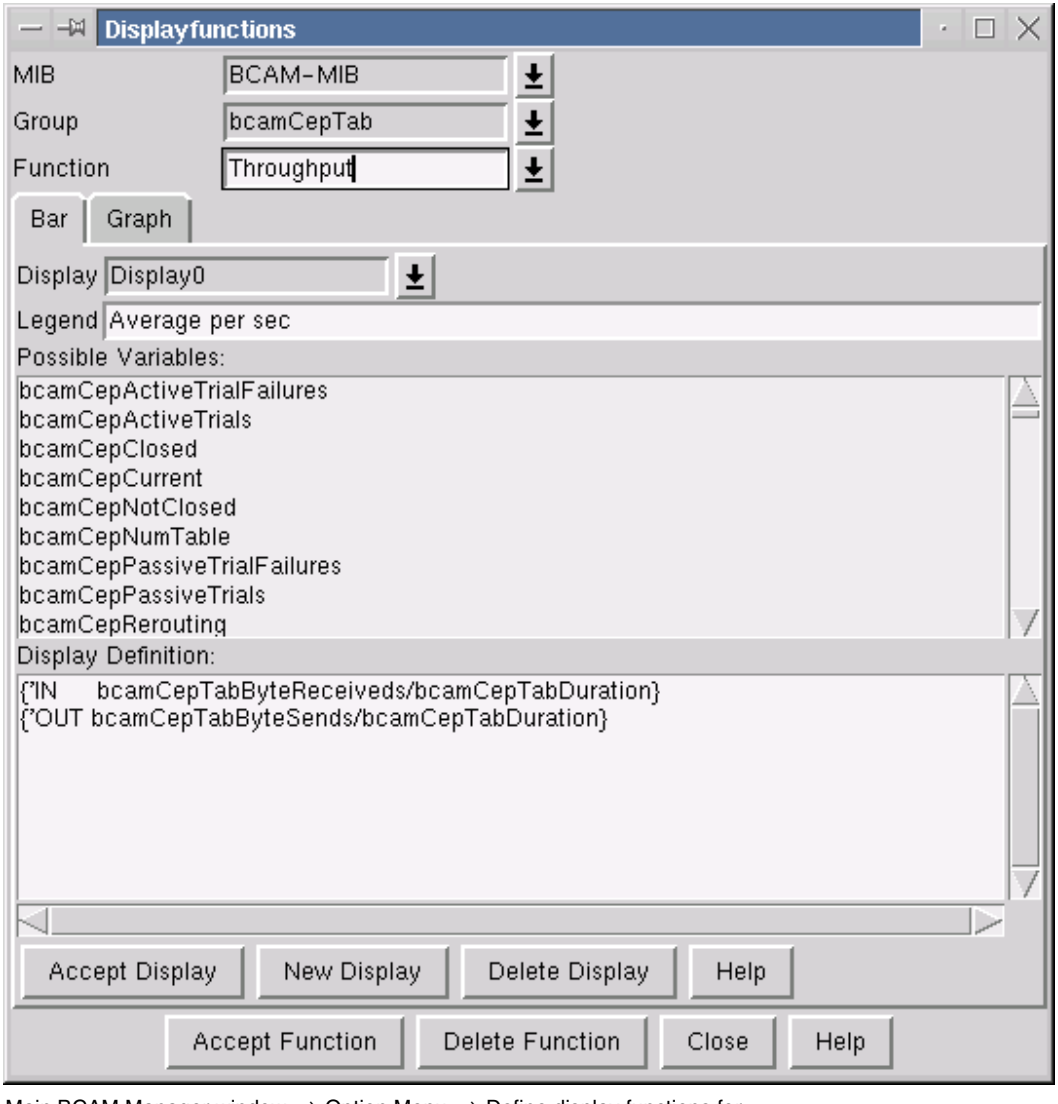
Accept Display
stores the definition of the current display.

New Display
generates a new display.

Delete Display
deletes the current display.

Help
displays a help text on how to define a bar.

Defining curve display functions for the current function



Main BCAM Manager window → Option Menu → Define display functions for

This tab allows you to define curve display functions for the current function. A *curve display function* comprises one or more displays, each of which represent a curve.

A *curve* represents a function expression involving one or more MIB variables graphically. If more than one MIB table instance exists, a curve is displayed for each instance.

A function expression can include any of the following operations:

+	addition
-	subtraction
*	multiplication
/	division
(or)	expressions in parentheses

The following functions, which include the name of exactly one MIB variable as an argument, are also available:

DIFF(x) .	Difference between the current value and the previous value of the variable.
GRADIENT(x) .	Difference between the current value and the previous value of the variable divided by the time interval (in seconds) between the two queries.

The following functions return exactly one value, even for more than one instance:

SUM(x).	Sum of the MIB variables over all instances.
GRADIENTSUM(x) .	Sum of GRADIENT(x) over all instances.

Working in the window

You must do the following if you want to create or modify a display function:

1. Generate a new display by clicking on the *New Display* button or select a display from the selection field *Display*. The name of the display is shown in the selection field *Display*.
2. Enter the legend in the input field *Legend*.
3. Enter the curve definition in the text field *Display Definition*.
4. Confirm your definition by clicking on the *Accept Display* button, otherwise the curve definition is lost when you switch to a different display.

Window elements

The selection field *Display*
contains the name of the current curve definition. This name is automatically assigned when a new curve is created (Display0 ... Display<n>). The names of existing curve definitions are listed in the corresponding selection list.

The input field *Legend*
allows you to define the legend to be displayed for the curve.

The list *Possible Variables:*
contains a list of all the MIB variables available for use in the curve definition. If you double-click on a variable, this variable is inserted into the text field *Display Definition* at the current cursor position.

The text field *Display Definition:*
contains the curve definition. The definition of a curve consists of an mathematical expression which can include the following:

- numeric constants (e.g.: 5 2.87 1.34e3)
- names of MIB variables from the displayed list of variables
- the operators + - * /
- the parentheses (and)
- the functions DIFF(x), GRADIENT(x), SUM(x), GRADIENTSUM(x)

Examples: (assume the MIB variables are a, b, c, d, e, f)

a

a + b * SUM(c)

a * (b + SUM(c)) / (GRADIENT(d) - GRADIENTSUM(e)) + DIFF(f)

Buttons

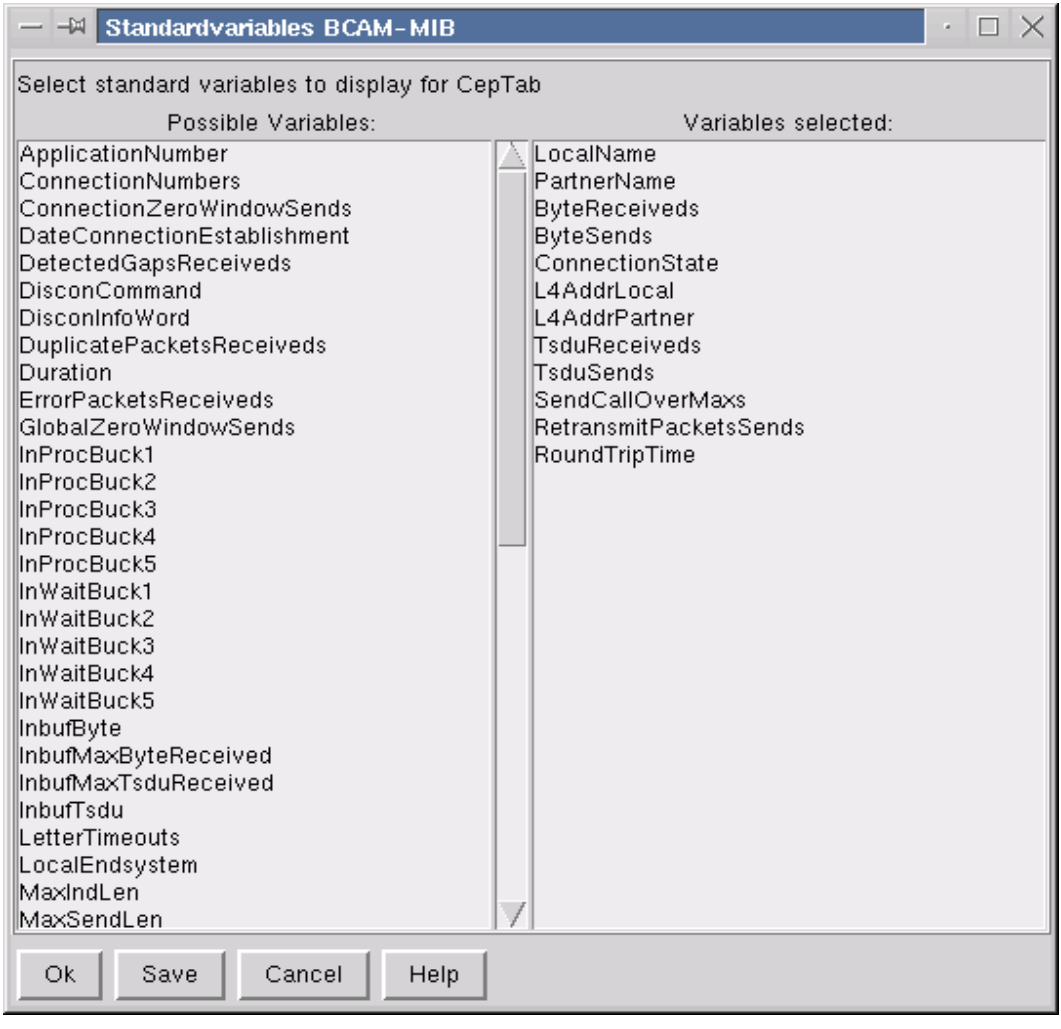
Accept Display
stores the definition of the current display.

New Display
generates a new display.

Delete Display
deletes the current display.

Help
displays a help text on how to define a curve.

4.3.3 Defining standard variables for a MIB group or MIB table



Main BCAM Manager window → Option menu → Define standard variables for

This window allows you to define the standard variables for a MIB group or table (i.e. the variables to be displayed in the *Systemview* window).

Working in the window

The window contains two lists. The list on the left is used to display the variables from the MIB group or MIB table which are available for selection, and the list on the right is used to display the variables selected as standard variables. If you click on a variable name with the left mouse button it is moved from one list to the other.

Window elements

Buttons

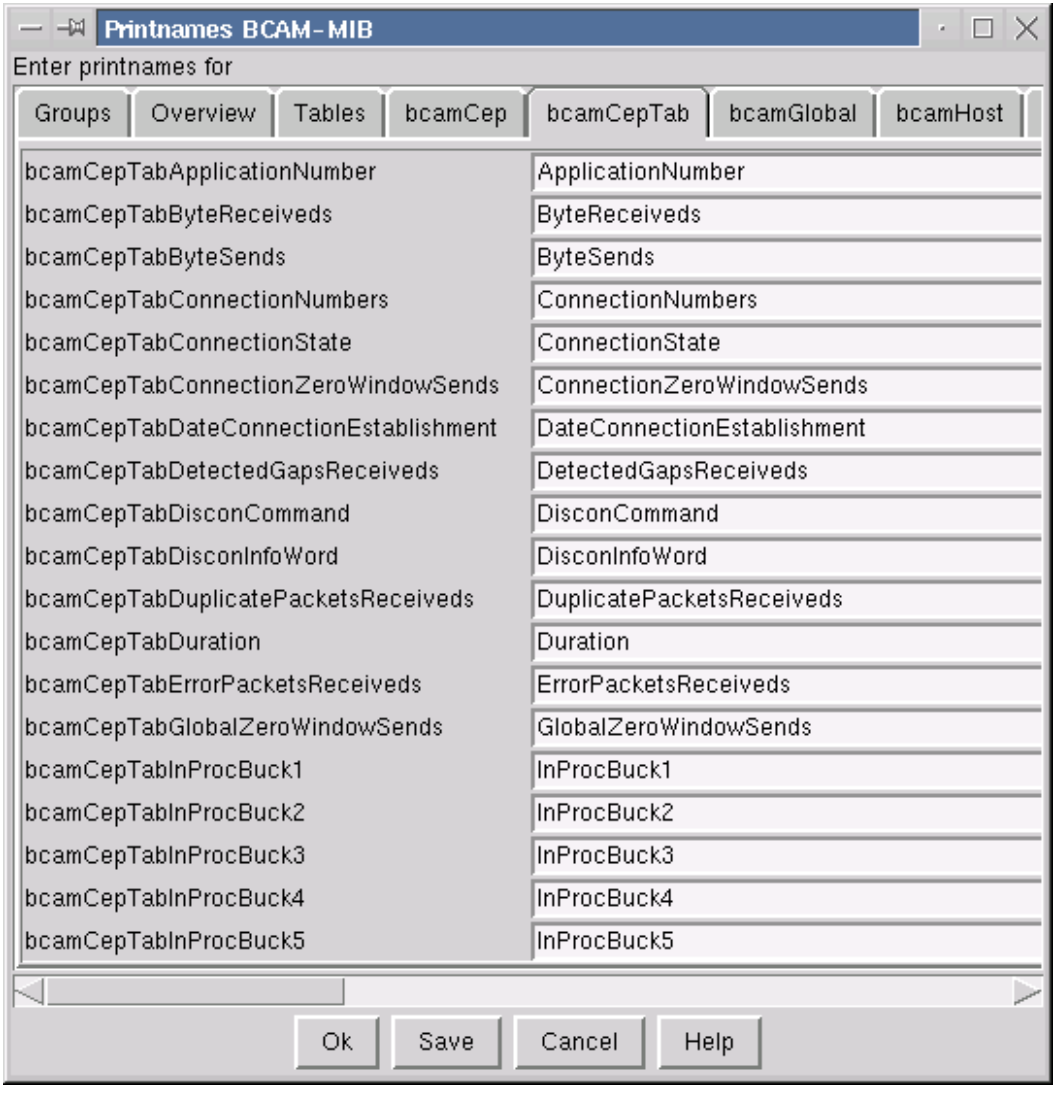
Ok
: accepts the variables currently displayed in the list on the right as temporary standard variables, which are valid until the program is terminated, and closes the window.

Save
: saves the variables currently displayed in the list on the right as permanent standard variables.

Cancel
: aborts the selection process and uses the previously valid variables as standard variables.

Help
: displays the appropriate help text.

4.3.4 Defining print names for MIB variables



Main BCAM Manager window → Option menu → Define print names for

This window allows you to define print names for MIB variables, which are used to represent the MIB variable names.

Window elements and working in the window

The window contains a separate tab for each of the groups and tables defined in the MIB. You use these tab to define print names for the variables on the tab. The tabs *Groups*, *Overview a*nd *Tables* also exist for each MIB. *Groups* and *Tables* allow you to define group and table names, while *Overview* contains all the variables from all the groups (but no table variables). These print names are used only in the *Overview* tab in the *Systemview* window since variables from different groups can be displayed here and the print names could be ambiguous. When you click on a tap with the left mouse button, this tab moves to the front. Each tab includes a row for each MIB variable which contains the complete MIB variable name and an input field with the print name.

Buttons

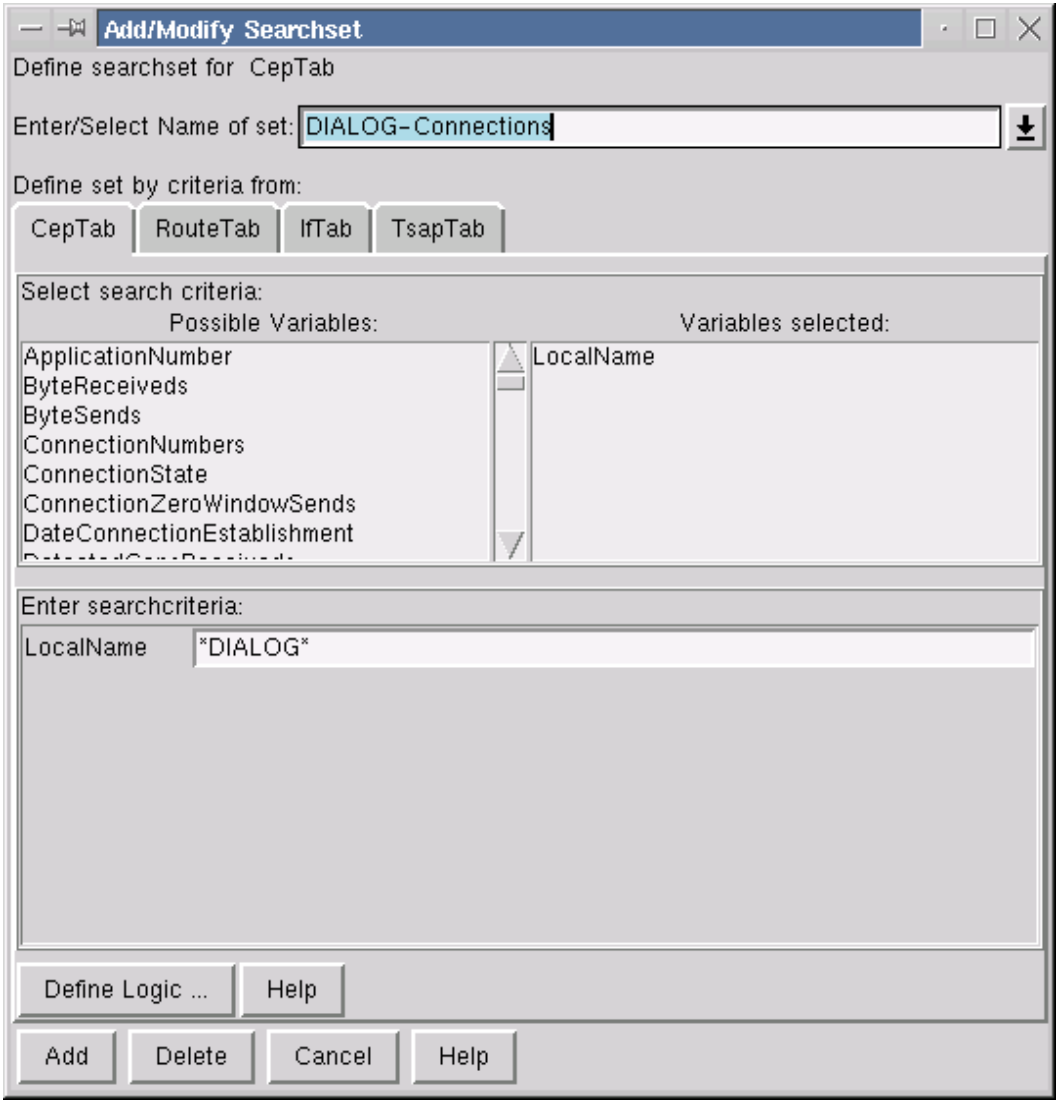
Ok
: accepts the print names as temporary print names, which are valid until the program is terminated, and closes the window.

Save
: saves the print names as permanent print names and closes the window.

Cancel
: cancels the selection operation and sets the previous print names. The window is closed.

Help
: displays the appropriate help text.

4.3.5 Adding and modifying search sets



Main BCAM Manager window → Option menu → Add/Modify Searchset

This window allows you to add or modify a search set. Search sets contain descriptions for searching for MIB table instances.

Each search set has a name and comprises a set of search criteria from one or more MIB tables, the search logic and the corresponding search values. Search criteria are the MIB tables values. Search values are possible values for these variables (named values, i.e. values with symbolic names, are possible for integer variables). If you specify several possible values, a search criteria is satisfied if the actual value corresponds to at least one search value. The search logic defines how the individual search criteria are linked (by default all the search criteria must be satisfied).
Search sets are defined for a specific MIB table.

Working in the window

The window contains an input field in which you enter the name of the new search set followed by the <Return> key. If you want to modify a search set, you can select the name of the search set you want to modify from the list of known search sets.

There is a tab for each MIB table from which you can select search criteria. Each tab contains a list of the possible search criteria on the left and the list of selected search criteria on the right. If you select a MIB variable from the list of possible variables by clicking on it with the left mouse button, it is entered in the list of selected variables and a new input field with this name is generated. You can use this input field to specify search values, separated by blanks, for MIB variables with non-named values.

You can use the following special characters for MIB variables of the type *DisplayString*:

* .	any number of arbitrary characters
?	exactly one arbitrary character
[list of characters] .	exactly one character from the specified list of characters
\character.	stands for the character *character* and cancels the special characters * ? [ ] \

The following ranges are possible for integer MIB variables:

> x.	greater than x
≥ x .	greater than or equal to x
< x .	lesser than x
≤ x.	lesser than or equal to x
= x.	equal to x
≠ x.	not equal to x
x - y.	lies in the range between x and y (including x and y)

In the case of MIB variables with named values, you may select exactly one value from the list of possible values.

The button *Define Logic...*
opens another window in which you can modify the search logic.
This window contains an editable text field which contains the names of the selected search criteria and their logical operators.

Possible logical operators are:

&&	AND
||	OR
!	NOT

The logical operations can be enclosed in parentheses.

