A MITEL
PRODUCT
GUIDE

Mitel OpenScape Fault Management

Mitel OpenScape Fault Management V13 Enterprise MIB Definition Editor

User Guide

10/2025

A31003-F30D0-U101-01-7619

Mitel

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Europe Limited. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos, and graphics (collectively “Trademarks”) appearing on Mitel’s Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively “Mitel), Unify Software and Solutions GmbH & Co. KG or its affiliates (collectively “Unify”) or others. Use of the Trademarks is prohibited without the express consent from Mitel and/or Unify. Please contact our legal department at iplegal@mitel.com for additional information. For a list of the worldwide Mitel and Unify registered trademarks, please refer to the website: http://www.mitel.com/trademarks.

Contents

Contents

1 Preface

This chapter discusses the following aspects:

- purpose of this guide
- audience of this guide
- terminology
- organization of this guide
- conventions used in this manual

1.1 Purpose

This manual gives an introduction into the work with the Enterprise MIB Definition Editor.

1.2 Audience

This guide is addressed to system administrators who want to learn how to create EPM definitions by using the Enterprise MIB Definition Editor. The Enterprise MIB Definition Editor can be started within the OpenScape FM Client if the Enterprise MIB plugin has been initialized. For a better understanding the users of the Enterprise MIB Definition Editor should familiarize themselves with the Enterprise MIB plugin for the OpenScape FM Desktop (see *Enterprise MIB Plugin User Guide*).

1.3 Terminology

- **OpenScape FM** is a shortcut for OpenScape Fault Management.
- **Server** means the OpenScape FM Server, i.e. the server where OpenScape FM Desktop with the Enterprise MIB Definition Editor has been installed.
- **Client** means the OpenScape FM Client, usually a web browser where OpenScape FM has been started.
- **Desktop** means the OpenScape FM Desktop.
- **EPM** is a shortcut for the Enterprise MIB.
- **EPM Editor** is a shortcut for the Enterprise MIB Definition Editor.
- **EPM Definition** means an Enterprise MIB Plugin Definition.

1.4 Organization of this Guide

This guide is organized as follows:

- *Chapter 1, "Preface"* explains the structure of the manual.
- *Chapter 2, "Introduction"* provides a brief explanation what the Enterprise MIB Definition Editor is.
- *Chapter 3, "Work with the Enterprise MIB Definition Editor"* describes the different functions of the Enterprise MIB Definition Editor.
- *Appendix A, "Installation of the Enterprise MIB Definition Editor"* describes the installation of the EPM-Editor as a stand alone program.

1.5 Conventions Used in this Manual

The following font conventions are used in this document:

Bold Font: Indicates that a word is a new or important term.

Example: **Proxy Agent** or **OK**.

Bold Computer Font: Indicated data to be entered by the user.

Example: **`java`**.

Computer Font: Indicates computer output, including UNIX prompts, an explicit directory or a file name.

Example: `prompt%.`

Italics: Indicates a reference to another manual or to a different section within the current manual.

Example: see *Enterprise MIB Plugin User Guide.*

Italic type is also used for emphasis.

Example: *All* users will be affected.

2 Introduction

The Enterprise MIB Definition Editor can be used to create or modify EPM Definition Files for the HiPath Enterprise MIB Plugin. The ASN.1 definition for SNMP MIBs serves as the base for the creation of these definition files. The Enterprise MIB Plugin for OpenScape FM provides an easy and seamless integration of SNMP MIB instances into the OpenScape FM. The following aspects of the SNMP MIB integration can be configured by the EPM Editor.

- **Discovery and Representation of the corresponding IP node:**
  This can be used to define how a certain SNMP MIB instance can be identified for an IP node and how/where the IP node shall be displayed in OpenScape FM.
- **Definition of Browsers for a MIB:**
  This can be used to define which information from a MIB shall be displayed in a browser for tables or in a browser for scalar values.
- **Definition of Events within the Event Browser (Trap Handling):**
  This can be used to define the priority of an event which describes the arrival of an SNMP trap and how it should be displayed in OpenScape FM.
- **Definition of the Object State for the MIB Object:**
  This can be used to define how the state of the MIB object in the OpenScape FM will be calculated.
- **Localization**:
  Resource files support the localization (language depending representation) of identifiers and texts.

The definition of the different aspects of the EPM Editor will be described in the following chapters.

3 Work with the Enterprise MIB Definition Editor

3.1 Starting the Editor

The Enterprise MIB Definition Editor can be started through the OpenScape FM Client, if the Enterprise MIB plugin has been installed (see *Section 3.2, "Integration into OpenScape FM"*). Details about the installation of the Enterprise MIB Plugin can be found in the Enterprise MIB Plugin User Guide.

The editor only accepts existing EPM definition files which have been uploaded within the Enterprise MIB plugin

3.2 Integration into OpenScape FM

The Enterprise MIB definition editor can be opened from within the OpenScape FM by using the main menu entry **SNMP->Enterprise MIB->MIB Defintions** to open the ‚MIB Definitions' window. After this the button **Open** in Editor can be used to open the Enterprise MIB definitions editor (see separate User Guide) for a selected MIB definition. The button is only active, if a single definition is selected within the table.

3.3 Structure of the Editor

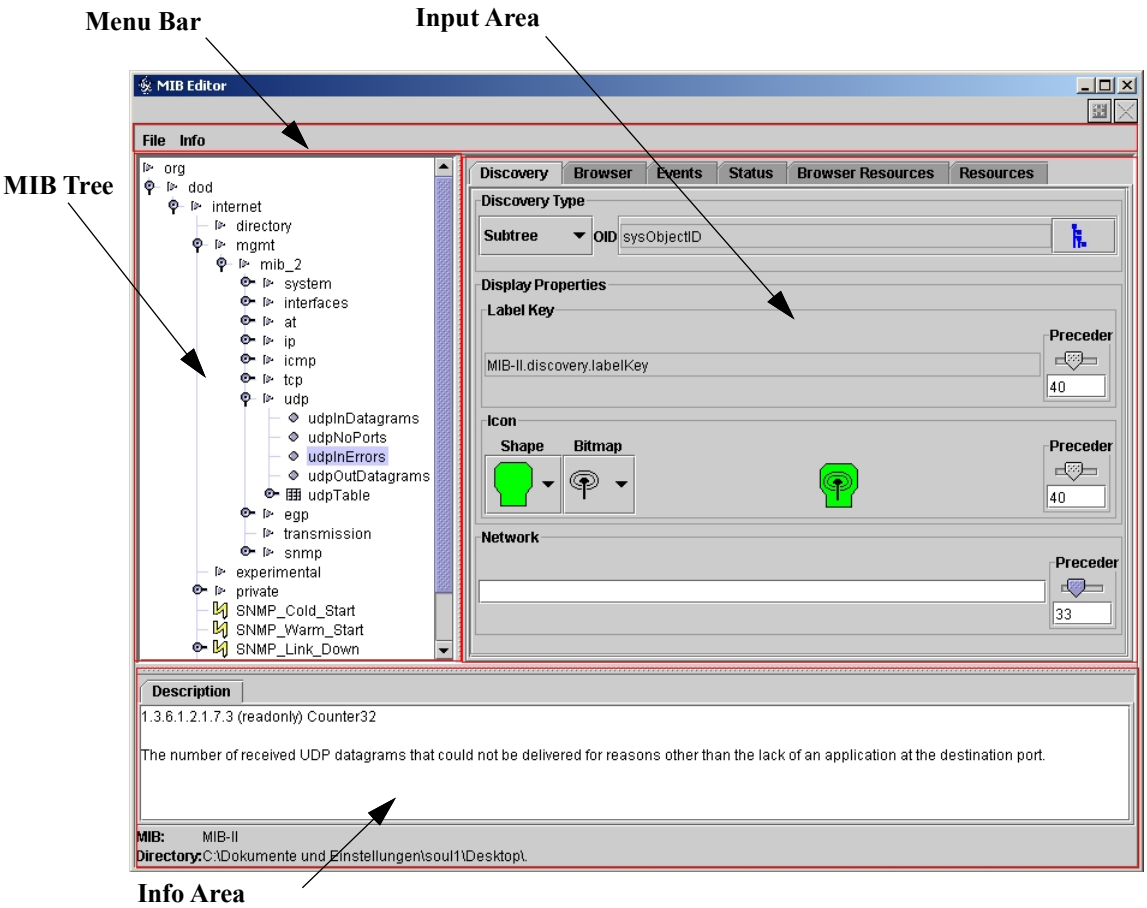


Figure 1 Enterprise MIB Definition Editor

The Enterprise MIB Definition Editor visualizes the contents of an EPM definition file, using a view which divides into the following main areas.

- **Menu Bar:**
  This area contains the menus for the EPM Editor.
- **MIB Tree:**
  This area is used to display the MIB tree. Within the tree OIDs can be selected. If a single OID gets selected, the description of this OID, as it is contained in the MIB definition, will be displayed in the Info Area. The MIB tree uses the following icons:
  - for menus
  - for scalar values or trap variables
  - for tables

- for table rows
- for table columns (attributes)
-  for events (SNMP traps)

- **Info Area:**
  This area is used to display information about the OID currently selected in the MIB tree.
- **Input Area:**
  This area contains the page holders in which the main aspects (discovery, browser, events and status) of the EPM definition file can be edited. The resources page holders contain support functions (see *Section 3.5, "Resources"*).

3.4 File Menu

The Menu Bar contains the menu **File**. This menu offers the functionality to save EPM definition files.

- **Save at...**
  This will open up a dialog, in which a directory can be selected. The contents of the editor will then be saved in this directory using the file name '`<MIB name>.jar`'.
- **Save on server ...**
  [This menu item is only selectable, when the Enterprise MIB Definition Editor has been started within a OpenScape FM Client.]
  The loaded Enterprise MIB Definition will be transferred to and loaded on the OpenScape FM Server. The EPM Definition can then be activated on the OpenScape FM Server via the EPM Plugin.
- **Exit**
  The user will be asked, whether she likes to save the data from the editor. After this the program will be exited.

3.5 Resources

The page holders **Browser Resources** and **Resources** support the other page holders by offering the possibility to localize texts. The localization is done by linking 'keys' to values. The keys will be automatically listed, when they are referenced in the other page holders. The column **key** contains these keys. The other columns (*****, **de**, ...) can be used to enter the localized texts. The fields in these columns are editable by clicking into them with the cursor.

By using the pop up menu of the table header further languages can be added. By selecting or deselecting check boxes within the pop up menu further columns can be added or removed for the localization.

First the OpenScape FM Server searches a key (e.g. 'MyMenuEntry') in the column corresponding to the country setting of the OpenScape FM Server (e.g. 'en'). If this column contains an entry for the key (e.g. 'en': 'MyMenuEntry') then this entry will be used. Else the column '*' will be searched for the key (e.g. '*': 'MyMenuEntry'). If this column contains an entry for the key, this entry will be used. If no entry can be found for the key, the key itself will be used as the value.

3.6 Page holder 'Discovery'

The page holder **Discovery** can be used, to define the criteria, how the EPM plugin recognizes the MIB on an IP node and how the corresponding IP node will be handled after the recognition.

IP nodes are the father objects of all objects discovered in a network. An IP node will be discovered through the Enterprise MIB plugin on the OpenScape FM Server, when MIBs activated in the EPM plugin exist on the IP node, and when the search criteria (see *Section 3.7.1, "Value Browsers"*) is met. An IP node can support a variety of MIBs.

The representation of an IP node and the network to which an IP node is associated, can be influenced by the MIBs supported by the IP node.

The recognition, the representation and the association to a net can be configured on the page holder **Discovery**. The configuration will be explained in the following chapters.

3.6.1 Discovery Type

In the Enterprise MIB Definition Editor the discovery type of the currently handled MIB can be set. The discovery type defines which criteria has to be met to recognize a MIB for an IP node. The area **Discovery Type** contains a pull down menu which contains the following values: 'sysObjectId', 'Subtree' and 'Value'.

Located besides this pull down menu is the field **OID** to which a MIB OID from the MIB tree can be transferred. To transfer a MIB OID from the MIB tree, a MIB OID has to be selected in the MIB tree. Pressing the button besides the field will then transfer the selected MIB OID to the field. This MIB OID will then be interpreted depending on the chosen discovery type. The three different discovery types are handled as followed:

- **sysObjectID:**
  During an IP discovery of the OpenScape FM Server the MIB-II variable 'sysObjectID' will be polled. If the returned value is identical to the one defined in the field OID, the OpenScape FM Server will recognize the MIB for the IP node.
- **Subtree:**

Figure 2 Discovery type 'Subtree'

  During an IP discovery the IP nodes will be checked for the configured OID. When an SNMP GetNext command started with the configured OID returns a result that begins with the configured OID, the OpenScape FM Server recognizes the MIB for the IP node. This is the case, when the configured OID exists as a scalar or when a subtree of the configured OID contains a value.