The *Ok* button
accepts the specified search logic and closes the window.

The *Cancel* button
cancels the modifications and closes the window.

Buttons

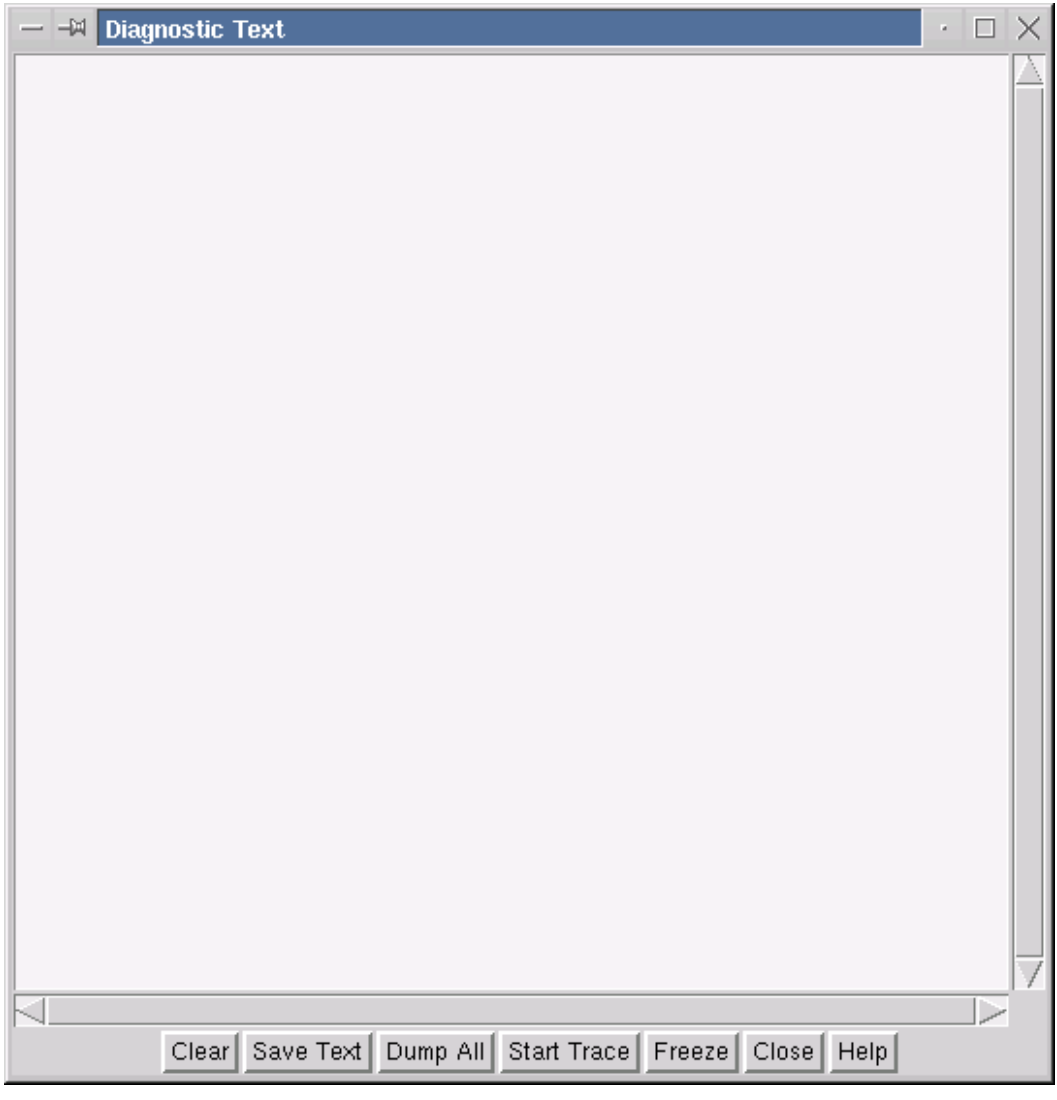
Add
saves the specified search set.

Delete
deletes the specified search set.

Cancel
closes the window

Help
displays the appropriate help text.

4.4 Displaying important events



The *Diagnostic Text* window is used to display important events.
It also supports the collection of the various types of diagnostic information in a file.

The background color changes with each event to make it easier to distinguish between the various events.

Window elements and working in the window

Buttons

Clear
deletes the text in the window.

Save Text
saves the text in a file.

Dump All
saves the following information in a file:

1. diagnostic text
2. the names and characteristics of all the windows
3. the names of all global variables and their values

Start Trace
starts saving the diagnostic text and other diagnostic information in a file. The label of the button changes to *Stop Trace*.

Stop Trace
closes the file opened with *Start Trace*.
The label of the button changes to *Start Trace*.

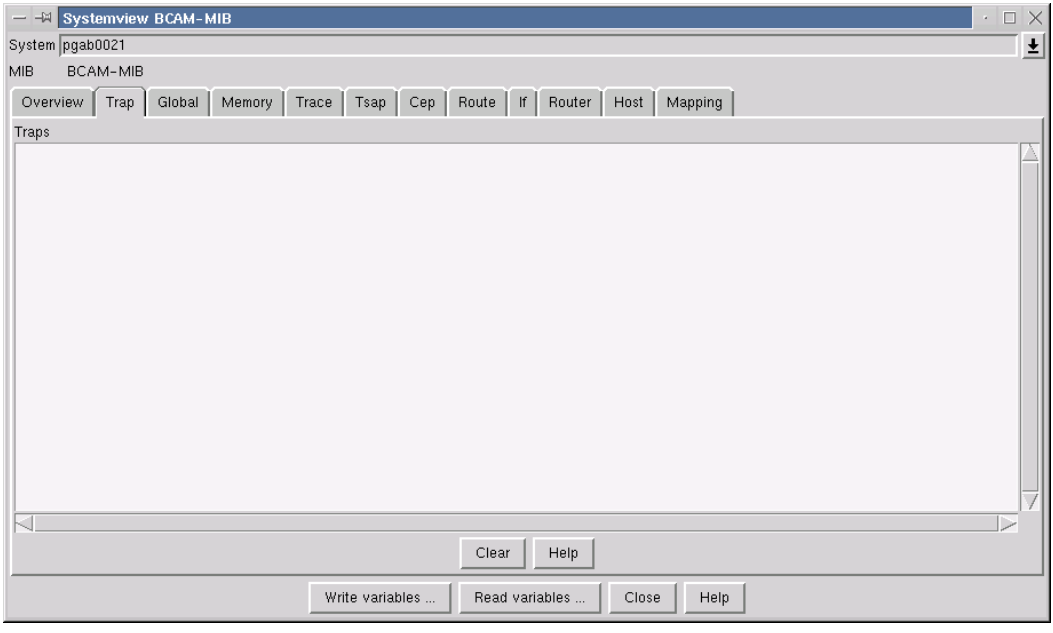
Freeze
freezes the text in the current window. Future output is displayed in a new window. The label of the button changes to *Search*.

Search
extends the window to include a search frame which makes it easier to find a search string in the text window. If you click on the *Forward* button first, the search begins with the first visible row (i.e. visible when you click on the *Search* button). If you click on the *Backward* button first, the search begins with the last visible row (i.e. visible when you click on the *Search* button). When the beginning/end of the text is reached, the search is continued at the end/beginning of the text.
If the string is found, the corresponding text element is positioned on the first visible row. Future search steps will start there.
If the string is not found, a reference to the new text window is displayed.

Close
closes the diagnostic text window.

Help
displays the appropriate help text.

4.5 Displaying all the traps for all loaded MIBs and known systems



This window is used to display all the traps received for all the MIBs that have been loaded and all known systems. This window is generated as soon as the first trap is received.

Window elements

The text field *Traps*
is used to display the traps received using the following format:

"Trap received from system SYMBNAME (XXX.XXX.XXX.XXX) YY.YY.YYYY ZZ:ZZ:ZZ"

SYMBNAME .	The symbolic name of the system or empty.
XXX.XXX.XXX.XXX.	The IP address from which the trap was received
YY.YY.YYYY .	The date with the format: day, month, year.
ZZ:ZZ:ZZ .	The time with the format hours, minutes, seconds.

This is followed by the MIB variables received with the trap with the format:

"VVVVV : WWWWWW" .

VVVVV: The symbolic MIB variable name (if known)

WWWWW: The value of the MIB variable

Buttons

Clear
deletes all the traps entered in the text field.

Help
displays the appropriate help text.

5 SNMP subagent for FTP

There is a separate subagent (FTP subagent) for the FTP server. The FTP subagent, like the MIB-II and BCAM subagents, are operated via the BCAM Manager.

The ASN1 source file for the FTP-MIB is included in the library SYSSPR.TCP-IP-AP.040 as a type-S member under the name FTP.MIB.

You will find information on software requirements, installation and deinstallation, and starting and stopping the FTP subagent in the chapter „SNMP for openNet Server and interNet Services“ (see page 11).

5.1 Interaction between the FTP subagent and the FTP server

The FTP server can reach the FTP subagent under the fixed port number 3237. Immediately after startup, the FTP server registers with the FTP subagent, provided that it has been started, and supplies it with the following information:

– the port number under which the FTP subagent can reach the FTP server
– the server port number for the control connection to the FTP clients

If no server entry with this server port number already exists, the FTP agent creates an appropriate server entry.

When an FTP server is started, it writes its two port numbers into the file SYSDAT.TCP-IP-AP.040.SNMP. If the FTP subagent is started later, it can obtain information about the currently active FTP servers in the file SYSDAT.TCP-IP-AP.040.SNMP and create the corresponding data structures.

When the FTP server is shut down, it deletes its entry from the file SYSDAT.TCP-IP-AP.040.SNMP.

5.2 Read and write accesses to the FTP server

All administration-related information about the FTP server is stored in the FTP-MIB:

– You can obtain information about the FTP server by reading MIB variables.

– You can control the FTP server via SNMP by setting MIB variables.

The BCAM Manager provides a number of windows in which you can read information from and/or write information to the FTP-MIB (see section „Windows for read and write accesses to the FTP-MIB“ on page 124).

5.2.1 Obtaining information about the FTP server

The FTP subagent provides the following information about the FTP server:

– notification about the start or shutdown of an FTP server
– server-specific data
– connection data

Server traps

The FTP subagent sends traps when

– an FTP server is started
– an FTP server is shut down

The FTP subagent always sends the traps together with the server port number.

Server-specific data

The FTP subagent provides the following data about the FTP server:

- port number of the FTP server for the control connections
- status of the server: running, waiting, restarting, shutdown, start, undefined
- maximum number of connections
- number of current connections
- timeout for connections
- FTAC level 0 / 1 / 2
- FTAC job class
- FTP version of the server
- host name of the system on which the FTP server is executed
- debugging level
- socket trace level

Connection data

This comprises:

- connection index
- status of the connection:
  - connected
  - logged-in
  - login complete
  - logoff in progress
  - logoff completed
- user ID used by the client for its login
- name of the client host

5.2.2 Controlling the FTP server via SNMP

The FTP subagent supports numerous write accesses involving the FTP-MIB and thus provides a number of options for controlling the FTP server:

– start the FTP server
– set the FTAC level (when the FTP server is started)
– shut down the FTP server
– activate/deactivate a socket trace
– activate/deactivate debugging
– save the log file
– increase the maximum possible number of parallel connections
– modify the timeout value for connections
– set the FTAC job class (FTACJob)

Starting the FTP server

The FTP subagent starts an FTP server as follows:

– If the desired port number is 21, the FTP subagent uses the existing file SYSENT.TCP-IP-AP.040.FTPD as the start procedure.

– If the desired port number <portnumber> is not 21 (standard port for the FTP server), the FTP subagent creates a new enter file with the name SYSENT.TCP-IP-AP.040.FTPD.<portnumber> from the enter file SYSENT.TCP-IP-AP.040.FTPD and uses this file as the start procedure.

– The FTP subagent saves the log in the file SYSOUT.TCP-IP-AP.040.FTPD.<portnumber>.

– The FTP subagent uses the name F<portnumber> as the job name (server option -A) (see the "*inter*Net Services Administrator's Guide").

– The FTP subagent uses the number <portnumber> as the port number (server option -P) (see the "*inter*Net Services Administrator's Guide").

To start the FTP server, write the desired port number in the FTP-MIB and set the server status (MIB variable *state*) in the FTP-MIB to "start" (see also page 133). You can set the FTAC level when you start the FTP server, but this only makes sense if the FTAC job class in the MIB table *FtpServerParamTable* is subsequently set to an appropriate value (see also page 135).

Shutting down the FTP server

To shut down the FTP server, set the server status (MIB variable *state*) in the FTP-MIB to "shutdown" (see also page 133). This has the same effect as the BS2000/OSD command /INTR (see the "*inter*Net Services Administrator's Guide").

Saving the log file

When the log file is saved, the FTP subagent copies the contents of the current log file <prot> to a file with the name <prot>.SAVE. You

To save the log file, set the MIB variable *ProtocolFileSave* in the MIB table *ftpServerTable* to “yes“.

Increasing the maximum possible number of parallel connections

To do this, increase the value of the variable *MaxCon* in the MIB table *ftpServerParamTable*. *MaxCon* indicates the maximum connection index, i.e. the maximum possible number of parallel connections. Values between 0 and 900 are permitted.

The connection index numbers the parallel connections of an FTP server in ascending order thus allowing each connection of the server involved to be clearly identified.

The value of the *MaxCon* variable remains unchanged

– if you specify the value “0“.

– if you specify the highest current connection index.

– If you specify a value smaller than the highest current connection index. In this case, *MaxCon* is automatically assigned the value of the highest current connection index (see also page 137).

Modifying the timeout value for connections

The timeout value for connections indicates the maximum time period during which a connection can be inactive without being terminated.

To modify the timeout value, modify the value of the *ConTimeout* variable in the MIB table *ftpServerParamTable*. Values between 0 and 2,147,483,647 are permitted; however, specifying the value “0” has no effect (see also page 138).

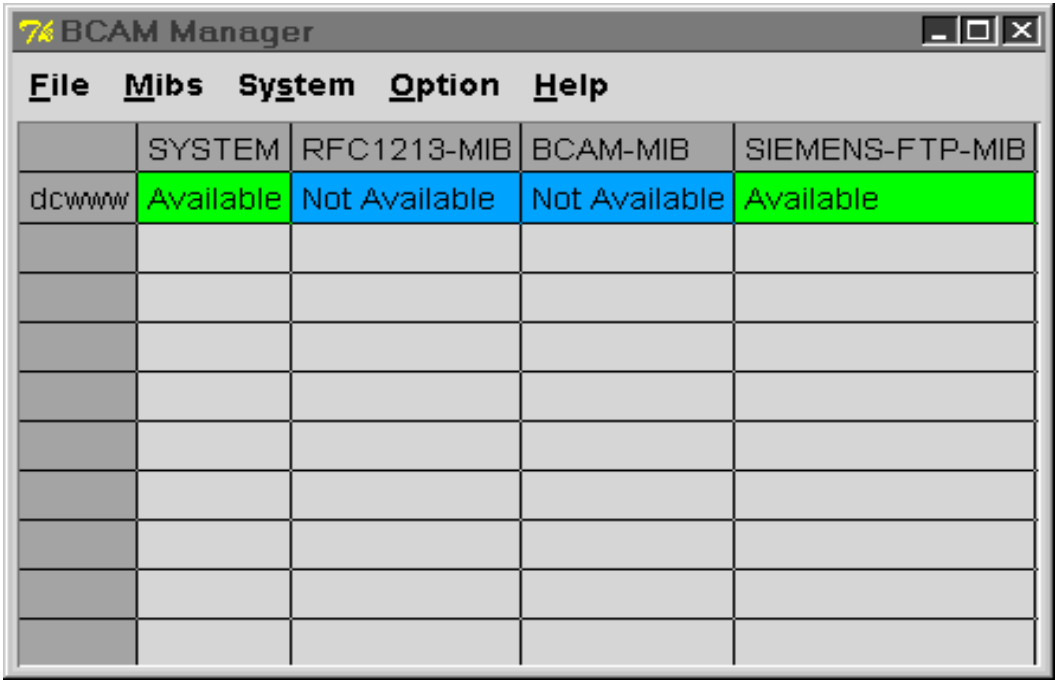
5.3 Windows for read and write accesses to the FTP-MIB

The BCAM Manager provides you with a number of windows which you can use to read information from and/or write information to the FTP-MIB.

The windows of the BCAM Manager are described in detail in the chapter „Windows of the BCAM Manager“ (see page 21). You will find general information on the windows and menus provided by the BCAM Manager in this chapter.

5.3.1 Start window for read and write accesses

The starting point is the main window of the BCAM Manager (see page 21).



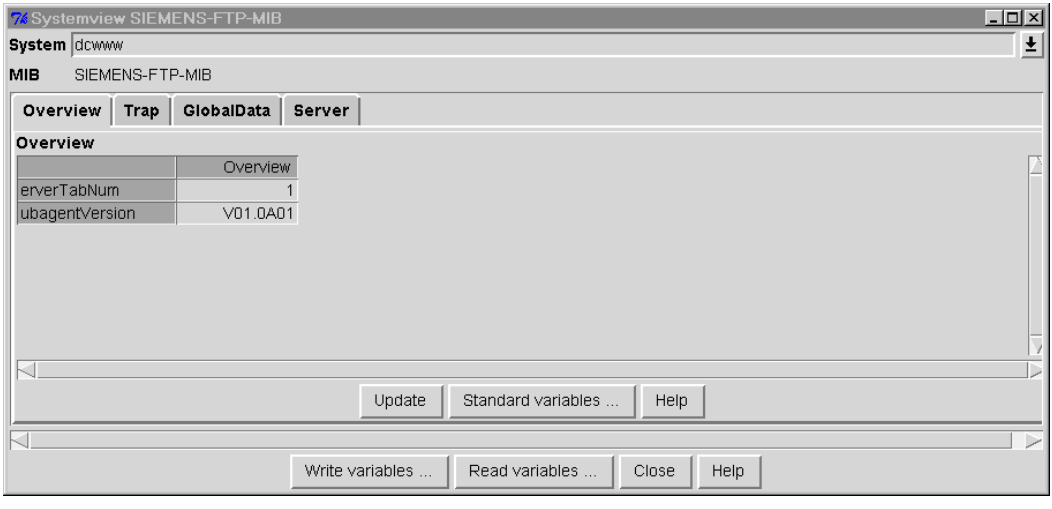
Section of the main window of the BCAM Manager

Start window for querying and modifying the FTP-MIB

You move to the start window of the FTP-MIB from the main window of the BCAM manager by performing the following steps:

1. Select *Mibs* → *Load MIB* from the menu in the main BCAM Manager window and enter the file name of the FTP-MIB. The *Systemview FTP-MIB* window is then displayed.
2. Select *System* → *Add/Modify System* from the menu in the main BCAM Manager window and enter the name of the system you want.
3. Select *System* → *Systemview for MIB* from the menu in the main BCAM Manager window and click on *FTP-MIB*.

The start window for querying and modifying the FTP-MIB is then displayed.



Start window for querying and modifying the FTP-MIB

If you want to query the values of the individual MIB groups, you will need to use the individual tabs. These tabs are shown and explained in the next section „Window for read accesses - querying MIB variables“ (see page 126).

If you want to set the values for individual MIB variables, click on the *Write variables* button. The windows used to set the MIB variables are shown and explained in the section „Windows for write accesses - setting MIB variables“ (see page 132).

5.3.2 Window for read accesses - querying MIB variables

The start window for querying and modifying the FTP-MIB contains a tab for each group of the FTP-MIB. If the MIB group contains MIB tables, the tab is subdivided even further.

Tabs in the start window for querying the FTP-MIB

The start window for querying an MIB contains the following tabs:

- *Overview*
- *Trap*
- *GlobalData*
- *Server*

***Overview** tab*

The Overview tab provides information about the MIB variables *ServerTabNum* and *SubagentVersion*.

ServerTabNum
Number of currently active FTP servers

SubagentVersion
Current version of the FTP subagent

***Trap** tab*

The *Trap* tab is used to display all of the traps sent for the FTP-MIB and system being viewed from the time the window was generated. The trap format is the same as the trap format for BCAM-MIB and MIB-II (see page 48).

The FTP subagent sends traps when the FTP server is started and when it is shut down. The variable is the server port number.