- **Value:**
  During an IP discovery the IP nodes will be checked for the configured OID and value. When an SNMP Get command started for the configured OID returns a value and when this value is identical to the value configured in the field Value, the OpenScape FM Server will recognize the MIB for the IP node.

Figure 3 Discovery Type 'Value'

3.6.2 Representation of the IP Nodes

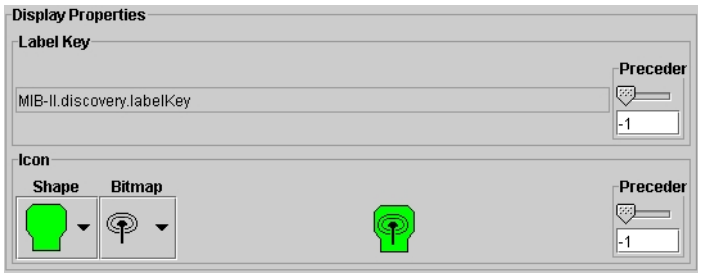


Figure 4 Display Properties of the IP Nodes

The representation of an IP node can be dependent on it's supported MIBs. The representation of an IP node consists of

- a label and
- an icon.

The label can be localized on the page holder **Resources**. The field **Label Key** is used to display a key. This key is used to find or define the localization within the page holder **Resources**. The key is generated by the editor and cannot be modified.
In the example the page holder **Resources** is used to set the default translation and the english translation for the key 'MIB-II.discovery.labelKey'.

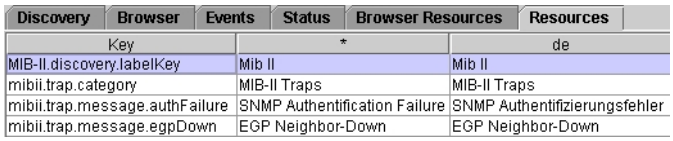


Figure 5 Setting within the place holder 'Resources'

An IP node can support a variety of MIBs and every MIB has the opportunity to influence the representation of the IP node. The slider **Preceder** is used to define the influence of the loaded MIB for the representation of the IP node. The larger the selected value, the larger is the influence. The setting '-1' stands for no influence at all, which means that the IP node will not be changed in its representation by the loaded MIB.

If more than one supported MIB is discovered by the OpenScape FM Server for an IP node, the representation defined by the MIB with the highest defined **Preceder** will be used.

The icon of an IP node can be set in the area **Icon** by selecting entries from the pull down menus **Shape** and **Bitmap**. The influence of the selected appearance can also be chosen by the respective slider **Preceder**.

3.6.3 Associated Networks for IP Nodes

If an IP node is discovered by the OpenScape FM Server, it can be automatically added to a network. The association to a network can be influenced by the supported MIBs.

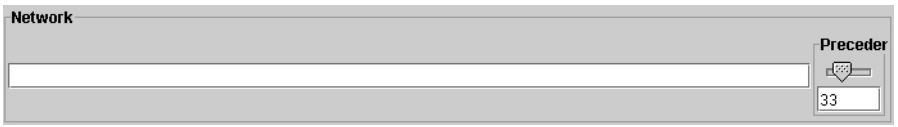


Figure 6 Field for the entry of the network

The field **Network** offers the opportunity to enter the name of the network to which the IP node should be added. If the network does not exist at the time of the discovery, it will be generated. As previously described, the influence of the field **Network** can also be set with the respective slider **Preceder**.

3.7 MIB Browser

The EPM plugin offers the opportunity to define browsers to display the data from a MIB instance. Within the OpenScape FM these browsers will be offered through the pop up menu of the MIB object.

Within the Enterprise MIB Definition Editor these pop up menus can be defined and the respective browsers can be configured. *Figure 7* shows a table browser within the OpenScape FM Client.

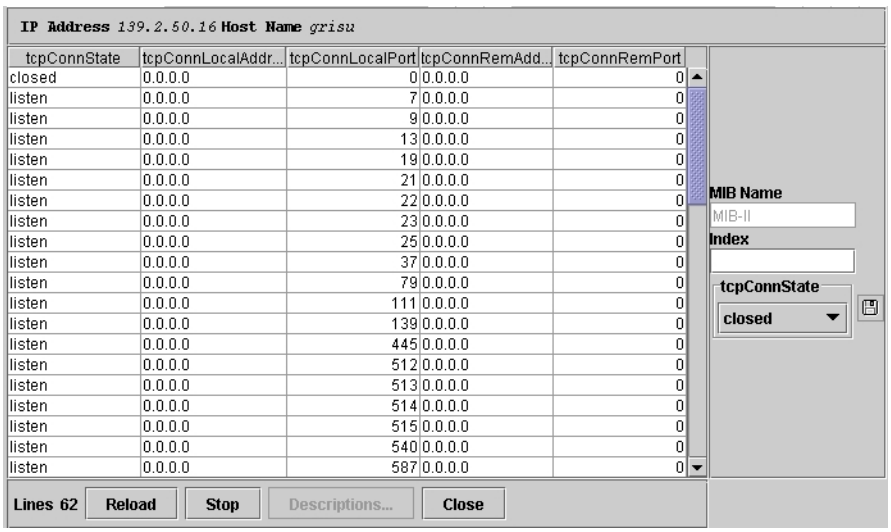


Figure 7 Table browser within the OpenScape FM Client

The page holder **Browser** can be used to configure the SNMP browsers for the EPM plugin within the OpenScape FM Server. The page holder **Browser** consists of two parts. The left side displays a menu tree. The right side displays the configuration of the browser within the OpenScape FM Client, which is connected to the selected menu entry.

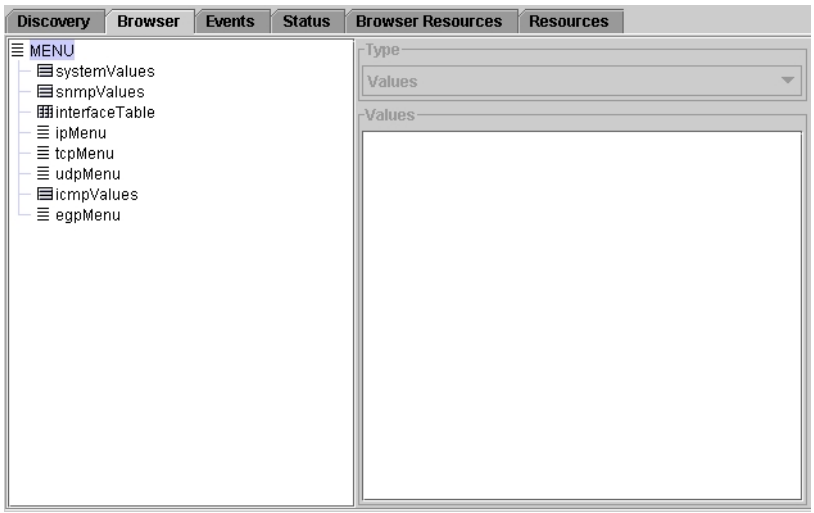


Figure 8 Page holder 'Browser'