***GlobalData** tab*

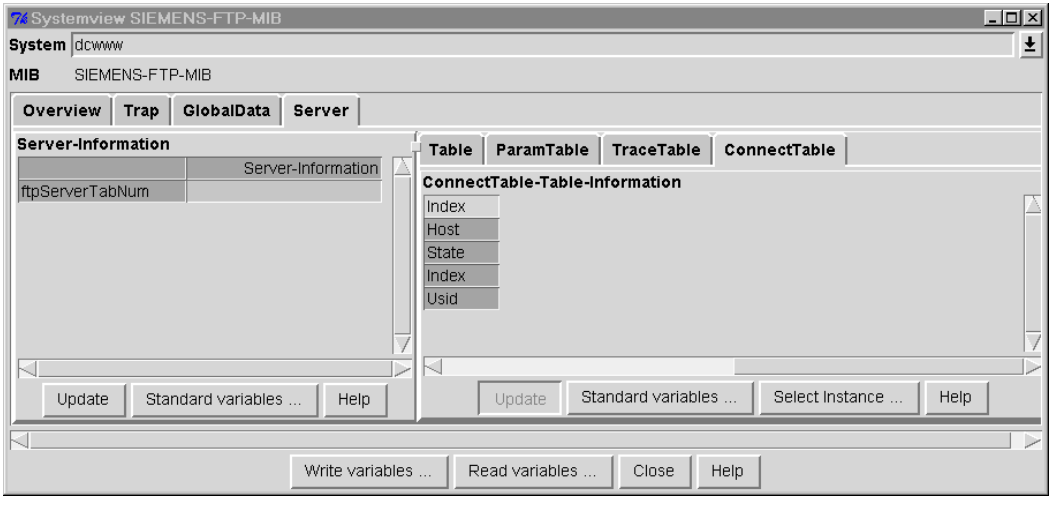
The *GlobalData* tab provides information about the MIB variable *ftpSubagentVersion*.

ftpSubagentVersion
Current version of the FTP subagent

***Server** tab*

The *Server* tab provides information about the MIB variable *ftpServerTabNum*.

ftpServerTabNum
Number of currently active servers



Server tab in the start window for querying and modifying the FTP-MIB

A bar with the available tables is also displayed (see “MIB tables displayed via the *Server* tab” below).

MIB tables displayed via the *Server* tab

The following MIB tables are displayed via the *Server* tab:

– MIB table *Table*
– MIB table *ParamTable*
– MIB table *TraceTable*
– MIB table *ConnectTable*

MIB table ***Table***

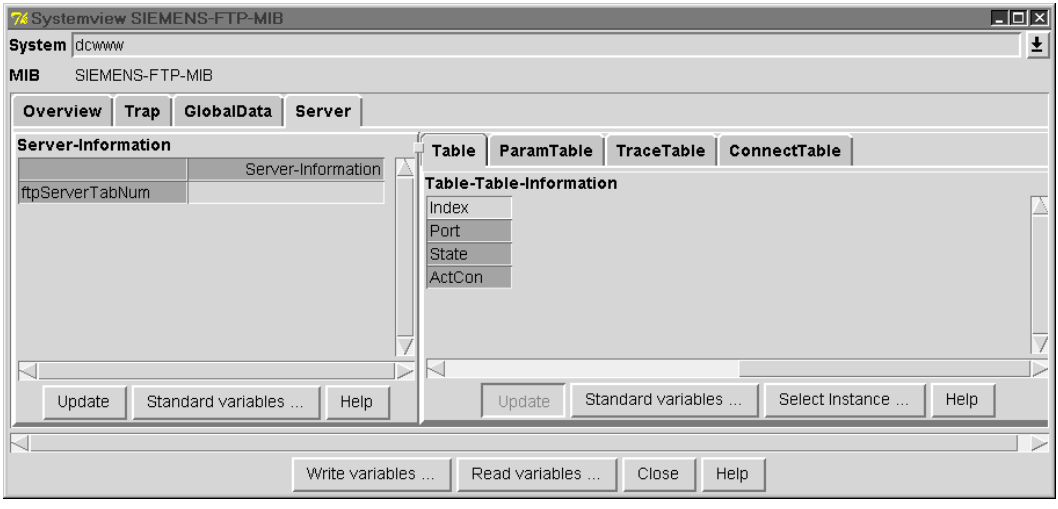


Table *Table* for the server group

Index
: indicates the index. The index is the same as the server port number.

ActCon
: Number of current connections

Port
: Server port number

State
: Status of the server

Possible values are:

- running
- waiting
- restarting
- shutdown
- undefined
- start

MIB table ***ParamTable***

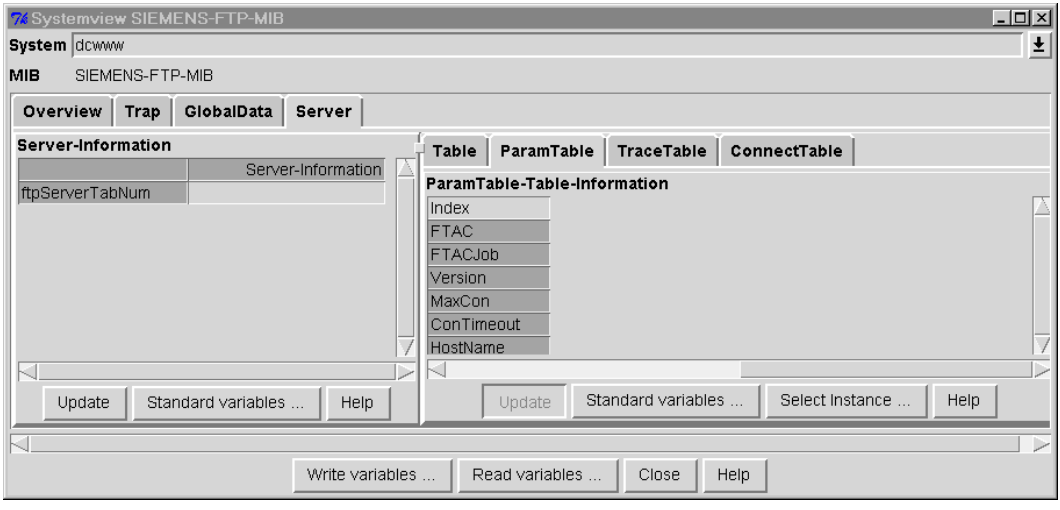


Table *ParamTable* for the server group

Index
indicates the index. The index is the same as the server port number.

ConTimeout
Timeout for the connection (in seconds)

FTAC
FTAC level (0 /1/ 2). The FTAC level indicates the level at which the FTAC check is to be performed:

– Value ”0“:
No FTAC access check

– Value “1“:
Access via a dialog ID is not checked by FTAC; access with a *FTAC-Transfer-Admission* is, however, also possible.

– Value “2“:
An FTAC check is always performed.

FTACJob
FTAC job class. It must be possible to start enter jobs in this class with the parameter SCHEDULINGTIME=*PARAMETERS(START=*IMMEDIATELY).

HostName
Name of the host on which the FTP server is executing

MaxCon
Maximum number of permitted connections

Version
Version of the FTP server

*MIB table **TraceTable***

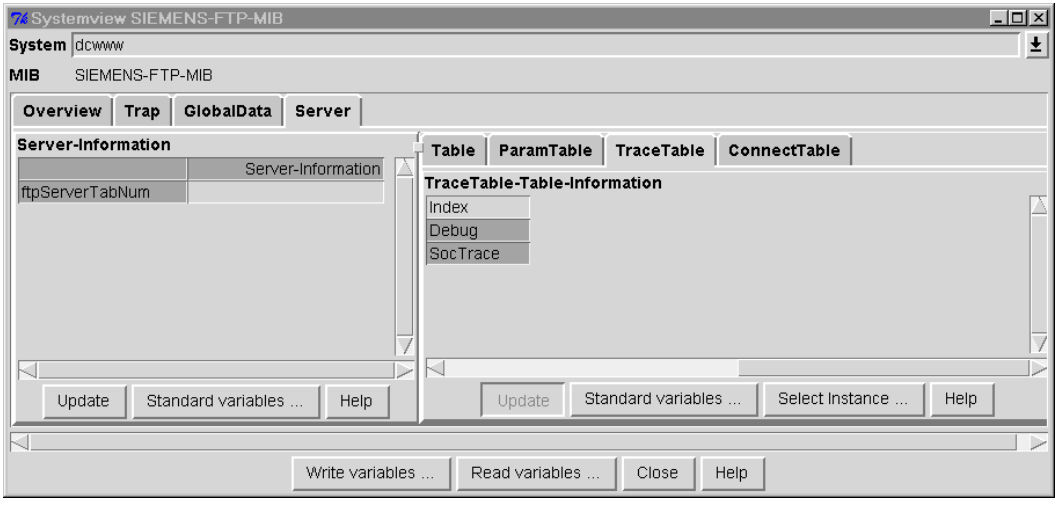


Table *TraceTable* of the server group

Index
indicates the index. The index is the same as the server port number.

Debug
Level of the FTP user trace

SocTrace
Level of the socket trace

MIB table ***ConnectTable***

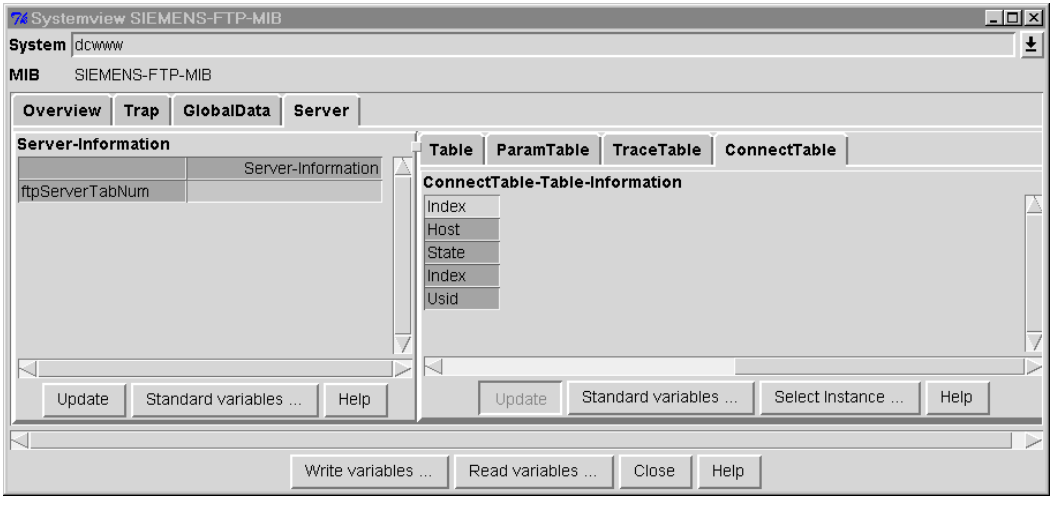


Table *ConnectTable* of the server group

Index
: indicates the index. The index is a pair of numbers consisting of the server port number and the number of the connection.

Host
: Name of the host on which the FTP client is running

Index
: Number of the connection (connection index)

State
: Status of the connection

Possible values are:

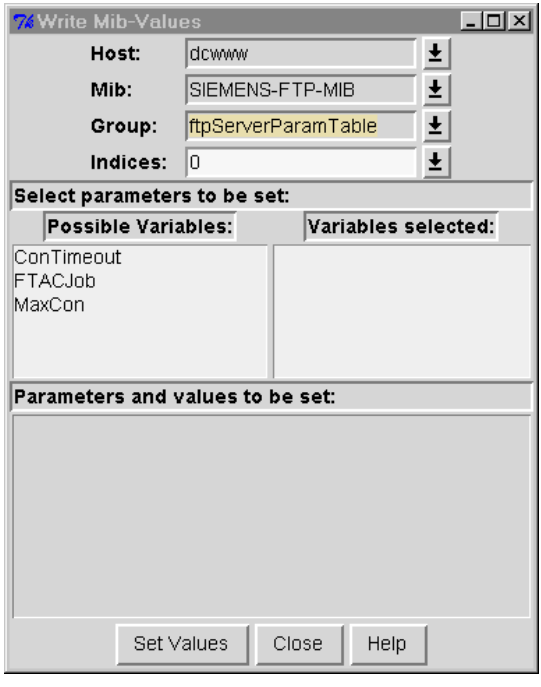
- connected
- login_in_progress
- logged_in
- logoff_in_progress
- logoff_completed
- undefined

Usid
: User ID under which the client logged in

5.3.3 Windows for write accesses - setting MIB variables

The FTP subagent supports various write accesses involving the FTP-MIB and thus provides a numerous of options for controlling the FTP server via SNMP:

- Click on the *Write variables* button in the start window for querying and modifying the FTP-MIB (see page 125). The start window *Write MIB Variables*, which is used to set MIB variables, is then displayed.



BCAM Manager → System → Systemview for a MIB → Write variables

Window elements

Indexes
Specify the desired server port number

Group
Specify the MIB group

You may enter the following values:

- ftpServerTable
- ftpServerTraceTable
- ftpServerParamTable

MIB variables for the *ftpServerTable* group

State

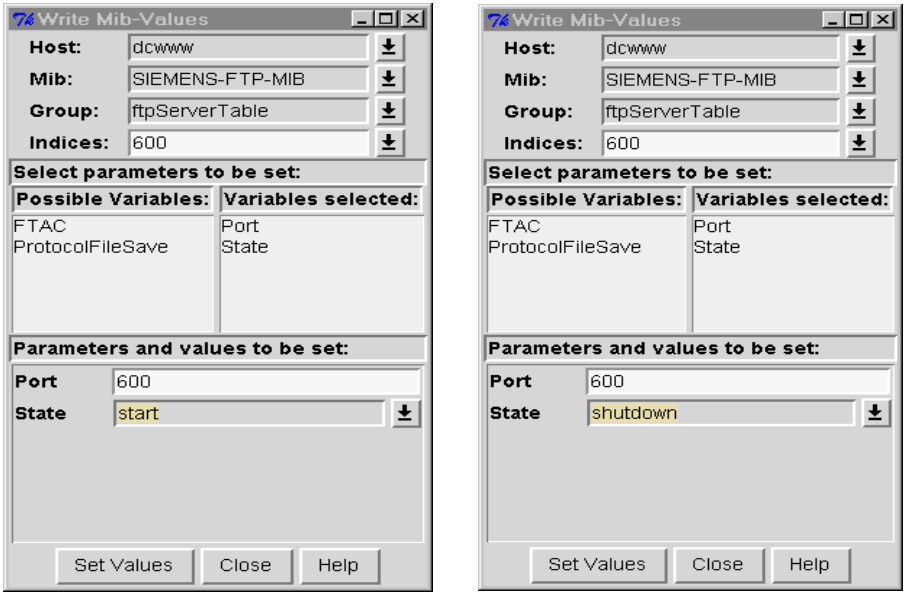
Status of the FTP server

The following values can be read:

– running
– waiting
– restarting
– shutdown
– undefined
– start

You may enter the following values:

– start
– shutdown



Window for starting or shutting down an FTP server

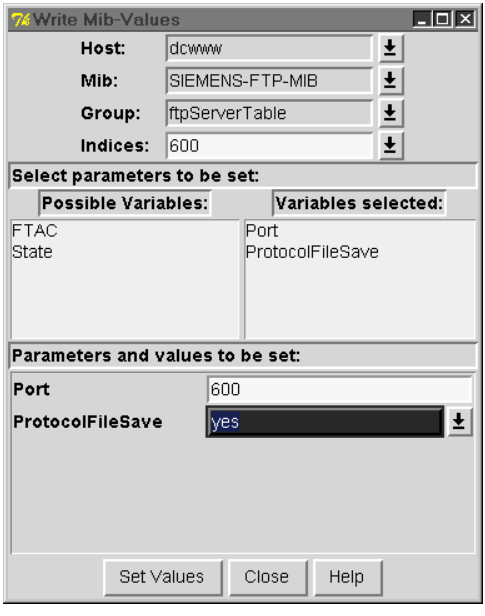
Port

Specify a port number.
If you specify a port number and, at the same time, specify the value “start” in the *State* variable, an FTP server is started with the specified port number. Please note that no FTP server can already have been started with this port number.

You may enter the following values: 1 .. 9999

ProtocolFileSave

Specify whether the log file of the FTP server is to be saved.
You may enter the following value: ”yes“



Indicating that the log file of the FTP server is to be saved

FTACLevel

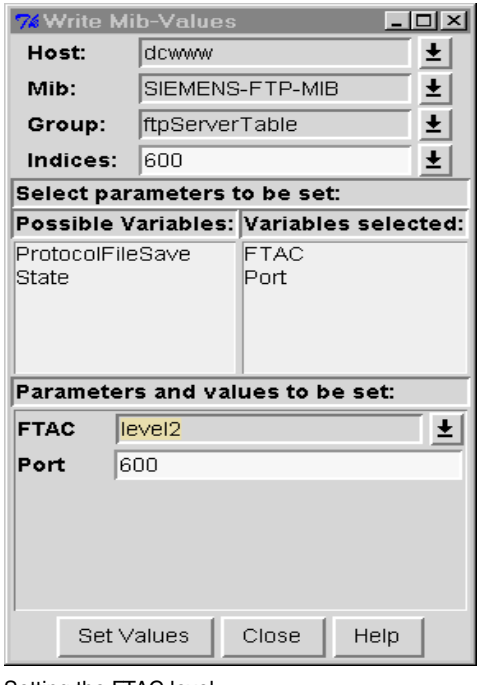
specify the FTAC level with which the FTP server is to be started.
You may enter the following values: 0 / 1 / 2

The FTAC level indicates the level at which the FTAC check is to be performed:

– Value ”0“:
No FTAC access check

– Value “1“:
Access via a dialog ID is not checked by FTAC; access with a *FTAC-Transfer-Admission* is, however, also possible.

– Value “2“:
An FTAC check is always performed.



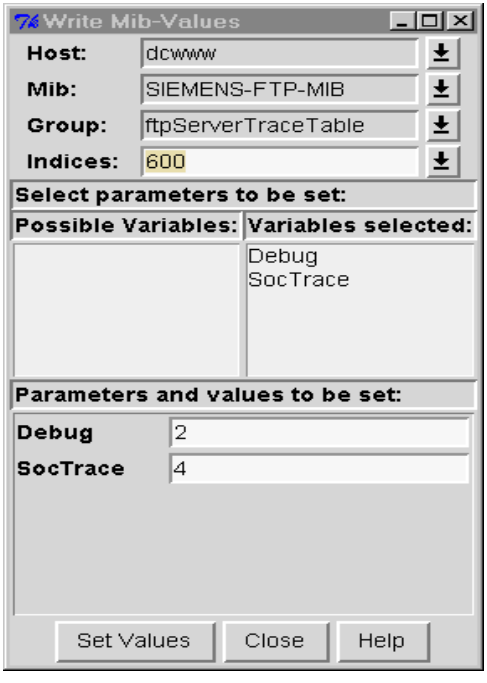
Setting the FTAC level

MIB variables for the *ftpServerTraceTable* group

Debug / SocTrace

Specify the level of the user trace (debugging) and/or socket trace

Possible values: 0 .. 9
You may specify the following values: 0 .. 9

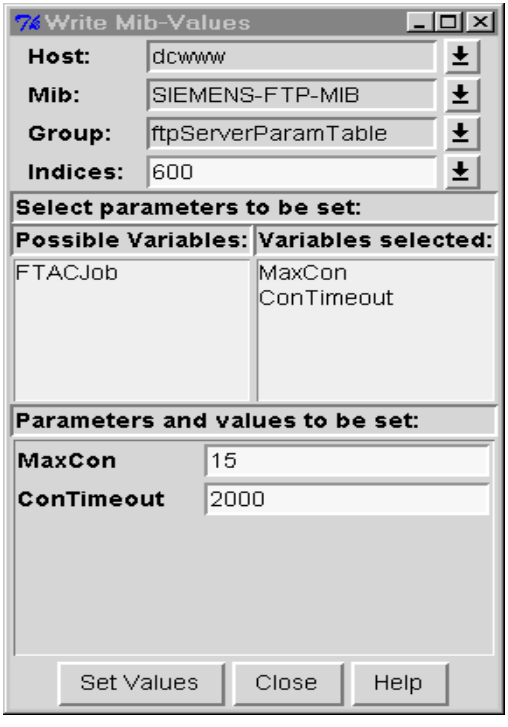


Specifying the values for the user and socket traces

MIB variables for the *ftpServerParamTable* group

MaxCon

Specify the maximum possible number of parallel connections.
You may specify the following values: 0 .. 900



Specifying the maximum possible number of parallel connections

In the following cases, the desired modification cannot be made even though a valid value has been specified:

– The specified *MaxCon* value is smaller than the connection index (see „MIB table ConnectTable“ on page 131). In this case, *MaxCon* is automatically assigned the value of the highest existing connection index.

– The value of the FTP server option *-S* of the server involved plus the number of digits in the desired value for *MaxCon* is greater than 8.