The menu tree consists of menu items representing the main menu ('MENU'), sub menus and actions. The following icons are used to distinguish the different types of entries:

- for menus
- for scalar values
- for tables

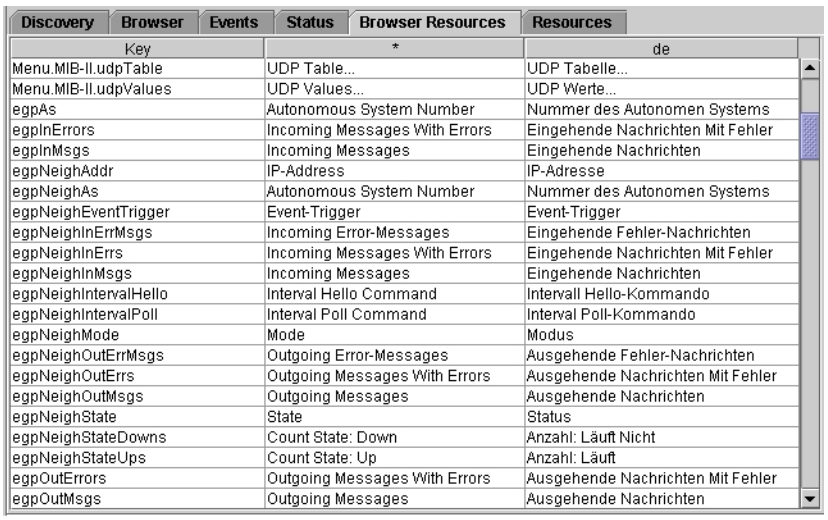
The main menu and the sub menus offer a pop up menu which provides the following entries:

- **New Menu:**
  A new sub menu will be created below the selected menu.
- **New Entry:**
  A new entry will be created below the selected menu.

For all menu entries besides for the main menu the following entries are available:

- **Rename...:**
  The input of a new name within the tree will be allowed for the menu entry.
- **Remove:**
  The menu entry and all entries in it's sub tree will be removed.
- **Up:**
  Within the tree the menu entry will be moved up one row.
- **Down:**
  Within the tree the menu entry will be moved down one row.

For every menu entry a translation for the language chosen in the OpenScape FM Client has to be given within the page holder **Browser Resources**. The key 'Menu.<MIB name>.<menu entry>' will be used for the menu entry '<menu entry>'. By right clicking on the header of the table **Browser Resources** a pull down menu will open, which can be used to add further languages to the resource table. The column '*****' contains the default settings, which will be used, when the column for the used language contains no value, or when no column exists for the used language.



Key	*	de
Menu.MIB-II.udpTable	UDP Table...	UDP Tabelle...
Menu.MIB-II.udpValues	UDP Values...	UDP Werte...
egpAs	Autonomous System Number	Nummer des Autonomen Systems
egpInErrors	Incoming Messages With Errors	Eingehende Nachrichten Mit Fehler
egpInMsgs	Incoming Messages	Eingehende Nachrichten
egpNeighAddr	IP-Address	IP-Adresse
egpNeighAs	Autonomous System Number	Nummer des Autonomen Systems
egpNeighEventTrigger	Event-Trigger	Event-Trigger
egpNeighInErrMsgs	Incoming Error-Messages	Eingehende Fehler-Nachrichten
egpNeighInErrs	Incoming Messages With Errors	Eingehende Nachrichten Mit Fehler
egpNeighInMsgs	Incoming Messages	Eingehende Nachrichten
egpNeighIntervalHello	Interval Hello Command	Intervall Hello-Kommando
egpNeighIntervalPoll	Interval Poll Command	Interval Poll-Kommando
egpNeighMode	Mode	Modus
egpNeighOutErrMsgs	Outgoing Error-Messages	Ausgehende Fehler-Nachrichten
egpNeighOutErrs	Outgoing Messages With Errors	Ausgehende Nachrichten Mit Fehler
egpNeighOutMsgs	Outgoing Messages	Ausgehende Nachrichten
egpNeighState	State	Status
egpNeighStateDowns	Count State: Down	Anzahl: Läuft Nicht
egpNeighStateUps	Count State: Up	Anzahl: Läuft
egpOutErrors	Outgoing Messages With Errors	Ausgehende Nachrichten Mit Fehler
egpOutMsgs	Outgoing Messages	Ausgehende Nachrichten

Figure 9 Localization of menu entries

When an entry gets selected within the menu tree, the browser that should be opened for the entry can be configured. Two types of browsers are available.

- Value Browsers and
- Table Browsers.

3.7.1 Value Browsers

Value Browsers are used to display configured (scalar) values within the OpenScape FM Client. Each of these values are displayed within a separate row. The Value Browsers will be configured within the Enterprise MIB Definition Editor by setting the pull down menu **Type** to **Values** for an action selected from the menu tree.

After this the values which should be displayed by the browser can be selected from the MIB tree. When the values which should be displayed are selected in the MIB tree, they can be added by selecting the entry **Add from Tree** within the pop up menu of the area **Values**. The order in which the values should be displayed can be changed by using the menu entries **Up** and **Down** of the same pop up menu or by using drag'n drop on the values which should be moved. The entry **Remove** within the pop up menu can be used to remove values which should no longer be displayed.

The page holder **Browser Resources** can be used to provide short descriptions which will be displayed in the browser column **Short Description**.

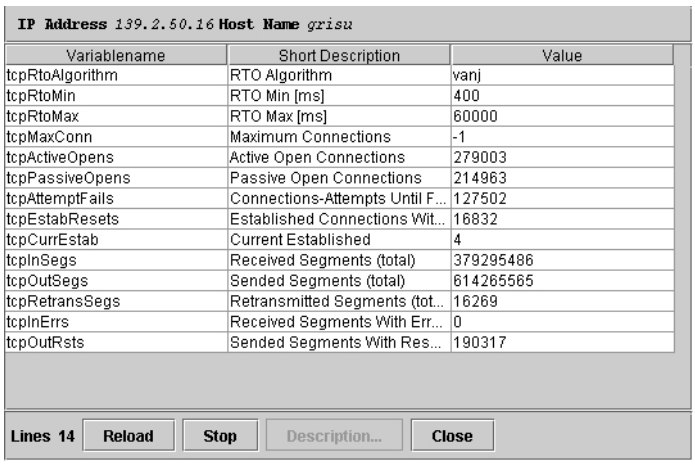


Figure 10 Value Browser within the OpenScape FM Client

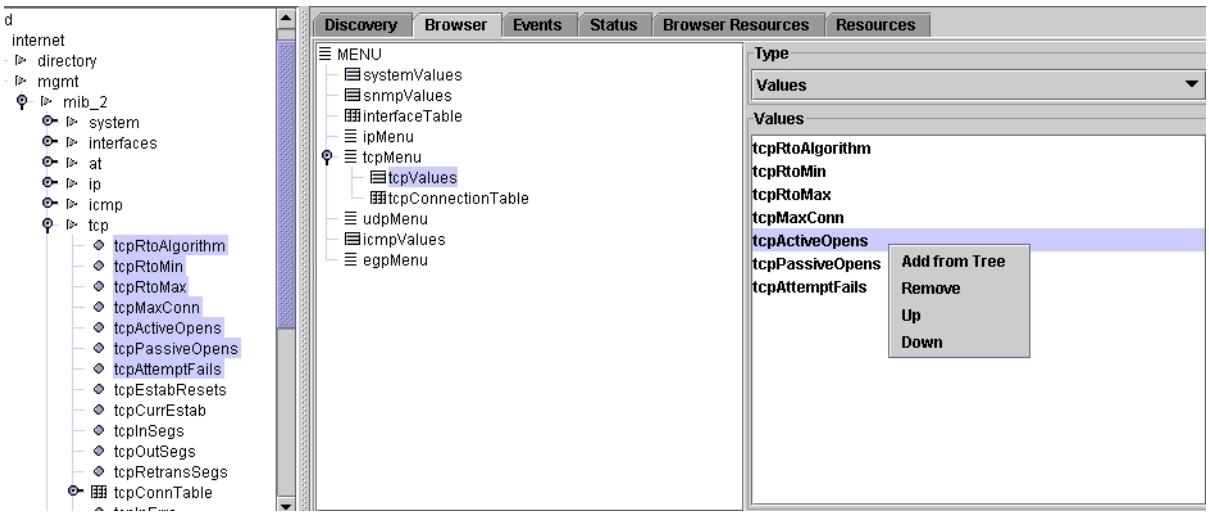


Figure 11 Value Browser: Transferring values from the MIB tree

3.7.2 Table Browser

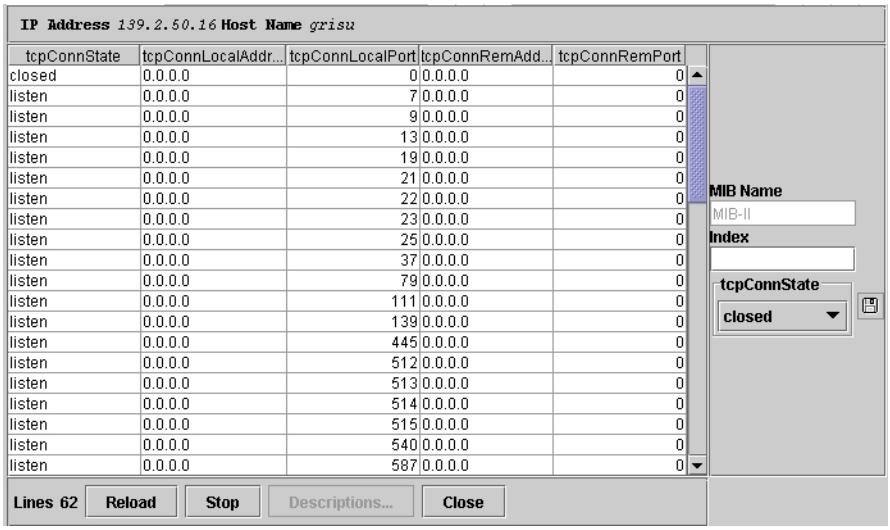


Figure 12 Table Browser within the OpenScape FM Client

Table Browsers are used to display configured columns from a table within the OpenScape FM Client. The Table Browsers will be configured within the Enterprise MIB Definition Editor by setting the pull down menu **Type** to **Table** for an action selected from the menu tree. After this the MIB table which should be displayed has to be selected within the MIB tree. When this is done, the table can be transferred to the field **Tablename** by pressing the button  .

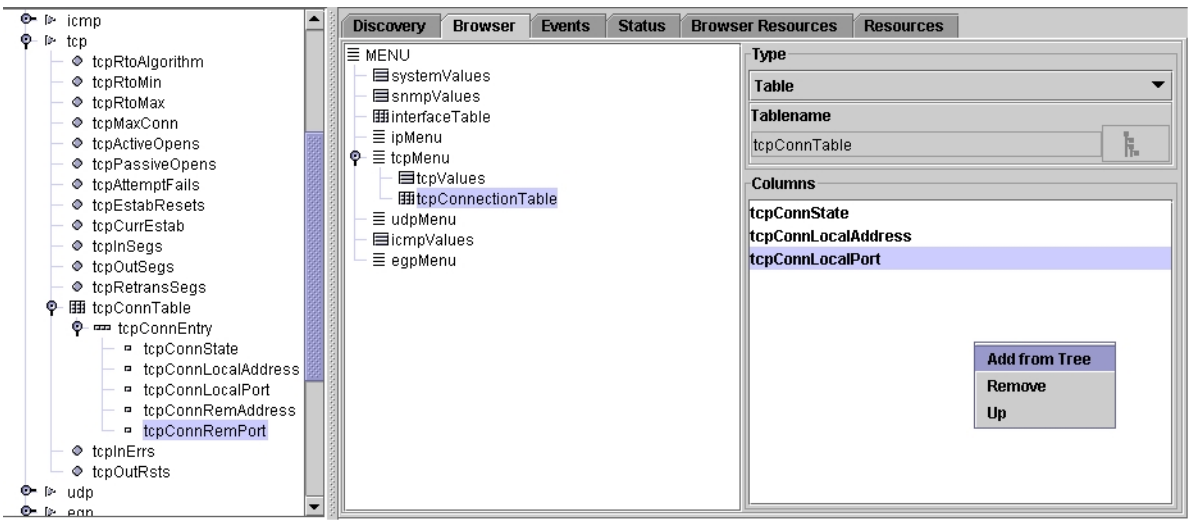


Figure 13 Table Browser: Transferring values from the MIB tree

When the table is defined, the table columns which should be displayed, must be selected within the MIB tree. They can then be added by selecting the menu item **Add from Tree** within the area **Columns**. This menu item is active when only columns from the table named in the field **Tablename** are selected. When a column no longer should be displayed, it can be removed by selecting the menu entry **Remove** from the same pop up menu.