  To find out whether this is the case, you can save the log file of the FTP server by setting the MIB variable *ProtocolFileSave* to “yes“ (see page 134). In the event of an error, the log file contains the following entry:

```
ftpd:maximum 8 figures for Terminalidentification and appl
```

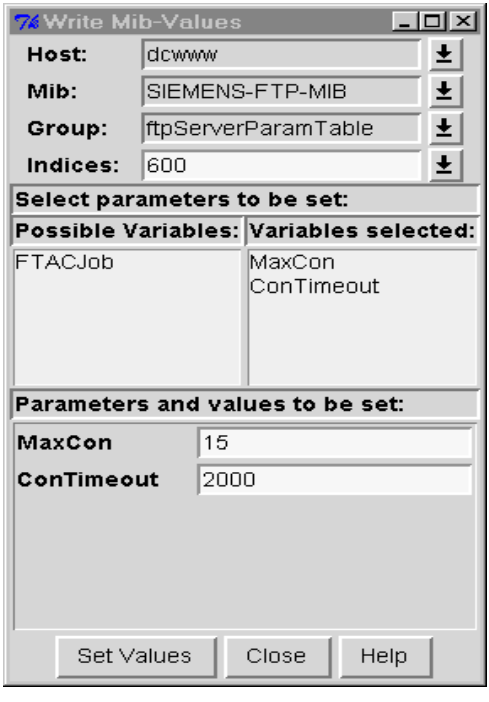
– The value of the FTP server option *-A* (length of the application prefix of the server involved) plus the number of digits in the desired value for *MaxCon* is greater than 8.

  To find out whether this is the case, you can save the log file of the FTP server by setting the MIB variable *ProtocolFileSave* to “yes“ (see page 134). In the event of an error, the log file contains the following entry:

```
ftpd:parameter appl too long
```

ConTimeout

Specify the timeout for connections.
You may specify the following values: 0..2,147,483,647



Specifying the timeout for connections

FTACJob

Specify the FTAC job class.
This specification is only relevant if the FTP server was started with an FTAC level > 0.
The *FTACJob* specification must be made before the first connection on the FTP server is started.

6 MIB-II (RFC 1213), BCAM.MIB (private), FTP.MIB

This chapter will provide you with a detailed description of

– MIB-II,
– BCAM.MIB and
– FTP.MIB.

6.1 MIB-II

The information supplied in the MIB-II is divided into five groups. MIB-II

Name	Definition
Interface group	In the interface group, you will find information on the network connections, their addresses and status. The information in the Interface group also includes the amount of data transmitted and any irregularities that occurred during transmission on the network connection.
IP group	The IP group contains information on the number of IP datagrams sent and received and any errors that occurred. In addition, tables are available that provide you with information on the assignment of IP addresses to network connections and hardware addresses, as well as routing information.
ICMP group	The ICMP group informs you of the reachability of IP hosts and provides you with information on errors that occurred when IP datagrams were forwarded.
TCP group	The TCP group provides information, for example, on the number of TCP connections, the total amount of data transmitted, and the number of attempts made to open and close connections. In addition, a table provides you with information on the partner and status of each current TCP connection.
UDP Group	The UDP group provides you with information on the total number of UDP datagrams received and sent, as well as the total number of errored UDP datagrams and those that could not be delivered. It also provides you with a table of the port numbers used locally.

MIB-II groups

The individual objects for which access is permitted by the BCAM subagent are shown in the following tables.

6.1.1 Interface group

Object name	Access	Explanation
ifNumber	read-only	Number of network connections (irrespective of whether inactive or active).
The following values exist for every individual network connection:		
ifIndex	read-only	Unique number of network connections. Is used by SNMP for unique addressing of the network connections.
ifDescr	read-only	Verbal description of the network connection
ifType	read-only	Hardware description of the network connection - values "!other", "ethernet-csmacd", "fddi" supplied
ifMTU	read-only	Maximum datagram size which can be sent or received via the network connection. Figure in bytes.
ifSpeed	read-only	Transmission speed (bit/sec) for this network connection
ifPhysAddress	read-only	Layer-2 address of the network connection
ifAdminStatus	read-write	Administrative status of the network connection (up, down, testing)
ifOperStatus	read-only	Operative status of the network connection (up, down, testing)
ifLastChange	read-only	Time of the last amendment to the operative status
ifInOctets	read-only	Number of bytes received
ifInUcastPkts	read-only	Number of unicast packets received, which were forwarded to layer 3.
ifInNUcastPkts	read-only	Number of non-unicast packets, which were forwarded to layer 3.

Interface group

Object name	Access	Explanation
ifInDiscards	read-only	Number of recommended packets which were OK, but were discarded for internal reasons.
ifInErrors	read-only	Number of packets received which were discarded due to layer-2 protocol errors.
ifInUnknownProtos	read-only	Number of packets received which contained an unknown layer-3 protocol.
ifOutOctets	read-only	Number of bytes sent.
ifOutUcastPkts	read-only	Number of unicast packets contained by layer 3 for sending.
ifOutNUcastPkts	read-only	Number of non-unicast packets contained by layer 3 for sending
ifOutDiscards	read-only	Number of packets to be sent that were OK but were discarded for internal reasons
ifOutErrors	read-only	Number of packets to be sent which were discarded due to errors
ifOutQLen	read-only	Number of packets to be sent which are still in the corresponding queue.
ifSpecific	read-only	Reference to specific MIB entries of this network connection; currently preset to { 0 0 }.

Interface group

6.1.2 IP group

Object name	Access	Explanation
ipForwarding	read-write	Display of the capability to route received IP datagrams further (IP routing functionality)
ipDefaultTTL	read-write	Default IP lifetime specification (Hop counter) BCAM default = 32 sec
ipInReceives	read-only	Number of IP datagrams received
ipInHdrErrors	read-only	Number of IP datagrams received with incorrect IP header
ipInAddrErrors	read-only	Number of received IP datagrams with unknown or incorrect destination IP address
ipForwDatagrams	read-only	Number of IP datagrams received which were routed further.
ipInUnknownProtos	read-only	Number of IP datagrams received which were discarded due to a higher protocol which was not supported.
ipInDiscards	read-only	Number of IP datagrams received which were discarded due to internal problems
ipInDelivers	read-only	Number of IP datagrams received and forwarded to a higher protocol
ipOutRequests	read-only	Number of send requests
ipOutDiscards	read-only	Number of IP datagrams to be sent which were discarded due to internal problems.
ipOutNoRoutes	read-only	Number of IP datagrams to be sent that could not be routed further to destination address.
ipReasmTimeout	read-only	Maximum time in seconds that fragmented datagrams may require to be fully reassembled in the IP layer.
ipReasmReqds	read-only	Number of IP fragments which had to be reassembled to complete datagrams.
ipReasmOKs	read-only	Number of IP datagrams which were reassembled successfully.

IP group

Object name	Access	Explanation
ipReasmFails	read-only	Number of IP datagrams, which were not successfully reassembled
ipFragOKs	read-only	Number of IP datagrams which were successfully fragmented
ipFragFails	read-only	Number of IP datagrams which could not be successfully fragmented
ipFragCreates	read-only	Number of IP fragments which were eventually sent.
ipRoutingDiscards	read-only	Number of IP datagrams to be routed which were discarded due to internal problems.
The following five values exist for each individual IP address:		
- ipAdEntAddr	read-only	IP address which the address information belongs to
- ipAdEntIfIndex	read-only	Reference to the network connection (see ifIndex)
- ipAdEntNetMask	read-only	Subnet mask belonging to IP address
- ipAdEntBcastAddr	read-only	Indicator whether the IP broadcast address consists of "0" or "1".
- ipAdEntReasmMaxSize	read-only	Maximum size of a IP datagram which can still be reassembled.
The following five values exist for each individual IP route:		
ipRouteDest	read-write	Destination address of the route
ipRouteIfIndex	read-write	Reference to the network connection (see ifIndex)
ipRouteMetric1	read-only	Characteristics according to ipRouteProto (preset to -1 in BCAM)
ipRouteMetric2	read-only	Characteristics according to ipRouteProto (preset to -1 in BCAM)
ipRouteMetric3	read-only	Characteristics according to ipRouteProto (preset to -1 in BCAM)
ipRouteMetric4	read-only	Characteristics according to ipRouteProto (preset to -1 in BCAM)
ipRouteNextHop	read-write	IP address of the next hop on this route

IP group

Object name	Access	Explanation
ipRouteType	read-write	Type of route, possible values: direct or indirect
ipRouteProto	read-only	Routing protocol used for the route; the values "other", "local", "icmp" are supplied.
ipRouteAge	read-only	Time in seconds since the last change to the route
ipRouteMask	read-only	Subnet mask of the route
ipRouteMetric5	read-only	Characteristics according to ipRouteProto (preset to -1 in BCAM)
ipRouteInfo	read-only	Reference to specific MIB entries of this route; currently preset to { 0 0 }.
The following values exist for the each assignment for an IP address to a physical address		
ipNetToMediaIfIndex	read-write	Reference to the network connection (see ifIndex)
ipNetToMediaPhysAddress	read-write	Layer-2 address which is assigned to the route.
ipNetToMediaNetAddress	read-write	corresponding IP address
ipNetToMediaType	read-write	Type of assignment, possible values: other invalid dynamic static

IP group

6.1.3 ICMP group

Object name	Access	Explanation
icmpInMsgs	read-only	Total number of received ICMP messages
icmpInErrors	read-only	Total number of received, incorrect ICMP messages
icmpInDestUnreachs	read-only	Total number of received destination unreachable messages
icmpInTimeExcds	read-only	Total number of received time exceeded messages
icmpInParmProbs	read-only	Total number of received parameter problem messages
icmpInSrcQuenchs	read-only	Total number of received source quench messages
icmpInRedirects	read-only	Total number of received redirect messages
icmpInEchos	read-only	Total number of received echo request messages
icmpInEchoReps	read-only	Total number of received echo reply messages
icmpInTimestamps	read-only	Total number of received time stamp request messages
icmpInTimestampReps	read-only	Total number of received time stamp reply messages
icmpInAddrMasks	read-only	Total number of received address mask request messages
icmpInAddrMaskReps	read-only	Total number of received address mask reply messages
icmpOutMsgs	read-only	Number of ICMP send requests
icmpOutErrors	read-only	Number of incorrect ICMP send requests
icmpOutDestUnreachs	read-only	Total number of sent destination unreachable messages
icmpOutTimeExcds	read-only	Total number of sent time exceeded messages
icmpOutParmProbs	read-only	Total number of sent parameter problem messages

ICMP group

Object name	Access	Explanation
icmpOutSrcQuenchs	read-only	Total number of sent source quench messages
icmpOutRedirects	read-only	Total number of sent redirect messages
icmpOutEchos	read-only	Total number of sent echo request messages
icmpOutEchoReps	read-only	Total number of sent echo reply messages
icmpOutTimestamps	read-only	Total number of sent time stamp request messages
icmpOutTimestampReps	read-only	Total number of sent time stamp reply messages
icmpOutAddrMasks	read-only	Total number of sent address mask request messages
icmpOutAddrMaskReps	read-only	Total number of sent address mask reply messages

ICMP group

6.1.4 TCP group

Object name	Access	Explanation
tcpRtoAlgorithm	read-only	Algorithm to determine the retransmit behavior of messages when sent. Supplied value: vanj
tcpRtoMin	read-only	Minimum time after which a retransmit is initialized.
tcpRtoMax	read-only	Maximum time after which a retransmit is initialized.
tcpMaxConn	read-only	Maximum number of connections which can exist at the same time. Supplied value: -1, because dynamic
tcpActiveOpens	read-only	Number of actively opened TCP connections
tcpPassiveOpens	read-only	Number of passively opened TCP connections
tcpAttemptFails	read-only	Total number of incorrectly opened TCP connections
tcpEstabResets	read-only	Number of TCP connections closed with RESET
tcpCurrEstab	read-only	Number of currently open TCP connections
tcpInSegs	read-only	Number of received TCP segments
	read-only	Number of TCP segments sent
tcpOutSegs	read-only	Number of TCP segments for which a retransmit was necessary.
tcpInErrs	read-only	Number of received, incorrect TCP segments
tcpOutRsts	read-only	Number of TCP resets sent

TCP group

Object name	Access	Explanation
The following values exist for every TCP connection:		
tcpConnState	read-only	Status of connection, possible values: closed(1) listen(2) synSent(3) synReceived(4) established(5) finWait1(6) finWait2(7) closeWait(8) astAck(9) closing(10) timeWait(11) deleteTCB(12)
tcpConnLocalAddress	read-only	Local IP address of the connection
tcpConnLocalPort	read-only	Local port number of the connection
tcpConnRemAddress	read-only	Remote IP address of the connection
tcpConnRemPort	read-onlyt	Remote port number of the connection

TCP group

Writing access

Object name	Access	Explanation
tcpConnState	read-only	Change status of the connection Only the following value can be set: deleteTCB - equivalent to a disconnect

TCP group

6.1.5 UDP group

Object name	Access	Explanation
udpInDatagrams	read-only	Number of received UDP datagrams
udpNoPorts	read-only	Number of received UDP datagrams for which no application was opened.
udpInErrors	read-only	Number of received, incorrect UDP datagrams
udpOutDatagrams	read-only	Number of sent UDP datagrams
The following values exist for every UDP application		
udpLocalAddress	read-only	Local IP address of the UDP application; 0.0.0.0 for applications that listen on all network ports.
udpLocalPort	read-only	Local port number of the UDP application

UDP group

6.2 BCAM.MIB (private)

The information in the BCAM.MIB can be divided into 12 groups.

Name	Definition
Trap group	The trap group is the first group. It does not contain any values that can be queried, but only the definitions required to send traps.
Global group	The global group contains BCAM-specific values that cannot be assigned to any of the following groups.
Memory group	The memory group contains values that indicate how much memory is being used by BCAM.
Trace group	The trace group provides information on the status of BCAM traces.
TSAP group	The TSAP group contains values that are specially related to applications. It consists of two sections: a general section and the table section.
Connection group	The connection group contains values associated with connections. It consists of two sections: a general section and the table section.
Route group	The route group contains values related to routes. It consists of two sections: a general section and the table section.
Interface group	The interface group contains values pertaining to network connections. It consists of two sections: a general section and the table section.
Router group	The router group contains values concerning routers. It consists of two sections: a general section and the table section.
Host group	The host group contains values related to hosts. It consists of two sections: a general section and the table section.
Mapping group	The mapping group contain values related to mapping. It consists of two section: a general section and the table section. As used here, mapping refers to the assignment of a layer-4 address (transport layer address) to an application name.

Groups of the BCAM.MIB

Note

Apart from the trap group, the global group, the memory group and the trace group, the general section contains values that cannot be exclusively assigned to one object. The Table section, by contrast, contains object-specific values in detailed form.

6.2.1 Trap group

Object name	Access	Explanation
bcamTrapString	not-accessible	Textual description of the trap. The trap string contains the values sent with the trap in printable form separated by the keywords given at the description of the individual traps.
bcamTrapOutPoolOverCurrent	not-accessible	The number of bytes of resident memory used for output cells when the trap condition (output pool over maximum) has been recognized.
bcamTrapOutPoolUnderCurrent	not-accessible	The number of bytes of resident memory used for output cells when the trap condition (output pool under maximum) has been recognized.
bcamTrapInPoolOverCurrent	not-accessible	The number of bytes of resident memory used for input cells when the trap condition (input pool over maximum) has been recognized.
bcamTrapInPoolUnderCurrent	not-accessible	The number of bytes of resident memory used for input cells when the trap condition (input pool under maximum) has been recognized.

Trap group (not accessible)

Object name	Access	Explanation
bcamTrapOutPoolOver	read-only	This trap will be sent if the resident memory used for output cells exceeds the limit (bcamMemoryPoolOutputLimitTrap) for more than 5 seconds. bcamTrapString has the fields: $HOST$: <hostname> $UPTIME$: <bcamuptime> $OUTLIMIT$: <memlimit> $OUTCURRENT$: <memcurrent> The following values will be sent together with the trap: bcamGlobalHostName bcamGlobalUpTime bcamMemoryPoolOutputLimitTrap bcamMemoryPoolOverCurrent bcamTrapString
bcamTrapOutPoolUnder	read-only	This trap will be sent if the resident memory used for output cells has exceeded the limit (bcamMemoryPoolOutputLimitTrap) and falls under the limit for more than 5 seconds. bcamTrapString has the fields: $HOST$: <hostname> $UPTIME$: <bcamuptime> $OUTLIMIT$: <memlimit> $OUTCURRENT$: <memcurrent> The following values will be sent together with the trap: bcamGlobalHostName bcamGlobalUpTime bcamMemoryPoolOutputLimitTrap bcamMemoryPoolUnderCurrent bcamTrapString

Trap group

Object name	Access	Explanation
bcamTrapInPoolOver	read-only	This trap will be sent if the resident memory used for input cells exceeds the limit (bcamMemoryPoolInputLimitTrap) for more than 5 seconds. bcamTrapString has the fields: $HOST$: <hostname> $UPTIME$: <bcamuptime> $INLIMIT$: <memlimit> $INCURRENT$: <memcurrent> The following values will be sent together with the trap: bcamGlobalHostName bcamGlobalUpTime bcamMemoryPoolInputLimitTrap bcamTrapInPoolOverCurrent bcamTrapString
bcamTrapInPoolUnder	read-only	This trap will be sent if the resident memory used for input cells has exceeded the limit (bcamMemoryPool InputLimitTrap) and falls under the limit for more than 5 seconds. bcamTrapString has the fields: $HOST$: <hostname> $UPTIME$: <bcamuptime> $INLIMIT$: <memlimit> $INCURRENT$: <memcurrent> The following values will be sent together with the trap: bcamGlobalHostName bcamGlobalUpTime bcamMemoryPoolInputLimitTrap bcamTrapInPoolUnderCurrent bcamTrapString

Trap group

Object name	Access	Explanation
bcamTrapLinkUp	read-only	This trap will be sent if an network connection enters the state active (bcamIfAdminState). bcamTrapString has the fields: $HOST$: <hostname> $UPTIME$: <bcamuptime> $NUMBER$: <linknumber> $NAME$: <linkname> The following values will be sent together with the trap: bcamGlobalHostName bcamGlobalUpTime bcamIfTabNumbers bcamIfTabName bcamTrapString
bcamTrapLinkDown	read-only	This trap will be sent if a network connection leaves the state active (bcamIfAdminState). bcamTrapString has the fields: $HOST$: <hostname> $UPTIME$: <bcamuptime> $NUMBER$: <linknumber> $NAME$: <linkname> The following values will be sent together with the trap: bcamGlobalHostName bcamGlobalUpTime bcamIfTabNumbers bcamIfTabName bcamTrapString
bcamTrapSystemUp	read-only	This trap will be sent after a successful start of BCAM. bcamTrapString has the fields: $HOST$: <hostname> $UPTIME$: <bcamuptime> The following values will be sent together with the trap: bcamGlobalHostName bcamGlobalUpTime bcamTrapString

Trap group

6.2.2 Global group

Object name	Access	Explanation
bcamGlobalBcamVersion	read-only	The version of BCAM in the form xx.yy xx: mainversion yy: subversion
bcamGlobalMibVersion	read-only	Version of BCAM specific MIB (beginning with 1). Will be increased with every enhancement of this MIB
bcamGlobalUpTime	read-only	Time in seconds since start of BCAM
bcamGlobalPortNonpriv	read-only	The lowest port number that may be used by non privileged applications
bcamGlobalPortFree	read-only	The lowest port number that will be assigned dynamical by BCAM
bcamGlobalNumBitmap	read-only	The number of bitmaps to administer slots in the data slot pool. Every bitmap can administer 2048 slots. So the maximum number of slots in a pool is bcamGlobalNumBitmap*2048
bcamGlobalMaxRemoteIp	read-only	The maximum length in bytes BCAM uses for an NSDU for an end system that is reached over at least one router
bcamGlobalEsCreationIp	read-only	The types of IP protocols that will lead to automatic end system creation: ipOff (1): no automatic end system creation due to IP protocols. ipInData (2): automatic end system creation due to IP data received. ipOutData (4): automatic end system creation due to IP data sent. ipRoutingProt (8): automatic end system creation due to IP routing protocols. ipRoutingData (16): automatic end system creation due to routed IP protocols.

Global group

Object name	Access	Explanation
bcamGlobalEsCreationIso	read-only	The types of ISO 8473 and ISO 9542 protocols that will lead to automatic end system creation: isoOff (1): no automatic end system creation due to ISO protocols. isoInData (2): automatic end system creation due to ISO data received. isoOutData (4): automatic end system creation due to ISO data sent. isoRoutingProt (8): automatic end system creation due to ISO routing protocols. isoRoutingData (16): automatic end system creation due to routed ISO protocols.
bcamGlobalBroadcast	read-only	Describes how incoming broadcasts will be treated. on (2): broadcasts will be evaluated and answered if necessary. off (1): broadcasts will be ignored.
bcamGlobalArp	read-only	Describes if the address resolution protocol (ARP) is used. off (1): no ARP will be used. quiet (2): ARP will be used the first time a route is used and periodically if there exists a transport connection to the system described by this route and there was no data transfer in the last supervision interval. on (4): ARP will be used the first time a route is used and periodically after the supervision interval.