The page holder **Browser Resources** can be used to attach short descriptions to the variables. These short descriptions will be shown as tool tips when the respective column is displayed.

3.8 Events (SNMP Traps)

The EPM plugin registers itself as an SNMP trap recipient for all MIB instances that are recognized. For every received trap an entry will be created within the Event Browser of the OpenScape FM. Using the page holder **Events,** the contents of these entries can be configured within the Enterprise MIB Definition Editor.

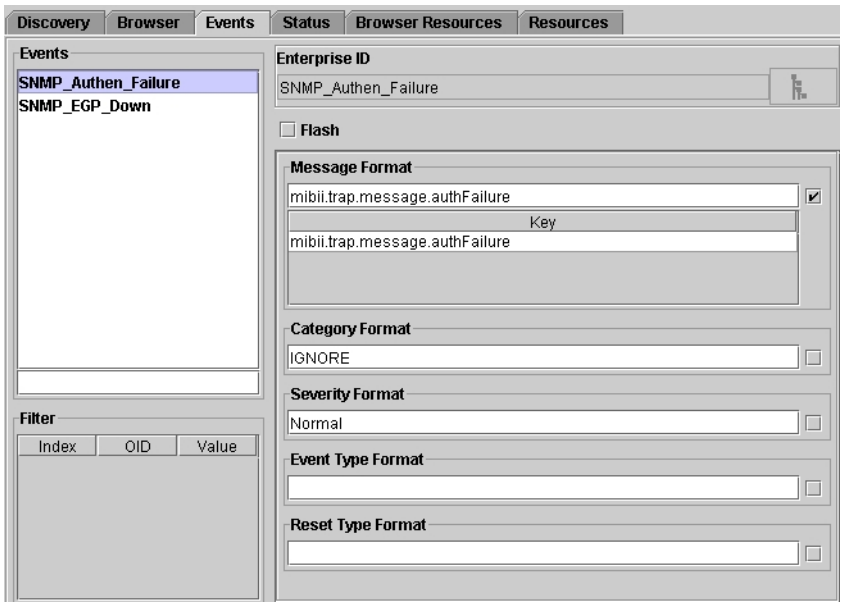


Figure 14 Page holder 'Events'

A new event for the Event Browser can be created by entering a new name for the event in the field below the list **Events** and by confirming this name by pressing the enter key. The entered name will then be added to the list. If this name gets selected in the list, the event can be configured in the different areas of the page holder.

3.8.1 Enterprise ID

After the definition of the event name, the OID of the SNMP trap, that should initiate the event within the EPM plugin, has to be defined. This can be done by selecting an SNMP trap within the MIB tree and by pressing the button. The selected value will then be transferred to the field **Enterprise ID**.

3.8.2 SNMP Trap Filter

Different events can be triggered through a single SNMP trap. To allow the association of a received SNMP trap to an event, the SNMP traps has to be filtered by the OpenScape FM. The respective filters consist of a set of trap variables and their requested values. A received SNMP trap matches for an event, when **all** configured variable values match. These values are configured in the area **Filter** in which each variable should not be entered more than once.

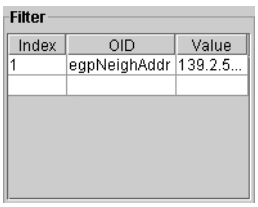


Figure 15 Entry area for filter values

To add a condition (variable/value) to a filter, first a new entry row has to be created within the **Filter** area. This is done by selection the menu entry **New** from the pop up menu of the **Filter** area.

Into the column **Index** of the new row the index of the SNMP trap variable has to be entered. Into the column **Value** the expected trap variable value has to be entered. If the SNMP trap variable is defined within the MIB, its name will be automatically added to the column **OID**.

3.8.3 Flashing

The checkbox **Flash** on the page holder **Events** can be used to define, whether an affected object should flash in the OpenScape FM, when a respective unacknowledged event exists within the OpenScape FM Event Browser.

3.8.4 Event Format Definition

SNMP traps contain information like type, time and trap variables. Events in the Event Browser have the following attributes, whose values can be derived from this information:

- Description,
- Category,
- Status and
- Acknowledged.

The configuration of the EPM plugin defines the mapping of the SNMP trap information to the values of the event attributes. For this attribute definitions are used. With the help of these definitions, messages are created and displayed in the Event Browser, when an respective SNMP trap is received.

Attribute definitions may contain text macros, which will be substituted with the respective SNMP trap information when messages are generated.

All text macros start with the dollar symbol ('$'). The following list provides an overview of the implemented macros. The macro will be substituted with:

- $1 - $99 the value of the respective trap variable
- $# the number of the variables
- $* all trap variables
- $@ the time in seconds when the event was generated
- $T the system time of the SNMP agent system, when the trap was sent
- $x the time as a formatted character string, when the event was generated
- $R/$r the source of the event
- $c the protocol category of the event
- $s the protocol severity of the event
- $e the Enterprise OID of the event
- $D[1-99] interprets the respective trap variable as a time stamp and generates a formatted character string as the output (for HiPath 3000)
- $G[1-99] interprets the respective trap variable as a time stamp in GMT and generates a formatted character string as the output (for HiPath 3000)

The usage of macros will be explained by examples in the following chapters.

3.8.4.1 Message Formats

The Event Browser of the OpenScape FM displays a short description (attribute **Description**) of the event. This description will be localized by the OpenScape FM Server. This will be done by executing the following steps:

1. The SNMP trap and the attribute definition (which may contain macros) will be used to create a key.
2. For this key a match will be searched within the page holder **Resources** considering the country settings. If a match is found, it will be used. If none is found, the key will be used as the value.
3. The value will be searched for macros, which will be replaced by the respective values from the SNMP trap.
4. The result will be displayed in the Event Browser.

Example:

A device of the type 'MyDevice' generates an SNMP trap when an intern module changes it's state. The first trap variable contains the new state ('Up' or 'Down'). The second variable contains the number of the module which was affected. Two different messages shall be generated, both for german and english OpenScape FM Clients:

1. A message for the state 'Up'
2. A message for the state 'Down' which should also include the number of the module

First the attribute definition has to be defined. The generated key has to be dependant on the first variable containing the state. The message format (the attribute definition for the event attribute description) should be 'MyDevice.message-$1'. This format uses the macro '$1', which adds the value of the first trap variable to the generated key.