Global group

Object name	Access	Explanation
bcamGlobalRarp	read-only	Describes if the reverse address resolution protocol (RARP) will be supported. off (1): RARP not supported. on (2): RARP supported.
bcamGlobalInWaitLimit1	read-write	The lowest bucket limit for the recording of the input wait times in milliseconds. The input wait time is the time interval between the announcement of a message to an application and corresponding receive call. Every change will reset all counters for the input wait time.
bcamGlobalInWaitLimit2	read-write	The second bucket limit for the recording of the input wait times in milliseconds. The input wait time is the time interval between the announcement of a message to an application and corresponding receive call. Every change will reset all counters for the input wait time.
bcamGlobalInWaitLimit3	read-write	The third bucket limit for the recording of the input wait times in milliseconds. The input wait time is the time interval between the announcement of a message to an application and corresponding receive call. Every change will reset all counters for the input wait time.
bcamGlobalInWaitLimit4	read-write	The highest bucket limit for the recording of the input wait times in milliseconds. The input wait time is the time interval between the announcement of a message to an application and corresponding receive call. Every change will reset all counters for the input wait time.
bcamGlobalInWaitChange	read-only	The number of times the limits of the input wait time buckets have been changed.
bcamGlobalInWaitSet	read-write	The state of counting into the input wait time buckets. off (2): the counting is stopped. on (1): the counting is running.

Global group

Object name	Access	Explanation
bcamGlobalReactLimit1	read-write	The lowest bucket limit for the recording of the application reaction times in milliseconds. The application reaction time is the time interval between a receive call and a directly following send call. Every change will reset all counters for the application reaction time.
bcamGlobalReactLimit2	read-write	The second bucket limit for the recording of the application reaction times in milli-seconds. The application reaction time is the time interval between a receive call and a directly following send call. Every change will reset all counters for the application reaction time.
bcamGlobalReactLimit3	read-write	The third bucket limit for the recording of the application reaction times in milli-seconds. The application reaction time is the time interval between a receive call and a directly following send call. Every change will reset all counters for the application reaction time.
bcamGlobalReactLimit4	read-write	The highest bucket limit for the recording of the application reaction times in milli-seconds. The application reaction time is the time interval between a receive call and a directly following send call. Every change will reset all counters for the application reaction time.
bcamGlobalReactChange	read-only	The number of times the limits of the application reaction time buckets have been changed.
bcamGlobalReactSet	read-write	The state of counting into the application reaction time buckets. off (2): the counting is stopped. on (1): the counting is running.

Global group

Object name	Access	Explanation
bcamGlobalInProcLimit1	read-write	The lowest bucket limit for the recording of the inbound processing times in milli-seconds. The inbound processing time is the time interval between the arrival of the first byte of a message at layer 2 and the receipt of the last byte this message by the application. Every change will reset all counters for the inbound processing time.
bcamGlobalInProcLimit2	read-write	The second bucket limit for the recording of the inbound processing times in milli-seconds. The inbound processing time is the time interval between the arrival of the first byte of a message at layer 2 and the receipt of the last byte this message by the application. Every change will reset all counters for the inbound processing time.
bcamGlobalInProcLimit3	read-write	The third bucket limit for the recording of the inbound processing times in milli-seconds. The inbound processing time is the time interval between the arrival of the first byte of a message at layer 2 and the receipt of the last byte this message by the application. Every change will reset all counters for the inbound processing time.
bcamGlobalInProcLimit4	read-write	The highest bucket limit for the recording of the inbound processing times in milli-seconds. The inbound processing time is the time interval between the arrival of the first byte of a message at layer 2 and the receipt of the last byte this message by the application. Every change will reset all counters for the inbound processing time.
bcamGlobalInProcChange	read-only	The number of times the limits of the inbound processing time buckets have been changed.

Global group

Object name	Access	Explanation
bcamGlobalInProcSet	read-write	The state of counting into the inbound processing time buckets. off (2): the counting is stopped. on (1): the counting is running.
bcamGlobalOutProcLimit1	read-write	The lowest bucket limit for the recording of the outbound processing times in milliseconds. The outbound processing time is the time interval between the send call of an application and the confirmation for the processing of the last byte of the message by layer 1. Every change will reset all counters for the outbound processing time.
bcamGlobalOutProcLimit2	read-write	The second bucket limit for the recording of the outbound processing times in milliseconds. The outbound processing time is the time interval between the send call of an application and the confirmation for the processing of the last byte of the message by layer 1. Every change will reset all counters for the outbound processing time.
bcamGlobalOutProcLimit3	read-write	The third bucket limit for the recording of the outbound processing times in milliseconds. The outbound processing time is the time interval between the send call of an application and the confirmation for the processing of the last byte of the message by layer 1. Every change will reset all counters for the outbound processing time.

Global group

Object name	Access	Explanation
bcamGlobalOutProcLimit4	read-write	The highest bucket limit for the recording of the outbound processing times in milliseconds. The outbound processing time is the time interval between the send call of an application and the confirmation for the processing of the last byte of the message by layer 1. Every change will reset all counters for the outbound processing time.
bcamGlobalOutProcChange	read-only	The number of times the limits of the outbound processing time buckets have been changed.
bcamGlobalOutProcSet	read-write	The state of counting into the outbound processing time buckets. off (2): the counting is stopped. on (1): the counting is running.
bcamGlobalSnmpRelease	read-write	The time interval in minutes for which the values for already closed connections and applications are still available.
bcamGlobalTrapPollInterval	read-write	The time interval in seconds after which the SNMP agent will poll for traps.
bcamGlobalFileApplTable	read-only	The name of the file that contains the assignments of NEA or ISO applications to the virtual host on which they shall run.
bcamGlobalFileSocketHostTable	read-only	The name of the file that contains the assignments of IP adresses to socket host names for automatically created end systems.
bcamGlobalFileProcTable	read-only	The name of the file that contains the assignments of host names to net addresses for automatically created end systems.

Global group

Object name	Access	Explanation
bcamGlobalAccessProcTable	read-only	The access of the file named in bcamGlobalFileProcTable: read (1): only end systems contained in the file will be created by automatic end system creation. update (2): the file will be extended if a new end system is created.
bcamGlobalHostName	read-only	NEA name of the local end system

Global group

6.2.3 Memory group

Object name	Access	Explanation
bcamMemoryClass3	read-only	The number of bytes of class 3 memory requested by BCAM.
bcamMemoryClass4	read-only	The number of bytes of class 4 memory requested by BCAM.
bcamMemoryLenLongEventSlot	read-only	The length of one long event slot in bytes.
bcamMemoryOccLongEventSlot	read-only	The number of long event slots currently used.
bcamMemoryLenShortEventSlot	read-only	The length of one short event slot in bytes.
bcamMemoryOccShortEventSlot	read-only	The number of short event slots currently used.
bcamMemoryLenTransParamSlot	read-only	The length of one transport parameter block in bytes.
bcamMemoryOccTransParamSlot	read-only	The number of transport parameter blocks currently used.
bcamMemoryLenSnmpParamSlot	read-only	The length of one SNMP parameter block in bytes.
bcamMemoryOccSnmpParamSlot	read-only	The number of SNMP parameter blocks currently used.
bcamMemoryLenApplCb	read-only	The length of one application control block in bytes.
bcamMemoryOccApplCb	read-only	The number of application control blocks currently used.
bcamMemoryLenEnaCb	read-only	The length of one control block in bytes.
bcamMemoryOccEnaCb	read-only	The number of control blocks currently used.
bcamMemoryLenExtApplCb	read-only	The length of one extended application control block in bytes.
bcamMemoryOccExtApplCb	read-only	The number of extended application control blocks currently used.
bcamMemoryLenShortResConn2	read-only	The length of one short resident connection control block 2 in bytes.
bcamMemoryOccShortResConn2	read-only	The number of short resident connection control blocks 2 currently used.

Memory group

Object name	Access	Explanation
bcamMemoryLenLongResConn2	read-only	The length of one long resident connection control block 2 in bytes.
bcamMemoryOccLongResConn2	read-only	The number of long resident connection control blocks 2 currently used.
bcamMemoryLenShortPagConn1	read-only	The length of one short pageable connection control block 1 in bytes.
bcamMemoryOccShortPagConn1	read-only	The number of short pageable connection control blocks 1 currently used.
bcamMemoryLenSnmpConn	read-only	The length of one SNMP connection control block in bytes.
bcamMemoryOccSnmpConn	read-only	The number of SNMP connection control blocks currently used.
bcamMemoryLenLongPagConn1	read-only	The length of one long pageable connection control block 1 in bytes.
bcamMemoryOccLongPagConn1	read-only	The number of long pageable connection control blocks 1 currently used.
bcamMemoryLenShortPagConn2	read-only	The length of one short pageable connection control block 2 in bytes.
bcamMemoryOccShortPagConn2	read-only	The number of short pageable connection control blocks 2 currently used.
bcamMemoryPoolInputCurrent	read-only	The number of bytes of resident memory currently used for input cells
bcamMemoryPoolInputResume	read-only	Describes whether an interface waits for memory for input . If zero, some interfaces will not have enough input buffer (A cell request was rejected because bcamMemoryPoolInputCurrent exceeded bcamMemoryPoolInputLimitTrap).
bcamMemoryPoolInputMonMax	read-only	The maximum number of bytes of resident memory used for input cells for the last 5 seconds.
bcamMemoryPoolInputMonMin	read-only	The minimum number of bytes of resident memory used for input cells for the last 5 seconds.

Memory group

Object name	Access	Explanation
bcamMemoryPoolInputLimit	read-only	The maximum number of bytes of resident memory to be used for input cells.
bcamMemoryPoolInputLimitTrap	read-only	The trap limit for the number of bytes of resident memory for input cells (if bcamMemoryPoolInputMonMin is greater than bcamMemoryPoolInputLimitTrap, a trap will be sent).
bcamMemoryPoolMaxResident	read-write	The maximum number of bytes of resident memory to be used for input and output cells.
bcamMemoryPoolMaxPageable	read-write	The maximum number of bytes of pageable memory to be used for input and output cells.
bcamMemoryPoolInputMaxPageable	read-only	The maximum number of bytes of pageable memory to be used for input cells.
bcamMemoryPoolOutputCurrent	read-only	The number of bytes of resident memory currently used for output cells.
bcamMemoryPoolOutputResume	read-only	Describes whether an interface waits for memory for output . If zero, some interfaces will not have enough output buffer (A cell request was rejected because bcamMemoryPoolOutputCurrent exceeded bcamMemoryPoolOutputLimitTrap).
bcamMemoryPoolOutputMonMax	read-only	The maximum number of bytes of resident memory used for output cells for the last 5 seconds.
bcamMemoryPoolOutputMonMin	read-only	The minimum number of bytes of resident memory used for output cells for the last 5 seconds.
bcamMemoryPoolOutputLimit	read-only	The maximum number of bytes of resident memory to be used for output cells.

Memory group

Object name	Access	Explanation
bcamMemoryPoolOutputLimitTrap	read-only	The trap limit for the number of bytes of resident memory for output cells (if bcamMemoryPoolOutputMonMin is greater than bcamMemoryPoolOutputLimitTrap, a trap will be sent).
bcamMemoryPoolMaxCells	read-only	The maximum number of cells contained in every pool type.
bcamMemoryPoolOutputMaxPageable	read-only	The maximum number of bytes of pageable memory to be used for output cells.
bcamMemoryPoolRoutingCurrent	read-only	The number of bytes of resident memory currently used for routing cells
bcamMemoryPoolRoutingLimit	read-only	The maximum number of bytes of resident memory to be used for routing cells.
bcamMemoryPoolPageableCurrent	read-only	The number of bytes of pageable memory currently used for pools
bcamMemoryPoolPageableFixed	read-only	The number of bytes of pageable fixed memory currently used for output cells.
bcamMemoryCellReqSucc	read-only	The number of successful cell requests.
bcamMemoryCellReqResOutExceed	read-only	The number of requests for resident output cells that failed because the limit has been reached.
bcamMemoryCellReqResInExceed	read-only	The number of requests for resident input cells that failed because limit has been reached.
bcamMemoryCellReqTempExceed	read-only	The number of cell requests failed because of temporary shortage.
bcamMemoryCellReqBitmapFull	read-only	The number of cell requests failed because bitmap was full.
bcamMemoryCellReqNoMemory	read-only	The number of cell requests failed because memory request failed.
bcamMemoryCellReqPagInExceed	read-only	The number of requests for pageable input cells that failed because limit has been reached.
bcamMemoryCellReqPagOutExceed	read-only	The number of requests for pageable output cells that failed because limit has been reached.

Memory group

6.2.4 Trace group

Object name	Access	Explanation
bcamTraceSavingState	read-only	Describes if any of the running BCAM traces are saved to a file. noSaving (32): no trace will be written initiated (1): the trace saving task has been created active (2): the trace saving task is active shutting (4): the trace saving task is terminating passive (8): the trace file has been closed waiting (16): the trace saving task is waiting for trace data to write Several bits may be set simultaneous.
bcamTraceFilename	read-only	The name of the file currently used for saving BCAM traces.
bcamTraceMaxFilesize	read-only	The maximum size of one file used for saving BCAM traces in bytes.
bcamTraceNumberFiles	read-only	The number of files used for saving BCAM traces.
bcamTraceAdmState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceAdmNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceAdmBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).

Trace group

Object name	Access	Explanation
bcamTraceBasicState	read-only	The current state of the trace. stopped(8) : the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceBasicNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceBasicBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).
bcamTraceConnState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceConnNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceConnBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).
bcamTraceLocalState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.

Trace group

Object name	Access	Explanation
bcamTraceLocalNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceLocalBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).
bcamTraceMappingState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceMappingNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceMappingBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).
bcamTraceMgmtState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceMgmtNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceMgmtBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).

Trace group

Object name	Access	Explanation
bcamTraceNetState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceNetNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceNetBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).
bcamTraceSnmpState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceSnmpNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceSnmpBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).
bcamTraceTransState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.

Trace group

Object name	Access	Explanation
bcamTraceTransNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceTransBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).
bcamTraceInfoState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamTraceInfoNumberBuffer	read-only	The number of buffers used for the trace
bcamTraceInfoBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).

Trace group

6.2.5 TSAP group

Object name	Access	Explanation
bcamTsapCurrOpen	read-only	The number of currently active TSAPs (including BCAM internal applications).
bcamTsapTotOpen	read-only	The number of TSAPs opened since start of BCAM (excluding the active TSAPs).
bcamTsapSendCall	read-only	Number of successful send calls of all TSAPs since start of BCAM
bcamTsapSendByteHigh	read-only	Number of bytes given to BCAM-interface to send from all TSAPs since start of BCAM (high word, multiply by 4294967296).
bcamTsapSendByteLow	read-only	Number of bytes given to BCAM-interface to send from all TSAPs since start of BCAM (low word, add to high word).
bcamTsapRecvCall	read-only	Number of successful receive calls of all TSAPs since start of BCAM
bcamTsapRecvByteHigh	read-only	Number of bytes given from BCAM to all TSAPs since start of BCAM (high word, multiply by 4294967296).
bcamTsapRecvByteLow	read-only	Number of bytes given from BCAM to all TSAPs since start of BCAM (low word, add to high word).
bcamTsapSendCallCless	read-only	The number of successful send calls for connectionless data of all TSAPs since start of BCAM
bcamTsapSendByteClessHigh	read-only	The number of bytes given to BCAM-interface to send connectionless from all TSAPs since start of BCAM (high word, multiply by 4294967296).
bcamTsapSendByteClessLow	read-only	The number of bytes given to BCAM-interface to send connectionless from all TSAPs since start of BCAM (low word, add to high word).
bcamTsapRecvCallCless	read-only	The number of successful send calls for connectionless data of all TSAPs since start of BCAM

TSAP group

Object name	Access	Explanation
bcamTsapRecvByteClessHigh	read-only	The number of bytes received connectionless given from BCAM to all TSAPs since start of BCAM (high word, multiply by 4294967296).
bcamTsapRecvByteClessLow	read-only	The number of bytes received connectionless given from BCAM to all TSAPs since start of BCAM (low word, add to high word).
bcamTsapNumTable	read-only	The number of table entries.
bcamTsapMaxTsap	read-only	The maximum number of not predefined applications that may be opened simultaneously.
bcamTsapMaxTsapTask	read-only	The maximum number of not predefined applications that may be opened simultaneously by one task.
bcamTsapMaxCepTsap	read-only	The maximum number of connections a non system application may have opened concurrently.
bcamTsapRejTsap	read-only	The number of rejected tsap openings because of reached maximum (see bcamTsapMaxTsap).
bcamTsapRejTsapTask	read-only	The number of rejected tsap openings because of reached maximum for one task (see bcamTsapMaxTsapTask).
bcamTsapRejCepTsap	read-only	The number of connection requests rejected because the maximum number of connections for this tsap has been reached (see bcamTsapMaxCepTsap).
Start of the TSAP table:		
bcamTsapTabNumbers	read-only	The unique identifier for this application. Even if the same application is disabled and then reenabled, the identfier will get a new value.

TSAP group

Object name	Access	Explanation
bcamTsapTabState	read-only	The current state of this application. opening (1): the application enableing is in progress. open (2): the application is ready for requesting or accepting connections. closing (3): the termination of the application is in progress. closed (4): the application has been closed. Several bits may be set simultaneous.
bcamTsapTabDuration	read-only	The time in seconds the application was enabled.
bcamTsapTabDateApplEnable	read-only	The date of the application enableing. Format: YYYYMMDD
bcamTsapTabTimeApplEnable	read-only	The time of the application enableing. Format: HHMMSS
bcamTsapTabTypName	read-only	The type of name of this application. NeaName (1) : the name consists of 8-byte application name and 8-byte name of end system. SocketName (2): the name consists of up to 32-byte socket name. IsoName (3): the name consists of up to 78-byte ISO name.
bcamTsapTabName	read-only	The name of this application according the type of the name (bcamTsapTabTypName).
bcamTsapTabPortnumber	read-only	The port number used by this application (if known).
bcamTsapTabOsiTsel	read-only	The OSI transport selector of this application (if known).
bcamTsapTabNeaTsel	read-only	The NEA transport selector of this application (if known).
bcamTsapTabHost	read-only	The name of the host on which the application resides.

TSAP group

Object name	Access	Explanation
bcamTsapTabDiagnostic	read-only	The reason why the application was disabled (for diagnostic purposes only).
bcamTsapTabTsduSends	read-only	The number of packets to send given to BCAM at the TSAP by this application.
bcamTsapTabByteSends	read-only	The number of bytes to send given to BCAM at the TSAP by this application.
bcamTsapTabTsduReceiveds	read-only	The number of packets received at the TSAP by this application.
bcamTsapTabByteReceiveds	read-only	The number of bytes received at the TSAP by this application.
bcamTsapTabSendCallOverMaxs	read-only	The number of send calls of this application while BCAM limits have been reached.
bcamTsapTabLetterTimeouts	read-only	The number of packets that were deleted without being delivered to the application due to timeout of the letter timer.
bcamTsapTabInWaitBuck1Hist	read-only	The number of times the input wait time was in the time interval between zero and bcamGlobalInWaitLimit1 for meanwhile closed applications.
bcamTsapTabInWaitBuck2Hist	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit1 and bcamGlobalInWaitLimit2 for meanwhile closed applications.
bcamTsapTabInWaitBuck3Hist	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit2 and bcamGlobalInWaitLimit3 for meanwhile closed applications.
bcamTsapTabInWaitBuck4Hist	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit3 and bcamGlobalInWaitLimit4 for meanwhile closed applications.
bcamTsapTabInWaitBuck5Hist	read-only	The number of times the input wait time was longer than bcamGlobalInWaitLimit4 for meanwhile closed applications.

TSAP group

Object name	Access	Explanation
bcamTsapTabReactBuck1Hist	read-only	The number of times the application reaction time was in the time interval between zero and bcamGlobalReactLimit1 for meanwhile closed applications.
bcamTsapTabReactBuck2Hist	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit1 and bcamGlobalReactLimit2 for meanwhile closed applications.
bcamTsapTabReactBuck3Hist	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit2 and bcamGlobalReactLimit3 for meanwhile closed applications.
bcamTsapTabReactBuck4Hist	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit3 and bcamGlobalReactLimit4 for meanwhile closed applications.
bcamTsapTabReactBuck5Hist	read-only	The number of times the application reaction time was longer than bcamGlobalReactLimit4 for meanwhile closed applications.
bcamTsapTabInWaitBuck1	read-only	The number of times the input wait time was in the time interval between zero and bcamGlobalInWaitLimit1 for opened applications.
bcamTsapTabInWaitBuck2	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit1 and bcamGlobalInWaitLimit2 for opened applications.
bcamTsapTabInWaitBuck3	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit2 and bcamGlobalInWaitLimit3 for opened applications.

TSAP group

Object name	Access	Explanation
bcamTsapTabInWaitBuck4	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit3 and bcamGlobalInWaitLimit4 for opened applications.
bcamTsapTabInWaitBuck5	read-only	The number of times the input wait time was longer than bcamGlobalInWaitLimit4 for opened applications.
bcamTsapTabReactBuck1	read-only	The number of times the application reaction time was in the time interval between zero and bcamGlobalReactLimit1 for opened applications.
bcamTsapTabReactBuck2	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit1 and bcamGlobalReactLimit2 for opened applications.
bcamTsapTabReactBuck3	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit2 and bcamGlobalReactLimit3 for opened applications.
bcamTsapTabReactBuck4	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit3 and bcamGlobalReactLimit4 for opened applications.
bcamTsapTabReactBuck5	read-only	The number of times the application reaction time was longer than bcamGlobalReactLimit4 for opened applications.
bcamTsapTabTsduSendHists	read-only	The number of packets to send given to BCAM at the TSAP by this application for meanwhile closed connections.
bcamTsapTabByteSendHists	read-only	The number of bytes to send given to BCAM at the TSAP by this application for meanwhile closed connections.
bcamTsapTabTsduReceivedHists	read-only	The number of packets received at the TSAP by this application for meanwhile closed connections.

TSAP group

Object name	Access	Explanation
bcamTsapTabByteReceivedHists	read-only	The number of bytes received at the TSAP by this application for meanwhile closed connections.
bcamTsapTabSendCallOverMaxHists	read-only	The number of send calls of this application while BCAM limits have been reached for meanwhile closed connections.
bcamTsapTabLetterTimeoutHists	read-only	The number of packets that were deleted without being delivered to the application due to timeout of the letter timer for meanwhile closed connections.
bcamTsapTabFunction	read-only	Interfacefunctionality of TSAP. nea (1): functionality according proprietary NEA. iso (2): functionality according ISO IS 8072. streams (3): functionality according the one defined in the INTERNET.
bcamTsapTabCurrConn	read-only	Number of connections of this TSAP not in the connection state closed.
bcamTsapTabCloseConns	read-only	Number of connections of this TSAP that have been opened and are now in the connection state closing or closed.
bcamTsapTabClessSendBytes	read-only	Number of bytes sent connectionless by this application
bcamTsapTabClessRecvBytes	read-only	Number of bytes received connectionless by this application
bcamTsapTabClessSendCalls	read-only	Number of successful send calls for connectionless data
bcamTsapTabClessRecvCalls	read-only	Number of successful receive calls for connectionless data
bcamTsapTabOutbufTsdu	read-only	The number of packets to send connectionoriented that are currently buffered by BCAM for this application.
bcamTsapTabOutbufByte	read-only	The number of bytes to send connectionoriented that are currently buffered by BCAM for this application.

TSAP group

Object name	Access	Explanation
bcamTsapTabInbufTsdu	read-only	The number of packets to receive connectionoriented that are currently buffered by BCAM for this application.
bcamTsapTabInbufByte	read-only	The number of bytes to receive connectionoriented that are currently buffered by BCAM for this application.
bcamTsapTabOutbufTsduCless	read-only	The number of packets to send connectionless that are currently buffered by BCAM for this application.
bcamTsapTabOutbufByteCless	read-only	The number of bytes to send connectionless that are currently buffered by BCAM for this application.
bcamTsapTabInbufTsduCless	read-only	The number of packets to receive connectionless that are currently buffered by BCAM for this application.
bcamTsapTabInbufByteCless	read-only	The number of bytes to receive connectionless that are currently buffered by BCAM for this application.
bcamTsapTabClessTimeout	read-only	The number of packets discarded because the datagram timer elapsed.

TSAP group

6.2.6 Connection group

Object name	Access	Explanation
bcamCepCurrent	read-only	The number of connections in the state open.
bcamCepClosed	read-only	The number of connections that had reached the state open and are now in the state closing or closed.
bcamCepNotClosed	read-only	The number of connections in the states opening, open, or closing.
bcamCepRerouting	read-only	The number of times an alternative route has been tried.
bcamCepActiveTrials	read-only	The number of active trials of connection establishment.
bcamCepActiveTrialFailures	read-only	The number of active trials of connection establishment that failed.
bcamCepPassiveTrials	read-only	The number of passive trials of connection establishment.
bcamCepPassiveTrialFailures	read-only	The number of passive trials of connection establishment that failed.
bcamCepNumTable	read-only	The number of table entries.
Start of the Connection table:		
bcamCepTabProtocolClass	read-only	The protocol class used for this connection. local (1): the partner application resides in the same host. nea (2): the proprietary NEA protocol is used as transport protocol. iso (3): the ISO protocol is used as transport protocol. tcp (4): the TCP protocol is used as transport protocol.

Connection group

Object name	Access	Explanation
bcamCepTabConnectionNumbers	read-only	The unique identifier for this connection. Even if a connection is closed and then reopened between the same applications, the identfier will get a new value.
bcamCepTabDisconCommand	read-only	First word that contains the reason for the closing of the connection (for diagnostic purposes only).
bcamCepTabDisconInfoWord	read-only	Second word that contains the reason for the closing of the connection (for diagnostic purposes only).
bcamCepTabTsduSends	read-only	The number of packets to send given to BCAM at the TSAP for this connection.
bcamCepTabByteSends	read-only	The number of bytes to send given to BCAM at the TSAP for this connection.
bcamCepTabTsduReceiveds	read-only	The number of packets received by BCAM for this connection.
bcamCepTabByteReceiveds	read-only	The number of bytes received by BCAM for this connection.
bcamCepTabSendCallOverMaxs	read-only	The number of send calls at the TSAP for this connection when connection specific limits have been reached.
bcamCepTabLetterTimeouts	read-only	The number of packets that have been deleted without delivering to the application due to the timeout of the letter timer.
bcamCepTabOutbufTsduSend	read-only	The number of packets to send that are currently buffered by BCAM for this connection.
bcamCepTabOutbufByteSend	read-only	The number of bytes to send that are currently buffered by BCAM for this connection.
bcamCepTabOutbufMaxTsduSend	read-only	The maximum number of packets to send that will be buffered by BCAM for this connection.
bcamCepTabOutbufMaxByteSend	read-only	The maximum number of bytes to send that will be buffered by BCAM for this connection.

Connection group

Object name	Access	Explanation
bcamCepTabInbufTsdu	read-only	The number of packets to receive that are currently buffered by BCAM for this connection.
bcamCepTabInbufByte	read-only	The number of bytes to receive that are currently buffered by BCAM for this connection.
bcamCepTabInbufMaxTsduReceived	read-only	The maximum number of packets to receive that will be buffered by BCAM for this connection.
bcamCepTabInbufMaxByteReceived	read-only	The maximum number of bytes to receive that will be buffered by BCAM for this connection.
bcamCepTabPacketsDataSends	read-only	The number of packets sent by BCAM that contained at least one byte of user data.
bcamCepTabPacketsWindowSends	read-only	The number of packets sent by BCAM that contained only window information.
bcamCepTabPacketsDataReceiveds	read-only	The number of packets received by BCAM that contained at least one byte of user data.
bcamCepTabPacketsWindowReceiveds	read-only	The number of packets received by BCAM that contained only window information.
bcamCepTabGlobalZeroWindowSends	read-only	The number of packets sent by BCAM that contained zero window due to global shortage of ressources.
bcamCepTabConnectionZeroWindowSends	read-only	The number of packets sent by BCAM that contained zero window due to shortage of connection specific ressources.
bcamCepTabZeroWindowReceiveds	read-only	The number of packets received that contained zero window.
bcamCepTabRoundTripTime	read-only	The round trip time for this connection in milliseconds.
bcamCepTabRetransmitPacketsSends	read-only	The number of packets sent that contain at least one byte of data that have been already sent.

Connection group

Object name	Access	Explanation
bcamCepTabDetectedGapsReceiveds	read-only	The number of packets received where there is a gap between the highest already received byte and the lowest byte in the packet.
bcamCepTabDuplicatePacketsReceiveds	read-only	The number of packets received that contain at least one byte of user data which has already been received.
bcamCepTabErrorPacketsReceiveds	read-only	The number of packets received that were discarded because of weak protocol violations.
bcamCepTabConnectionState	read-only	The current state of the connection. opening (1): the connection establishment is in progress. open (2): the connection is ready for data transfer. closing (3): the connection is terminating. closed (4): the connection has been closed.
bcamCepTabApplicationNumber	read-only	The unique identifier of the local application the connection belongs to.
bcamCepTabRouteNumber	read-only	The unique identifier of the route that is used for the connection.
bcamCepTabDuration	read-only	The time the connection was in the state open.
bcamCepTabDateConnectionEstablishment	read-only	The date of the connection establishment. Format: YYYYMMDD
bcamCepTabTimeConnectionEstablishment	read-only	The time of the connection establishment. Format: HHMMSS

Connection group

Object name	Access	Explanation
bcamCepTabTypPartnerName	read-only	The type of name used for the partner application. neaName (1): the name consists of 8-byte application name and 8-byte name of end system. socketName (2): the name consists of up to 32-byte socket name. isoName (3): the name consists of up to 78-byte ISO name.
bcamCepTabPartnerName	read-only	The name of the partner application according the type of the name (bcamCepTabTypPartnerName).
bcamCepTabLocalName	read-only	The name of the local application according the type of the name (bcamCepTabTypPartnerName).
bcamCepTabTypeL4Addr	read-only	The type of layer-4 addresses used for this connection. portNumber (1): the address consists of 2-byte port number. neaTsel (2): the address consists of 8-byte NEA transport selector. osiTsel (3): the address consists of up to 33-byte OSI transport selector.
bcamCepTabL4AddrPartner	read-only	The layer-4 address of the partner application according the type of the address (bcamCepTabTypeL4Addr).
bcamCepTabL4AddrLocal	read-only	The layer-4 address of the partner application according the type of the address (bcamCepTabTypeL4Addr).
bcamCepTabPartnerEndsystem	read-only	The name of the host where the partner application resides.
bcamCepTabInWaitBuck1	read-only	The number of times the input wait time was in the time interval between zero and bcamGlobalInWaitLimit1.

Connection group

Object name	Access	Explanation
bcamCepTabInWaitBuck2	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit1 and bcamGlobalInWaitLimit2.
bcamCepTabInWaitBuck3	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit2 and bcamGlobalInWaitLimit3.
bcamCepTabInWaitBuck4	read-only	The number of times the input wait time was in the time interval between bcamGlobalInWaitLimit3 and bcamGlobalInWaitLimit 4.
bcamCepTabInWaitBuck5	read-only	The number of times the input wait time was longer than bcamGlobalInWaitLimit4.
bcamCepTabReactBuck1	read-only	The number of times the application reaction time was in the time interval between zero and bcamGlobalReactLimit1.
bcamCepTabReactBuck2	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit1 and bcamGlobalReactLimit2.
bcamCepTabReactBuck3	read-only	The number of times the application reaction time was in thr time interval between bcamGlobalReactLimit2 and bcamGlobalReactLimit3.
bcamCepTabReactBuck4	read-only	The number of times the application reaction time was in the time interval between bcamGlobalReactLimit3 and bcamGlobalReactLimit4.
bcamCepTabReactBuck5	read-only	The number of times the application reaction time was longer than bcamGlobalReactLimit4.
bcamCepTabOutProcBuck1	read-only	The number of times the outbound processing time was in the time interval between zero and bcamGlobalOutProcLimit1.

Connection group

Object name	Access	Explanation
bcamCepTabOutProcBuck2	read-only	The number of times the outbound processing time was in the time interval between bcamGlobalOutProcLimit1 and bcamGlobalOutProcLimit2.
bcamCepTabOutProcBuck3	read-only	The number of times the outbound processing time was in the time interval between bcamGlobalOutProcLimit2 and bcamGlobalOutProcLimit3.
bcamCepTabOutProcBuck4	read-only	The number of times the outbound processing time was in the time interval between bcamGlobalOutProcLimit3 and bcamGlobalOutProcLimit4.
bcamCepTabOutProcBuck5	read-only	The number of times the outbound processing time was longer than bcamGlobalOutProcLimit4.
bcamCepTabInProcBuck1	read-only	The number of times the inbound processing time was in the time interval between zero and bcamGlobalInProcLimit1.
bcamCepTabInProcBuck2	read-only	The number of times the inbound processing time was in the time interval between bcamGlobalInProcLimit1 and bcamGlobalInProcLimit2.
bcamCepTabInProcBuck3	read-only	The number of times the inbound processing time was in the time interval between bcamGlobalInProcLimit2 and bcamGlobalInProcLimit3.
bcamCepTabInProcBuck4	read-only	The number of times the inbound processing time was in the time interval between bcamGlobalInProcLimit3 and bcamGlobalInProcLimit4.
bcamCepTabInProcBuck5	read-only	The number of times the inbound processing time was longer than bcamGlobalInProcLimit4.
bcamCepTabMaxSendLen	read-only	The maximum number of bytes per send call that can be given to BCAM at the interface.
bcamCepTabMaxIndLen	read-only	The maximum number of bytes BCAM will announce per data indication.

Connection group

Object name	Access	Explanation
bcamCepTabLocalEndsystem	read-only	The name of the host where the local application resides.

Connection group

6.2.7 Route group

Object name	Access	Explanation
bcamRouteNumTable	read-only	The number of all routes known by BCAM
bcamRouteActive	read-only	The number of active routes
bcamRouteArpDefault	read-only	The ARP state to use as default when a route of type node (bcamRouteTabTyp) is activated. Several bits may be set simultaneous. off (1): no usage of ARP. on (2): periodic usage of ARP. quiet (4): ARP will only be used if no connection oriented data were transferred during the last ARP period.
bcamRouteRoutingReqIp	read-only	The number of forwarding requests for IP
bcamRouteRoutingReqIso	read-only	The number of forwarding requests for ISO
bcamRouteRoutingReqNea	read-only	The number of forwarding requests for NEA
bcamRouteSuccRoutingReqIp	read-only	The number of successful forwarding requests for IP
bcamRouteSuccRoutingReqIso	read-only	The number of successful forwarding requests for ISO
bcamRouteSuccRoutingReqNea	read-only	The number of successful forwarding requests for NEA
Start of the Route table:		
bcamRouteTabNumbers	read-only	The unique identifier for this route.
bcamRouteTabName	read-only	The name of this route.
bcamRouteTabNumNeaConn	read-only	The current number of NEA connections using this route.
bcamRouteTabNumIsoConn	read-only	The current number of ISO and TCP connections using this route.

Route group

Object name	Access	Explanation
bcamRouteTabMaxNeaConn	read-only	The maximum number of NEA connections allowed for this route.
bcamRouteTabMaxIsoConn	read-only	The maximum number of ISO and TCP connections allowed for this route.
bcamRouteTabBadNeaElems	read-only	The number of erroneous NEA fragments received on this route.
bcamRouteTabBadIsoElems	read-only	The number of erroneous ISO and TCP fragments received on this route.
bcamRouteTabMaxL4Conn	read-only	The maximum number of layer-4 connections per layer-3 connection for this route.
bcamRouteTabMaxUnackTpdu	read-only	The maximum number of unacknowledged TPDUs per layer-3 connection that may be sent over this route.
bcamRouteTabRouteNetTyp	read-only	Describes whether there are other intermediate routes necessary to be known by BCAM to reach the system described by this route. gatewayRouter (1): the system is reached via a gateway route. neaRouter (2): the system is reached via a NEA router route. intfIpRouter (4): the system is reached via an INTF or IP router route. int0Router (8): the system is reached via an INT0 router route. Several bits may be set simultaneous.
bcamRouteTabNumberIsoIpRouter	read-only	The number of the route which describes the INT0, INTF or IP router used to reach the system described by this route.
bcamRouteTabNumberNeaRouter	read-only	The number of the route which describes the NEA router used to reach the system described by this route.
bcamRouteTabNumberGateway	read-only	The number of the route which describes the gateway used to reach the system described by this route.

Route group

Object name	Access	Explanation
bcamRouteTabFunction	read-only	Describes the function of the route: onlyServerAccess (1): the route is used for access to a specified server port only. transportRoute (2): the route leads to an end system Several bits may be set simultaneous.
bcamRouteTabProcNetTyp	read-only	Describes whether there are other intermediate systems necessary to be known by BCAM to reach the system described by this route. gatewayProc (1): an intermediate gateway system is used. neaProc (2): an intermediate NEA system is used. isoIpProc (4): an intermediate ISO or IP system is used. Several bits may be set simultaneous.
bcamRouteTabMaxTsduLen	read-only	The maximum length of packets that can be sent over this route.
bcamRouteTabNameEndsystem	read-only	The name of the remote system described by this route.
bcamRouteTabIso4WindowTimer	read-only	The current value of the window timer for this route in seconds.
bcamRouteTabIso4RetransTimer	read-only	The current value of the retransmit timer for this route in seconds.
bcamRouteTabAckTimer	read-only	The current value of the acknowledgement timer for this route in seconds.
bcamRouteTabErrorRecovTimer	read-only	The current value of the error recovery timer for this route in seconds.
bcamRouteTabRejectTimer	read-only	The current value of the reject timer for this route in seconds.
bcamRouteTabExpedRetransTimer	read-only	The current value of the expedited retransmit timer for this route in seconds.
bcamRouteTabNameGateway	read-only	The name of the intermediate system where protocol transitions, if necessary for this route, are done.

Route group

Object name	Access	Explanation
bcamRouteTabNameX25Station	read-only	The name of the station which performs the X25 net access.
bcamRouteTabL3InputProfil	read-only	The input profile on the network layer. nea (1): proprietary NEAN protocol. int0 (2): protocol according ISO 8473 (inactive network layer). intf (3): protocol according ISO 8473. gateway (4): no network layer protocol used. ip (5): protocol according RFC 791.
bcamRouteTabTransState	read-only	The current transfer state of this route as viewed by transport layer. notReady (4): no packets can be sent. waitForTransferInit (1): route may be initialised. waitForReadyToTransfer (2): address resolution running ready (3): packets may be sent
bcamRouteTabOption	read-only	The currently valid options for this route (only valid for TCP). delayedAckAllowed (1): do not try do minimize the number of outgoing acknowledgements optimalSegmentSizeOn (2): do not try do minimize the number of incoming acknowledgements Several bits may be set simultaneous.

Route group

Object name	Access	Explanation
bcamRouteTabTyp	read-only	Describes how the remote system described by this route is reached. direct (1): the remote system is connected directly to the local system (e.g via DAST). remote (2): the remote system is reached via at least one router. node (3): the remote system is connected directly to ethernet or fddi node.
bcamRouteTabUsage	read-only	Indicates whether this route describes an intermediate system. noIntermediate (16): no intermediate route gatewayRouter (1): route describes an intermediate system used as gateway intfIpRouter (2) route describes an INTF or an IP router neaRouter (4): route describes a NEA router int0Router (8): route describes an INT0 router Several bits may be set simultaneous.
bcamRouteTabL3Subprofile	read-only	rfc1042 (1): a layer-3 subprotocol according RFC1042 is used on this route. netConnLess (2): a connectionless layer-3 protocol is used on this route. neaNetConn (4): a NEA net connection is used on this route. neattNetConn (8): a NEATT net connection is used on this route. Several bits may be set simultaneous.

Route group

Object name	Access	Explanation
bcamRouteTabCommandState	read-only	The current state of this route caused by route specific commands. included (1): a BCIN command was given successfully for this route. active (2): a BCACT command was given successfully for this route. Several bits may be set simultaneous.
bcamRouteTabChangeState	read-only	Describes how the route was defined or modified. generated (16): the route has been defined by KOGS and not been modified. dynamic (1): the route was created dynamically by a BCIN command. defByProtocol (2): the route was defined by routing protocol. switchByProtocol (4): the route has been switched by routing protocol. changed (8): the route has been modified. Several bits may be set simultaneous.
bcamRouteTabIso9542	read-only	Describes the ISO9542 processing, if this route is an INTF route eshReceived (1): end system hello received ishReceived (2): intermediate system hello received. iso8473QueryReceived (4): ISO 8473 query configuration request received. refreshRequired (8): refreshing of routing information is required. Several bits may be set simultaneous.

Route group

Object name	Access	Explanation
bcamRouteTabMaxNetLength	read-only	The maximum length of a packet used on layer 3 (only valid for FDDI routes). ethernet (1): the length will be limited to the maximum length for ethernet. fddi (2): the length will be limited to the maximum length for FDDI.
bcamRouteTabState2	read-only	Describes the state of the route viewed by routing protocols. silent (16): route not yet or no more used. testing (1): delivery problems on route detected. working (2): route working. waitAddressResolution (4): waiting for address resolution. lifetimeSupervisionFailed (8): lifetime supervision failed. Several bits may be set simultaneous.
bcamRouteTabDeactReason	read-only	The reason for the deactivation of the route. flush (1): an ISO9542 protocol element that requests the invalidation of the routing information has been received. supposedDown (2): the route is supposed to be down (derived from state of TCP protocol machine or ICMP message received). noArpReply (4): no arp reply has been received. Several bits may be set simultaneous.