The attribute definition has to be entered into the area **Message Format**. After this the table, which will be activated by the check box on the right side, contains the column '$1' (for every variable, which is referenced in the key format, a column gets visible in the table). When a new row is added to the table by using the pop up menu entry **New**, the possible values for the first trap variable can be entered ('Up' and 'Down').

Important Note:
The table only displays column '$1' when '$1' is part of the format.

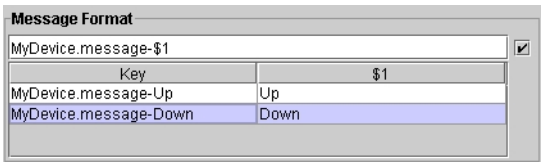


Figure 16 Entry of the message format and variable values

This generates the keys 'MyDevice.message-Up' and 'MyDevice.message-Down' on the page holder **Resources**. These keys will be displayed in the column **Key**.

In this example these keys shall be localized for english and german servers:

german:

- MyDevice.message-Up: Gerät $R ist betriebsbereit.
- MyDevice.message-Down: Gerät $R: Fehler in Modul $2.

english:

- MyDevice.message-Up: Device $R is up.
- MyDevice.message-Down: Device $R: Error in module $2.

In the page holder **Resources** the localizations have to be entered. By right clicking the header of the table **Resources** a pop up menu will open, which can be used to add columns for further languages to the table **Resources**. The text can then be entered by clicking in the respective fields of the table.

The text macros '"$2' and '$R' in the example will be replaced with the number of the module and with the IP address, when a message will be created:

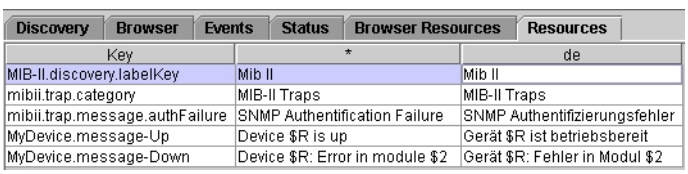


Figure 17 Page holder 'Resources' with entries for the example

A down message for an english OpenScape FM Server for module 23 would look like this:

- Device 139.2.50.150: Error in module 23.

3.8.5 Category Format

Like the message format the category format (attribute definition for the event attribute ’Category’) can be defined within the area **Category Format**. The generated text will be displayed in the column **Category** of the OpenScape FM Event Browser. Often the category shall not be localized and can directly be generated from the OID of the SNMP trap. In this case it is sufficient to enter the category into the **Category Format** without entering a value for the key in the page holder **Resources**. The attribute format will then be used as the entry in the column **Category** within the Event Browser.

Example:

The attribute format for the category will be set to ’MyDevice’.

Figure 18 Entry area ’Category Format’

3.8.6 Severity Format

The OpenScape FM Server assigns a severity to each event. The severity describes the importance of the event. The OpenScape FM Server uses the severities: Unset, Unknown, Normal, Warning, Minor, Major, Critical, Unmanaged, Testing, Disabled, Restricted.

These severities can be calculated from the SNMP trap variables. Like for the message format, the OpenScape FM Server uses the SNMP trap and the attribute definition to generate a key.

The attribute definition will be entered in the area **Severity Format**. For the generated keys, like for the message format, a severity can be defined on the page holder **Resources**. Since severities cannot be localized, only the value in the column ’*****’ needs to be entered.

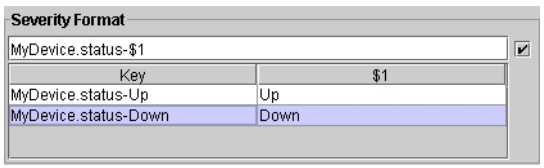
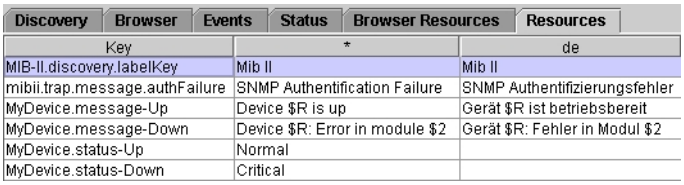


Figure 19 Keys for the severities for the example



Discovery | Browser | Events | Status | Browser Resources | Resources

Key	*	de
MIB-II.discovery.labelKey	Mib II	Mib II
mibii.trap.message.authFailure	SNMP Authentification Failure	SNMP Authentifizierungsfehler
MyDevice.message-Up	Device $R is up	Gerät $R ist betriebsbereit
MyDevice.message-Down	Device $R: Error in module $2	Gerät $R: Fehler in Modul $2
MyDevice.status-Up	Normal	
MyDevice.status-Down	Critical	

Figure 20 Page holder ’Resources” with the entries for the example

Example:

For the SNMP trap from the example in *Section 3.8.4.1, “Message Formats”* severities shall be defined. The value ’Up’ shall be mapped to ’Normal’ and the value ’Down’ shall be mapped to ’Critical’.

The severity format will be set to ’MyDevice.status-$1’. For the first trap variable ’Up’ and ’Down’ will be entered as values into the table, which results in the keys ’MyDevice.status-Up’ and ’MyDevice.status-Down’ (column **Key**). On the page holder **Resources** the column ’*****’ can be used to enter the respective values to the keys.

- MyDevice.status-Up: Normal
- MyDevice.status-Down: Critical

3.8.7 Event Type Format and Reset Type Format

Often events are connected to each other in a way that an event reverses the effects of other (previous) events. An operator then no longer needs to consider the previous events.

The EPM plugin provides a mechanism to allow events to acknowledge previous events within the Event Browser. For this the areas **Event Type Format** and **Reset Type Format** are available. For every event an Event Type Format is defined. If an event arrives which has a Reset Type Format which is identical to the Event Type Format of previous events, all respective previous events will be acknowledged in the Event Browser. In the example of *Section 3.8.4.1, “Message Formats”* SNMP traps with the value ’Up’ can be defined to acknowledge events with the value ’Down’.

The entry of attribute definitions can be done on the page holder **Events** in the areas **Event Type Format** and **Reset Type Format**, like it is done for the message formats. The event types and reset types do not need to be localized, since they are not displayed in the program.

Example:

The SNMP traps of the example in *Section 3.8.4.1, “Message Formats”* are set to the event type ’MyDevice.type-$1’ and the reset type ’MyDevice.reset-$1’. The macro ’$1’ is used to distinguish between ’Up’ and ’Down’ traps.

In the page holder **Resources** the keys will be filled with device independent values (’$R’).