Route group

Object name	Access	Explanation
bcamRouteTabSwitchType	read-only	The type of the last performed route switching. localRemote (1): the route has been changed from local to remote route. remoteLocal (2): the route has been changed from remote to local route. remoteRemote (4): the route has been changed from remote to remote route. lsapChanged (8): the route has been switched to another LSAPCB. lanAddrChanged (16): the lan address of the route has been changed. Several bits may be set simultaneous.
bcamRouteTabReasonCreation	read-only	Describes the reason for the automatic creation of the route. incomingData (1): the route has been created because of incoming data. outgoingData (2): the route has been created because of outgoing data. routingProtocol (4): the route has been created because of incoming routing protocol. routingData (8): the route has been created because of data to be routed. Several bits may be set simultaneous.
bcamRouteTabOrigLanAddress	read-only	The MAC address (if known) of the remote end system or the first intermediate system of this route as used when the route has been created.
bcamRouteTabLanAddress	read-only	The current MAC address (if known) of the remote end system or the first intermediate system of this route.

Route group

Object name	Access	Explanation
bcamRouteTabTypAddress	read-only	The type of address used for this route. nea (1): 2-byte (processor number and region number) intf (2): up to 20-byte full internet address streams (3): 4-byte IP address int0 (4): 6-byte MAC address
bcamRouteTabLocalAddr	read-only	The layer-3 address of the local system described by this route according to bcamCepTabTypeL4Addr.
bcamRouteTabRemoteAddr	read-only	The layer-3 address of the remote system described by this route according to bcamCepTabTypeL4Addr.
bcamRouteTabOutPacketsDatas	read-only	The number of packets sent over this route that contained at least one byte of user data.
bcamRouteTabOutPacketsFlowControls	read-only	The number of packets sent over this route that contained only flow control information.
bcamRouteTabInPacketsDatas	read-only	The number of packets received on this route that contained at least one byte of user data.
bcamRouteTabInPacketsFlowControls	read-only	The number of packets received on this route that contained only flow control information.
bcamRouteTabOutRetransPackets	read-only	The number of packets sent over this route that contained at least one byte of user data that were already sent.
bcamRouteTabInDetectedGaps	read-only	The number of packets received where there was a gap between the highest already received byte and the lowest byte in the packet.
bcamRouteTabInDuplicatedPackets	read-only	The number of packets received that contained at least one byte of user data that has already been received
bcamRouteTabInIncorrectPackets	read-only	The number of erroneous packets received on this route.

Route group

Object name	Access	Explanation
bcamRouteTabRoundTripTimeClosed	read-only	The average round trip time on this route for meanwhile closed connections.
bcamRouteTabRoundTripTimeCurrent	read-only	The average round trip time on this route for currently open connections.
bcamRouteTabArpReqSend	read-only	The number of ARP requests sent for this route.
bcamRouteTabArpRepSend	read-only	The number of ARP replies sent for this route.
bcamRouteTabArpReqRec	read-only	The number of ARP requests received for this route.
bcamRouteTabArpRepRec	read-only	The number of ARP replies received for this route.
bcamRouteTabIcmpReq	read-only	The number of ICMP ECHO requests sent for this route.
bcamRouteTabIcmpReply	read-only	The number of ICMP ECHO replies received for this route.
bcamRouteTabIcmpRedirect	read-only	The number of ICMP redirects received for this route.
bcamRouteTabSwitched	read-only	The number of route switchings performed for this route.
bcamRouteTabDown	read-only	The number of times the state of the route has changed to not working.
bcamRouteTabOspfHello	read-only	The number of OSPF HELLOs received for this route.
bcamRouteTabPacketNoConn	read-only	The number of packets received on this route that could not be assigned to a connection.
bcamRouteTabPacketInternDiscon	read-only	The number of packets received on this route that resulted in an abortion of a connection.
bcamRouteTabPacketBadProtocol	read-only	The number of packets received on this route that were invalid.
bcamRouteTabConnReqOut	read-only	The number of connection requests sent on this route.
bcamRouteTabConnReqOutAck	read-only	The number of connection requests sent on this route that have been acknowledged by the remote transport system.

Route group

Object name	Access	Explanation
bcamRouteTabConnReqOutRej	read-only	The number of connection requests sent on this route that have been rejected by the remote transport system.
bcamRouteTabConnReqIn	read-only	The number of connection requests received on this route.
bcamRouteTabConnReqInAck	read-only	The number of connection requests received on this route that have been acknowledged by the local transport system.
bcamRouteTabConnReqInRej	read-only	The number of connection requests received on this route that have been rejected by the local transport system.
bcamRouteTabDisconnOut	read-only	The number of disconnect requests sent.
bcamRouteTabDisconnOutAck	read-only	The number of disconnect requests sent that have been acknowledged by the remote transport system.
bcamRouteTabDisconnIn	read-only	The number of disconnect requests received.
bcamRouteTabDisconnInAck	read-only	The number of disconnect requests received that have been acknowledged by the local transport system.
bcamRouteTabNumberLink	read-only	The number of the interface that is used for this route.
bcamRouteTabArpFlag	read-only	The ARP state of the route. off (1): no usage of ARP. on (2): periodic usage of ARP. quiet (4): ARP will only be used if no connection oriented data were transferred during the last ARP period. Several bits may be set simultaneous.
bcamRouteTabNsduLen	read-only	The length of a network service data unit used on this route
bcamRouteTabMinNsduLen	read-only	The minimum maximal length of a network service data unit used on this route

Route group

Object name	Access	Explanation
bcamRouteTabMaxNsduLen	read-only	The maximum maximal length of a network service data unit used on this route

Route group

6.2.8 Interface group

Object name	Access	Explanation
bcamIfNumTable	read-only	The number of interface table entries.
Start of the Interface table:		
bcamIfTabNumbers	read-only	A unique identifier for the interface. Serves for references from other tables
bcamIfTabName	read-only	The name of this interface
bcamIfTabProfile	read-only	The layer-2 profile of this interface nealkp (1): proprietary link protocol with permanent dialog. nealke (2): proprietary event driven link protocol to BS2000 system. nealkeS (3): proprietary event driven link protocol to System 6000. nealkh (4): a HDLC balanced link protocol (duplex mode). llc1 (5): CSMA/CD link protocol. sinix (6): proprietary link protocol to SINIX port. fddi (7): FDDI link protocol.
bcamIfTabMnemonicWrite	read-only	The device mnemonic for write operations
bcamIfTabMnemonicRead	read-only	The device mnemonic for read operations
bcamIfTabLanAddress	read-only	The MAC address (if supported)

Interface group

Object name	Access	Explanation
bcamIfTabConfigUpdate	read-only	Describes whether assigned routes are allowed to be switched from this to an alternative interface. updateAllowed (4): switching allowed updateForbidden (1): switching forbidden primaryRequested (2): switch back of all routes original assigned to this interface requested Several bits may be set simultaneous.
bcamIfTabMaxLpdu	read-only	The maximum length of a link protocol data unit depending on bcamIfTabProfile. ethernet (1514): 1514 bytes fddi (4494): 4494 bytes atm (65535): 65535 bytes nealkh (65549): 65549 bytes nealkp (4113): 4113 bytes maxReass (16392): 16392 bytes
bcamIfTabL2Monitoring	read-only	The state of the layer-2 monitoring. on (1): layer-2 monitoring is running. off (2): layer-2 monitoring is not running.
bcamIfTabDevice	read-only	Describes whether the interface is an ESCON channel to channel adapter. esconCtc (1): the interface is an ESCON channel to channel adapter.
bcamIfTabAdminState	read-only	The administrative state of this interface requested by commands. excluded (4): no BCIN command given included (1): BCIN command given active (2): BCACT command given
bcamIfTabCurrentState	read-only	The actual working state of this interface none (8): interface not working waitForAct (1): activation in progress working (2): interface working waitForDeact (4): deactivation in progress Several bits may be set simultaneous.

Interface group

Object name	Access	Explanation
bcamIfTabMode	read-only	Describes the operational parameters of this interface stopModeOn (16): stop PDN in case of abnormal deactivation slowPollOn (8): automatic reactivation shall be tried shortWaitOn (4): stop activation after timeout multicastOn (2): evaluate multicasts broadcastOn (1): evaluate broadcasts Several bits may be set simultaneous.
bcamIfTabPortName	read-only	The port name of this interface
bcamIfTabLenTraceOut	read-only	The maximum length of layer-2 data to be traced in a trace entry for output.
bcamIfTabLenTraceIn	read-only	The maximum length of layer-2 data to be traced in a trace entry for input.
bcamIfTabNumRouteSwitchings	read-only	The number of route switchings performed for this interface
bcamIfTabTimeLastChange	read-only	The time interval (in seconds) since the last change of the working state of this interface
bcamIfTabMnemonicDiag	read-only	The device mnemonic for diagnosis.
bcamIfTabNumMulticastAddr	read-only	The number of multicast addresses enabled for this interface.
bcamIfTabMulticastAddr1	read-only	multicast address 1 or zero
bcamIfTabMulticastAddr2	read-only	multicast address 2
bcamIfTabMulticastAddr3	read-only	multicast address 3
bcamIfTabMulticastAddr4	read-only	multicast address 4
bcamIfTabMulticastAddr5	read-only	multicast address 5
bcamIfTabMulticastAddr6	read-only	multicast address 6
bcamIfTabMulticastAddr7	read-only	multicast address 7
bcamIfTabMulticastAddr8	read-only	multicast address 8
bcamIfTabMulticastAddr9	read-only	multicast address 9
bcamIfTabMulticastAddr10	read-only	multicast address 10

Interface group

Object name	Access	Explanation
bcamIfTabNumNeaAddress	read-only	The number of NEA addresses enabled for this interface.
bcamIfTabNeaAddress1	read-only	NEA address 1
bcamIfTabNeaAddress2	read-only	NEA address 2
bcamIfTabNeaAddress3	read-only	NEA address 3
bcamIfTabNeaAddress4	read-only	NEA address 4
bcamIfTabNeaAddress5	read-only	NEA address 5
bcamIfTabNeaAddress6	read-only	NEA address 6
bcamIfTabNumIpAddress	read-only	The number of IP addresses enabled for this interface.
bcamIfTabIpAddress1	read-only	IP address 1 or zero
bcamIfTabIpAddress2	read-only	IP address 2
bcamIfTabIpAddress3	read-only	IP address 3
bcamIfTabIpAddress4	read-only	IP address 4
bcamIfTabIpAddress5	read-only	IP address 5
bcamIfTabIpAddress6	read-only	IP address 6
bcamIfTabNumInt0Address	read-only	The number of INT0 addresses enabled for this interface.
bcamIfTabInt0Address1	read-only	INT0 address 1 or zero
bcamIfTabInt0Address2	read-only	INT0 address 2
bcamIfTabInt0Address3	read-only	INT0 address 3
bcamIfTabInt0Address4	read-only	INT0 address 4
bcamIfTabInt0Address5	read-only	INT0 address 5
bcamIfTabInt0Address6	read-only	INT0 address 6
bcamIfTabNumIntfAddress	read-only	The number of INTF addresses enabled for this interface.
bcamIfTabIntfAddress1	read-only	INTF address 1
bcamIfTabIntfAddress2	read-only	INTF address 2
bcamIfTabIntfAddress3	read-only	INTF address 3
bcamIfTabIntfAddress4	read-only	INTF address 4
bcamIfTabIntfAddress5	read-only	INTF address 5
bcamIfTabIntfAddress6	read-only	INTF address 6
bcamIfTabBytesOutHighs	read-only	The number of bytes sent (high word, multiply by 4294967296)

Interface group

Object name	Access	Explanation
bcamIfTabBytesOutLows	read-only	The number of bytes sent (low word, add to high word)
bcamIfTabBytesInHighs	read-only	The number of bytes received (high word, multiply by 4294967296)
bcamIfTabBytesInLows	read-only	The number of bytes received (low word, add to high word)
bcamIfTabIOsOutHighs	read-only	The number of IO operations for output (high word, multiply by 4294967296)
bcamIfTabIOsOutLows	read-only	The number of IO operations for output (low word, add to high word)
bcamIfTabIOsInHighs	read-only	The number of IO operations for input (high word, multiply by 4294967296)
bcamIfTabIOsInLows	read-only	The number of IO operations for input (low word, add to high word)
bcamIfTabOutputStops	read-only	The number of times BCAM sent a RNR (Receive Not Ready) HDLC frame.
bcamIfTabInputStops	read-only	The number of times BCAM received a RNR (Receive Not Ready) HDLC frame.
bcamIfTabIOErrorOuts	read-only	The number of IO errors for output operations
bcamIfTabIOErrorIns	read-only	The number of IO errors for input operations
bcamIfTabPacketsNotReceiveds	read-only	The number of packets not received due to receive not ready
bcamIfTabInPacketsLanHighs	read-only	The number of input packets (only valid for ethernet or FDDI) (high word, multiply by 4294967296)
bcamIfTabInPacketsLanLows	read-only	The number of input packets (only valid for ethernet or FDDI) (low word, add to high word)
bcamIfTabOutPacketsLanHighs	read-only	The number of output packets (only valid for ethernet or FDDI) (high word, multiply by 4294967296)
bcamIfTabOutPacketsLanLows	read-only	The number of output packets (only valid for ethernet or FDDI) (low word, add to high word)
bcamIfTabUnicastInHighs	read-only	The number of unicasts received (high word, multiply by 4294967296)

Interface group

Object name	Access	Explanation
bcamIfTabUnicastInLows	read-only	The number of unicasts received (low word, add to high word)
bcamIfTabUnicastOutHighs	read-only	The number of unicasts sent (high word, multiply by 4294967296)
bcamIfTabUnicastOutLows	read-only	The number of unicasts sent (low word, add to high word)
bcamIfTabMulticastInHighs	read-only	The number of multicasts received (high word, multiply by 4294967296)
bcamIfTabMulticastInLows	read-only	The number of multicasts received (low word, add to high word)
bcamIfTabMulticastOutHighs	read-only	The number of multicasts sent (high word, multiply by 4294967296)
bcamIfTabMulticastOutLows	read-only	The number of multicasts sent (low word, add to high word)
bcamIfTabErrorPacketIns	read-only	The number of erroneous packets received.
bcamIfTabErrorPacketOuts	read-only	The number of erroneous packets sent.
bcamIfTabDiscardIns	read-only	The number of packets received that were discarded.
bcamIfTabDiscardOuts	read-only	The number of packets that were discarded before sending.
bcamIfTabUnknownProtoIns	read-only	The number of packets received that were discarded because of unknown protocol.
bcamIfTabTraceState	read-only	The current state of the trace. stopped (8): the trace is not running. running (1): the trace is running. save (2): the trace is running and saved to a file. hold (4): the trace data will be kept in case of an error. Several bits may be set simultaneous.
bcamIfTabTraceNumberBuffer	read-only	The number of buffers used for the trace
bcamIfTabTraceBufferLen	read-only	The length of one trace buffer in bytes (rounded up to multiples of 256).

Interface group

6.2.9 Router group

Object name	Access	Explanation
bcamRouterNumTable	read-only	The number of table entries in the router table
Start of the Router table:		
bcamRouterTabIpLow	read-only	The lower IP address limit of the IP address range reachable via this router.
bcamRouterTabIpHigh	read-only	The upper IP address limit of the IP address range reachable via this router.
bcamRouterTabIpLocal	read-only	The local IP address used to this end-system.
bcamRouterTabIpRouter	read-only	The IP address of the router
bcamRouterTabNumRouter	read-only	The route number of the router.

Router group

6.2.10 Host group

Object name	Access	Explanation
bcamHostNumTable	read-only	The number of table entries in the host table
Start of the Host table:		
bcamHostTabNumber	read-only	The unique identifier for this host.
bcamHostTabName	read-only	Nea name of the host
bcamHostTabSocketName	read-only	Socket name of the host
bcamHostTabTyp	read-only	The type of the host. normal (1): the host is the BCAM standard host. virtual (2): the host is a virtual host.
bcamHostTabState	read-only	The current state of the host: active (1): the host is working. included (2): the host is known by BCAM. Several bits may be set simultaneous.

Host group

6.2.11 Mapping group

Object name	Access	Explanation
bcamMappingNumTable	read-only	The number of table entries in the mapping table
bcamMappingMaxMappings	read-only	The maximum number of mappings currently possible.
bcamMappingTabNumber	read-only	The unique identifier for this mapping.
bcamMappingTabType	read-only	The mapping type. local (1): mapping for a local application global (2): mapping for a remote application special (3): mapping for a local and a remote application
bcamMappingTabLocalNameType	read-only	The type of name used for the local application. neaName (1): the name consists of 8-byte application name and 8-byte name of endsystem. socketName (2): the name consists of up to 32-byte socket name. isoName (3): the name consists of up to 78-byte ISO name.
bcamMappingTabLocalName	read-only	The name of the local application according the type of the name
bcamMappingTabLocalTselNea	read-only	The NEA transport selector of the local application (if known)
bcamMappingTabLocalTselIso	read-only	The OSI transport selector of the local application (if known)
bcamMappingTabLocalPort	read-only	The port number used by the local application (if known)

Mapping group

Object name	Access	Explanation
bcamMappingTabRemoteNameType	read-only	The type of name used for the remote application. neaName (1): the name consists of 8-byte application name and 8-byte name of endsystem. socketName (2): the name consists of up to 32-byte socket name. isoName (3): the name consists of up to 78-byte ISO name
bcamMappingTabRemoteName	read-only	The name of the remote application according the type of the name
bcamMappingTabRemoteHost	read-only	The name of the host on which the application resides.
bcamMappingTabRemoteTselNea	read-only	The NEA transport selector of the remote application (if known)
bcamMappingTabRemoteTselIso	read-only	The OSI transport selector of the remote application (if known)
bcamMappingTabRemotePort	read-only	The port number of the remote application (if known)
bcamMappingTabRemoteRoute1	read-only	The name of the route to be used first during an active attempt to set up a connection.
bcamMappingTabRemoteRoute2	read-only	The name of the route to be used second during an active attempt to set up a connection.
bcamMappingTabRemoteRoute3	read-only	The name of the route to be used third during an active attempt to set up a connection.
bcamMappingTabRemoteRoute4	read-only	The name of the route to be used fourth during an active attempt to set up a connection.
bcamMappingTabRemoteRoute5	read-only	The name of the route to be used fifth during an active attempt to set up a connection.
bcamMappingTabRemoteRoute6	read-only	The name of the route to be used sixth during an active attempt to set up a connection.

Mapping group

Object name	Access	Explanation
bcamMappingTabRemoteRoute7	read-only	The name of the route to be used seventh during an active attempt to set up a connection.
bcamMappingTabRemoteRoute8	read-only	The name of the route to be used eightth during an active attempt to set up a connection.
bcamMappingTabRemoteX25CUD	read-only	User data to be transferred via an active network connection for X.25 connection setup.

Mapping group

6.3 FTP.MIB

The information supplied in the FTP-MIB is divided into three groups:

– Global data
– Traps
– FTP server data

6.3.1 Global data

Object name	Access	Explanation
ftpSubagentVersion	read-only	The version of the FTP subagent

Global data

6.3.2 Traps

Object name	Access	Explanation
ftpServerPort	read-only	The port number of the FTP server

Traps

6.3.3 FTP server data

Server table

Object name	Access	Explanation
ftpSubagentPort	read-only	The port number of the FTP server for the controll connection
ftpServerState	read-only	The current status of the FTP server
ftpServerActCon	read-only	The number of current connections
ftpServerProtocolFileSave	write-only	Save the log file of the FTP server
ftpServerFTACLevel	write-only	FTAC level for restarting an FTP server

Server table

Server param table

Object name	Access	Explanation
ftpServerParamMaxCon	read-write	The maximum number of parallel connections
ftpServerParamConTimeout	read-write	The timeout value for connections (in seconds)
ftpServerParamFTAC	read-only	The FTAC level 0/1/2
ftpServerParamFTACJob	read-write	The job class for data transfer (if FTAC level > 0)
ftpServerParamVersion	read-only	The version of the FTP server
ftpServerParamHostName	read-only	The name of the host on which the FTP server is executed

Server Param Table

Server trace table

Object name	Access	Explanation
ftpServerTraceDebug	read-write	The level of the FTP trace
ftpServerTraceSocTrace	read-write	The level of the FTP socket trace

Server trace table

Server connection table

Object name	Access	Explanation
ftpServerConnectIndex	read-only	The connection index: a unique value for each connection with values between 1 and *ftpServerParamMaxConn*
ftpServerConnectState	read-only	The status of the connection (connected, logged-in, login complete, logoff in progress, logoff completed)
ftpServerConnectUsid	read-only	User ID under which the user logged in
ftpServerConectHost	read-only	Name of the host on which the FTP client is running

Server connection table

Glossary

address family

An address family comprises addresses with the same address structure.

agent

The agent is also known as the management agent. It is the implementation of a management protocol that exchanges management information with a *management station*. An agent is a program that runs on a *device* and reports the current information about the device to a manager or corresponding management application.

ATM

Asynchronous Transfer Mode: method standardized by *CCITT* which, with the aid of cell relay technology, can implement high-speed *WANs* and is suitable for transmitting data, voice, and moving images.

bar

A bar is the graphic representation of an x-value and a function expression involving one or more *MIB variables*.

bar display function

A bar display function consists of one or more displays, each of which represent a *bar* or *group of bars*.

bar definition

A bar definition consists of a pair of *x-coordinates* and *y-coordinates*. One *bar* is displayed for each pair of coordinates.
The definition of a *group of bars* is also possible.

bar chart

The graphic representation of periods of time and relative proportions.
A bar chart is used to define or modify the individual *bar definitions* of the chart.
In a bar chart, you can display one or more bar definitions simultaneously.

CCITT

Comité Consultatif International Télégraphique et Téléphonique (International Telegraph and Telephone Consultative Committee), a subagency of the UN that issues definitions in the field of telecommunications services (renamed *ITU-TSS* in 1993).

communication architecture

Model of the communication world.

communications domain

Address and *protocol families* are collected together into a communications domain. Communications domains are used to group together the common characteristics of processes which communicate over *sockets*.

connection endpoint (CEP)

Each *MIB table instance* describes a connection endpoint (CEP), e.g. the representative of a connection-oriented communication relationship.
A CEP can be uniquely identified by its table index. Table instances with the same name but different indices indicate that a communication relationship has been set up more than once. If these are not *parallel connections*, only one CEP can have an “open” status.

CSMA/CD

Carrier Sense Multiple Access / Collision Detection. A protocol defined in *IEEE* 802.3 and *ISO* 8802-3 for *LANs*. *Ethernet* and 802.3 are similar, but both use the CSMA/CD protocol. For this reason, the two terms are often used synonymously.

curve

A curve is the graphic representation of a function expression involving one or more *MIB variables* over time.

curve display function

A curve display function consists of one or more displays, each of which represent a *curve*.

curve definition

A curve definition consists of a function expression involving one or more *MIB variables*.

datagram

Term for messages that are sent in connectionless communication. There is no guarantee that datagrams will reach the receiver at all, that they will arrive in the correct order, or that they will not arrive more than once.

device

A network system, *router*, hub or other addressable equipment within the network.

de-facto standard

Standard not created by an acknowledged standardization body but generally accepted. May be transformed into an internationally agreed standard.

de-jure standard

International standard issued by *ISO* oder *CCITT*, or a national standard issued by a national standardization body.

display function

Display functions represent periodically queried values graphically.
A display function has a name and can contain a *graph* (line diagram) or *bar chart*.

domain

Area of a network in which similar or coordinated functions and methods that are controlled by an entity are valid for a specific range of functions.

element

The representation of a *vector* in certain diagrams is referred to as an element and is identified by a name, the *legend*.

Ethernet

A *LAN* introduced by XEROX that is based on yellow cable and uses *CSMA/CD* as a transmission method. Similar to an *IEEE* 802.3-LAN.

FDDI

Fiber Distributed Data Interface: a method for *LANs* defined in ISO 9314 that is similar to a *token ring* with a higher speed.

first-hop router

The *router* closest to your own system on a *route*.

front-end processor

Processor connected to the I/O channel of a host system, which supports host communication and handles communication protocols.
Examples: TD960 for BS2000

graph

is used to define or modify the individual *curves* on a line diagram.

group of bars

A group of bars is a set of *bars* defined within a display which usually belong together from a logical point of view.
A group of bars can be labeled using a *legend*.

host

see also *virtual host*

IEEE

Institute of Electrical and Electronic Engineers: US association of engineers which has done foundational work mainly in the *LAN* field, which has defined standards in the past, and which continues to do so.

industry standard

Synonymous with *de-facto standard*: *standard* which is generally accepted in a particular industry because of market forces.

instance

see *MIB table instance*

interface

is the connection of a data processing system to a network. Several interfaces can be connected to the same *LAN subnetwork*.
Each *MIB table instance* in the interface table describes an interface which provides physical access to the network.
An interface is uniquely identified by its *MIB table index* or its name.

Internet

The name used for a large number of interconnected networks that use the Internet protocol.
Communication architecture characterized by its use of *TCP* and *IP* that resulted from the ARPA network in the USA. Expansions are controlled by the IAB by means of the *RFC* process.

IP

Internet Protocol: connectionless protocol of the *internet* architecture.

IP address

The IP address is a network-wide unique address for a host.
Representation of a connection point in the *Internet* (32 bits).

ISO

International Organization for Standardization

ITU-TSS

Telecommunications Standards Sector, with the same functions as the former *CCITT*.

LAN

Local Area Network: originally a high-speed, short-range network. Today, also any network with a long range that operates as per *CSMA/CD*, *Token Ring* or *FDDI*.

legend

The representation of a *vector* in certain diagrams is referred to as an *element* and is identified by a name, the legend.

MIB

MIB stands for “Management Information Base“. This designates a data model that describes the network elements to be administered with network management (managed nodes), in an abstract form. This data model consists of the formal description of *object types* (object classes) that are constructed according to the RFC1157 conventions.
The object types can be arranged in tables. Each row corresponds to a *MIB table instance*, each column to a *MIB table variable*.

MIB-II

MIB-II is a standard MIB whose use is obligatory in the *Internet*. It offers an adequate data model for managing *devices*. MIB-II is standardized and is defined in RFC1213. It is an extension of MIB-I (RFC1156).

MIB table index

The MIB table index is used to uniquely identify a certain *MIB table instance* (*route*, *interface*, *router* or a *host*). The MIB table index can be the row number in a MIB table, for example.

MIB table instance

Object instance of an *object type* which is structured as a table.

MIB table variables

are MIB variables which are located in a table. They make up the rows in a table and can be used as *search criteria* for a *search set*.

MIB variable

A variable which is identified by a name and which can be referenced.

NEA

Name of the network architecture from Siemens.

network management station

A system in the network on which TransView SNMP, Unicenter TNG or a similar application runs.

object instance

Represents the characteristics (attribute values) of a *device*. The entities are managed by the device *agents*.
The object instance is specified by the instance identifier or index.

object type

A class of object entities of the same type, that is defined by a formal description. There may be one or more entities on one *device* for an object type. The object type is in the form of a table if more than one instance is possible for an object type on one device. Each row of this table represents one object instance and the columns the attributes of the object type.

OSI

Open Systems Interconnection: *communication architecture* defined in *ISO* 7498; adopted by *CCITT* as Rec. X.200.

parallel connection

There is normally only one connection between two *connection endpoints* (*CEPs*). Within the framework of the *OSI communication architecture*, however, it is possible for several connections to be set up between two CEPs. These connections are referred to as parallel connections.

port number

In *LAN* networks set up according to the *TCP/IP* standard, the 2-byte port number can be used to address a specific application unambiguously. Certain port numbers are reserved network-wide for standard applications. The combination of port number (*socket*) and host number (*IP address*) uniquely identifies the sender or recipient of a message within a network.

protocol family

A protocol family defines a set of protocols which implement the *socket* types in the *domain*.

reference variable

Reference variables can be defined for a MIB table in TV-BMBS2 (management application for BCAM). The values of the reference variables of a *MIB table instance* are stored locally during a search operation via the network. If another search is performed which uses one of these reference variables as a *search criteria*, local storage is searched first. If there is a hit, the values of all the *MIB variables* specified as search criteria are fetched via the network and compared.

Once the search in the local storage has been completed, a search operation is performed via the network starting with the highest table index in the storage. A search operation via the network is only performed if the search criteria do not include any of the reference variables.
Variables which remain the same for the entire life of an MIB table instance, e.g. name, address, table index, references to table instances of other tables, are suitable for use as reference variables.

RFC

Request for Comments: method used in *Internet* for commenting on recommended *standards*, definitions, or reports; also the name of a document issued by means of this process. The series of documents that describe the Internet protocol and related standards.

route

The logical data path between two systems is referred to a a rout. The route is defined by a pair of network addresses, the local and the remote address, which are used as either the source address or target address (depending on the direction) for *datagrams* exchanged via the network. More emphasis is placed on a logical path through the network than on taking physical factors into consideration.
Each *MIB table instance* in the route table describes a route.
Each route is assigned an *interface*, which provides access to the network.
A route can be assigned auxiliary routes such as, for example, the route to the *first-hop router*.
A route is uniquely identified by its *MIB table index* in the route table or its name.
It is also possible to have several routes to another system.

router

Element in a network that resides between networks and forwards message streams through the network while handling routing, flow control, addressing, and other functions. Operates on Layer 3 of the *OSI* model.
Each *MIB table instance* in the router table describes a main router or a virtual router.
A router is uniquely identified by its *MIB table index* in the router table or its name.

search criteria

Search criteria are the *MIB table variables* used for a search.

search logic

The search logic defines how the individual *search criteria* are linked.
By default, all the search criteria must be satisfied.

search set

A search set describes a set of *MIB table instances* using their common characteristics. Search sets can be used for a number of purposes. They provide the basis for search operations within the MIB tables. In order to perform complex search operations in different tables, the MIB must have a special structure. It must, for example, be possible to reference a table in the MIB from another table in the MIB.
Each search set has a name and comprises a set of *search criteria* from one or more MIB tables, the *search logic* and the corresponding *search values*.
Search sets are MIB-table-specific definitions.

search value

Search values are possible values for these variables (named values, i.e. values with symbolic names, are possible for integer variables). If you specify several possible values, a *search criteria* is satisfied if the actual value corresponds to at least one search value.

SNA

Systems Network Architecture, IBM's communication architecture.

SNMP

SNMP stands for „Simple Network Management Protocol". SNMP is a standard protocol defined in *Internet* for *TCP/IP* for the transfer of management informations and for network management in TCP/IP networks.

socket

A socket is a basic component for developing communications applications by forming a communications endpoint. It can be assigned a name via which the socket can then be accessed and addressed.
Each socket belongs to a specific type and has at least one affiliated process. Several related processes can use the same socket and a process can also have connections to several sockets.
A socket belongs to a specific *communications domain*. Communications domains are used to group together the common characteristics of processes which communicate over sockets.

standard

Document containing definitions and requirements in order to fulfill a well-specified function or purpose. See also *de-facto standard*, *industry standard*, *de-jure standard*.

subarea (SNA)

Area of an SNA network that is assigned to a type-5 or type-4 node.

subnetwork

Technically or administratively homogeneous part of a network.

tab

A tab is one of the elements of the graphical user interface for TV-BMBS2 (management application for BCAM). The contents of only ONE tab are visible on the screen at any one time; only the names of the other tabs can be seen. When you click on the name of a tab, this tab is moved to the front and its contents are visible.
A tab is used, for example, to define the print names for the variables on the tab for any group or table defined in the *MIB*. A tab is available for each MIB table from which search criteria can be selected.
Tabs are also used to define *bar display functions* for a current function or for defining *curve display functions* for a current function.
The tabs Groups, Overview and Tables also exist for each MIB. The tabs Groups and Tables allow you to define group and table names, while Overview contains all the variables from all the groups (but no table variables).

TCP

Transmission Control Protocol: transport protocol in the *Internet* architecture.

TCP/IP

TCP/IP stands for “Transmission Control Protocol/Internet Protocol“, i.e. the *Internet* protocol. A number of rules that define how systems communicate with each other in an open (not manufacturer-bound) environment. This is normally a large communication infrastructure (Internet).

token ring

Technique used in token ring *LANs*, by which a token circulates in a ring-shaped LAN and regulates the transmission rights of the different stations.

trap

Under SNMP, traps are problem reports that are sent automatically by a device agent.

Transport Service Access Point (TSAP)

Each *MIB table instance* of the TSAP table describes a communications application (transport service access point, TSAP).
A TSAP is uniquely identified by its table index in the TSAP table. Table instances with the same name but different indices indicate that a communications application has been closed and opened again under the same name.

variable

Under *SNMP*, a variable is the result of linking an *object instance* name with an assigned value.

vector

is a physical or mathematical entity. It is represented by an arrow in a systematically structured table containing all possible vectors.

virtual host

Formerly a large data processing system that required a *front-end processor* for communication. Now designates BS2000 systems or MVS systems and others. In a figurative sense, a (virtual) host designates a carrier system for applications which is addressable from the network. A data processing system can include several (virtual) hosts.
Each *MIB table instance* of the host table describes either the main host or a virtual host. A host is uniquely identified by its *MIB table index* or its name.
The first host generated is referred to as the standard host. The second host and any other hosts generated are referred to as virtual hosts. The number of virtual hosts is limited only by *NEA* addressing capabilities.

WAN

Wide Area Network: public or private network that bridges long distances and, unlike *LANs*, operates at a relatively low speed with a higher error rate. In *ATM* networks, for example, these two characterizations no longer apply.

x-coordinate

An x-coordinate can consist of a fixed string or the current value of a *MIB variable*.

y-coordinate

A y-coordinate consists of a function expression which can involve one or more *MIB variables*.

Related publications

***open*Net Server V2.0** (BS2000/OSD)
BCAM V16.0A Volume 1
User Guide

Target group
The manual is intended for network planners, generators and administrators who define BS2000 systems.
Contents
BCAM Volume 1 describes BCAM itself, how it is embedded in TRANSDATA and TCP/IP and ISO networks, plus generation and administrative activities.
Generation examples illustrate the description.
Additionally BCAM tools for generation and diagnosis are described.

***open*Net Server V2.0** (BS2000/OSD)
BCAM V16.0A Volume 2
Reference Manual

Target group
The manual is intended for network operators, generators and administrators who define BS2000 systems.
Contents
BCAM Volume 2 is based on Volume 1 and describes in detail the BCAM commands required for generation and operation.
The KOGS macros required for static generation are introduced and the BCAM messages are listed.

***inter*Net Services V2.0** (BS2000/OSD)
Administrator Guide

Target group
This manual is intended for network planners, generators and administrators who wish to use Internet Services in BS2000/OSD.
Contents
The manual describes the functionality of the Internet Services
BOOTP/DHCP, TFTP, DNS, FTP, LDAP and NTP in BS2000/OSD. It also covers the installation, administration, operation, and logging and diagnostic options of the individual components, as well as the FTP exit and the TELNET exits.

***inter*Net Services V2.0** (BS2000/OSD)
User Guide

Target group
This manual is intended for users and network planners, generators and administrators who wish to use Internet Services in conjunction with BS2000/OSD.
Contents
The manual introduces the components of *inter*Net Services. It contains a detailed description of FTP, the FTAC interface for FTP and TELNET. Network administrators require this manual as a supplement to the Administrator Guide.

***inter*Net Value Edition V1.0B** (BS2000/OSD)
User Guide

Target group
This manual is intended for network planners, generators and administrators who wish to use Mail Service in BS2000/OSD.
Contents
*inter*Net Value Edition is a supplement to *inter*Net Services that is available free of charge. The manual introduces the components of *inter*Net Value Edition and provides information on the installation, administration and operation of Mail Service in BS2000/OSD.

***open*Net Server** (BS2000/OSD)
IPv6 Introduction and Conversion Guide, Stage 1
User Guide

Target group
This manual is intended for everyone responsible for deciding as to the introduction of IPv6 in BS2000/OSD, as well as anyone using the IPv6 functionality on BS2000/OSD mainframes or planning to install IPv6 in BS2000/OSD.
Contents
The manual explains the commercial and technical foundations of IPv6. In addition, it describes the transition from IPv4 to IPv6 with the aid of examples and outlines the current status of the implementation of IPv6 in BS2000/OSD. Detailed information on “IPv6 addressing” and “DNS utilization” can be found in the appendix.

DCAM (TRANSDATA)
Program Interfaces
Reference Manual

Target group
– Managers
– Application planners
– Programmers
– System and network administrators

Contents
Description of the Data Communication Access Method DCAM

DCAM (TRANSDATA)
Macros
User Guide

Target group
Programmers of DCAM ASSEMBLER programs

Contents
– Special techniques when using DCAM macros
– DCAM macros, arranged according to functions
– Catalog of all DCAM macros

DCAM (TRANSDATA)
COBOL Calls
User Guide

Target group
Programmers of DCAM COBOL programs

Contents
– Special techniques for the use of DCAM COBOL calls, data structures and communication areas
– DCAM COBOL calls, arranged according to their functions
– Examples, programs and program drafts

PDN-GA
(PDN)
Generating a Data Communication System
User Guide

Target group
The manual addresses generators of data communication systems, and network and system administrators.
Contents
The manual describes network generation from the PDN viewpoint. The KOGS macros and generation procedures are explained, and supplemented by a number of generation examples and also information on generation.

SNMP Management V5.0
SNMP Management for BS2000/OSD
User Guide

Target group
The manual addresses network administrators/operators and system administrators who wish to integrate a BS2000 system in SNMP-based management or operate such a system.
Contents

This manual describes how SBA-BS2, SSC-BS2, SSA-SM2-BS2 and SSA-OUTM-BS2 are embedded in BS2000/OSD, the installation and configuration procedures required to enable operation, and actual system operation. The Agents and their MIBs which are required for monitoring are dealt with in detail. Installation and configuration of the relevant management applications on the Unicenter TNG, TransView SNMP and HP OpenView management platforms are also described.
Further central topics of the manual are access to management information via the World Wide Web, and the Trap Server for Solaris and Reliant UNIX.

IMON (BS2000/OSD)
Installation Monitor
User Guide

Target group
This manual is intended for systems support staff of the BS2000/OSD operating system.
Contents
The manual describes the installation and administration of BS2000 software using the IMON installation monitor and its three components IMON-BAS, IMON-GPN and IMON-SIC. Installation (standard and customer-specific) using the component IMON-BAS for systems with BS2000-OSD V2.0 and as of BS2000-OSD V3.0 is described in detail with the aid of examples in two separate chapters.

JV (BS2000/OSD)
Job Variables
User Guide

Target group
The manual addresses both nonprivileged users and systems support.
Contents
The manual describes management and possible uses of job variables. The command descriptions are divided according to function areas. The macro calls are described in a separate chapter.

CMX(BS2000)
Communication Method in BS2000 User Guide

Target group
Programmers of transport service (TS) applications
Contents
CMX (BS2000) offers application programs a uniform interface to the transport services. You can use CMX (BS2000) to create application programs which can communicate with other applications independently of the transport system.

XHCS
(BS2000/OSD)
8-Bit Code Processing in BS2000/OSD
User Guide

Target group
Users of the DCAM, TIAM and *open*UTM access methods, system support staff, and users migrating from EHCS to XHCS.
Contents
XHCS (Extended Host Code Support) is a software package of BS2000/OSD that lets you use extended character sets in conjunction with 8-bit terminals. XHCS is also the central source of information on the coded character sets in BS2000/OSD. XHCS replaces EHCS.

VTSU V11.0A
Virtual Terminal Support
User Guide

Target group
Users of the DCAM and TIAM access methods and of UTM, and also the system and network administrator.
Contents
VTSU (Virtual Terminal Support) is a software product of the BS2000 operating system. It implements a virtual line terminal. A virtual terminal permits programming that is independent of the physical characteristics of the terminal in question.

Ordering manuals

Please apply to your local office for ordering the manuals.

Index

Contents

*open*Net Server V2.0, *inter*Net Services V2.0 (BS2000/OSD)
SNMP Management for *open*Net Server and *inter*Net Services

User Guide

Target group
This manual is intended for network and system administrators wishing to use SNMP-based network and system management.

Contents
The manuals contains detailed descriptions of the MIBs delivered with *open*Net Server, the FTP-MIB delivered with *inter*Net Services, and the installation and operation of the subagents. Operation of the BCAM Manager is described in detail in a separate chapter.

Edition: February 2001

File: net_snmp.pdf

Copyright © Fujitsu Siemens Computers GmbH, 2001.

All rights reserved.
Delivery subject to availability; right of technical modifications reserved.

All hardware and software names used are trademarks of their respective manufacturers.

Fujitsu Siemens computers GmbH
User Documentation
81730 Munich
Germany

Fax: (++49) 700 / 372 00000

e-mail: manuals@fujitsu-siemens.com
http://manuals.fujitsu-siemens.com

Comments
Suggestions
Corrections

Submitted by

Comments on *open*Net Server V2.0, *inter*Net Services V2.0
SNMP-Management for *open*Net Server and *inter*Net Services

Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format *…@ts.fujitsu.com*.

The Internet pages of Fujitsu Technology Solutions are available at *http://ts.fujitsu.com/*...
and the user documentation at *http://manuals.ts.fujitsu.com*.

Copyright Fujitsu Technology Solutions, 2009

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form *…@ts.fujitsu.com*.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter *http://de.ts.fujitsu.com/*..., und unter *http://manuals.ts.fujitsu.com* finden Sie die Benutzerdokumentation.

Copyright Fujitsu Technology Solutions, 2009