- MyDevice.type-Up: MyDevice.type-Up.$R
- MyDevice.type-Down: MyDevice.type-Down.$R
- MyDevice.reset-Up: MyDevice.type-Down.$R
- MyDevice.reset-Down: MyDevice.type-Up.$R

The macro ’$R’ guarantees that only events that are sent by the same device will be used to acknowledge each other. Only ’Up’ events of the same device will acknowledge ’Down’ events and vice versa.

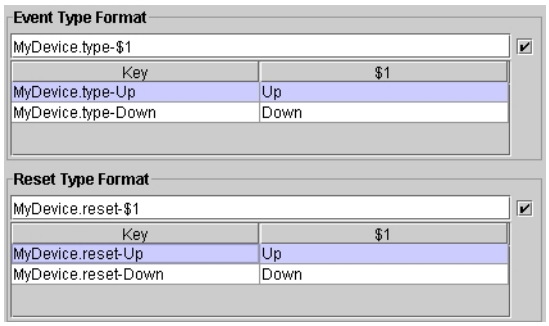


Figure 21 Event Type Format and Reset Type Format

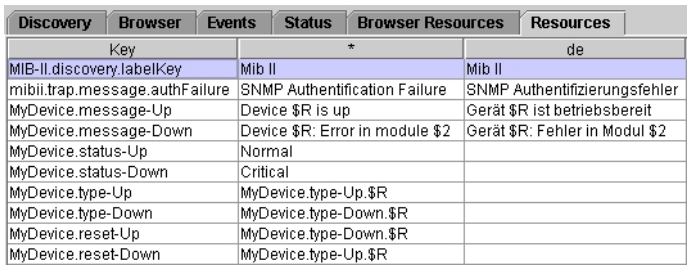


Figure 22 Page holder ’Resources’ with entries for the example

Entry of Multiple Keys:

In some cases one event should be reset by multiple events.

In this case a separator can be assigned for the event type format. This separator will be used to divide the event type into a number of substrings. Each of these substrings will be treated as a different key for the event, therefor adding more than one key to a single event.

If an event should reset more than one event and these events use different keys, then a separator can be assigned to the reset type format.

The separator will be used to divide the reset type into substrings. Each of these substrings will be treated separately, resetting the matching events.

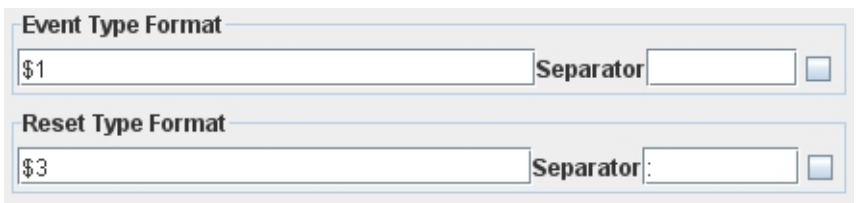


Figure 23 Formats with Separator

Example:

Using the definition in *Figure 23* the variable $1 will be taken as the key for incoming events.

Since the separator ':' is defined for the reset type format, the content of variable $3 will be devided into parts. Each of these parts will be handled as a different reset key. E. g. if $3 contains the value '128:130:132' then the events with the keys '128', '130' and '132' will be resetted.

If '&' would have been assigned as a separator for the event type format, and if variable $1would consist of the string '100&200&300', then the event keys '100', '200' and '300' would be added to the event. The event will be resettet, when a reset key contains one of these three keys.

3.9 Status

In OpenScape FM objects have a status. For the object that represents the SNMP MIB also a status is calculated. This status is defined by the status of the **Status Objects** that are located below the SNMP MIB object. The status of the SNMP MIB object is identical to the status with the highest severity (see *EPM plugin User Guide).*

The status of a Status Object will be defined by the current setting of the MIB variables.

To create a Status Object an entry has to be created in the list **Status Objects** on the page holder **Status**. For this the name of the new Status Object has to be entered into the field below the list **Status Objects** and be confirmed by the enter key.

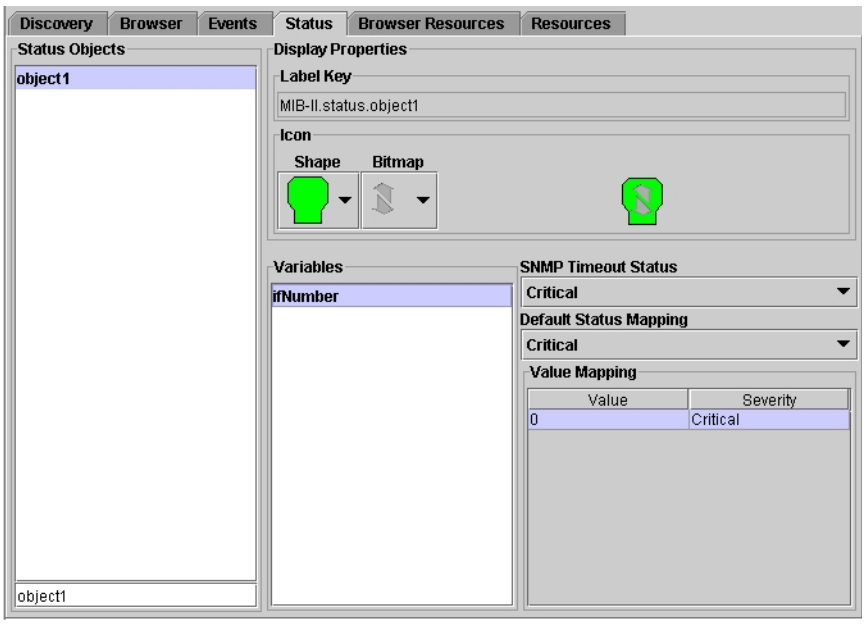


Figure 24 Page holder 'Status'

When a Status Object gets selected in the list, the display characteristics of the selected Status Object can be defined. This is done as described in *Section 3.6.2, "Representation of the IP Nodes"*.

The status of a Status Object will be defined by the setting of the MIB variables. For this a MIB variable has to be selected within the MIB tree. The entry **Add from Tree** from the pop up menu of the area **Variables** can then be used to transfer it into the field **Variables**.

Note:
The entry **Add from Tree** is only active when a scalar value is selected in the tree.

The value in the selector **SNMP Timeout Status** will be used, when the MIB instance does not answer to requests. The value in the selector **Default Status Mapping** will be used, when no value from the mapping table fits to the current values of the variables. For the variables selected in the list **Variables** Value/Severity combinations can be entered in the **Value Mapping** area.

Status

A Installation of the Enterprise MIB Definition Editor

The Enterprise MIB Definition Editor is integrated into the OpenScape FM and will be installed as a part of the OpenScape FM Server installation. It will be activated by initializing the EPM plugin.

Index