



NuDesign Visual MIBrowser Pro Advanced Features Guide

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



| | |
|---|-----------|
| WHAT IS VISUAL MIBROWSER PRO™ | 5 |
| DEVICES VIEW | 6 |
| <u>A Hierarchical Tree Layout</u> | 6 |
| <u>Color Coded Alarm Conditions</u> | 7 |
| <u>Icons Depict Device Types</u> | |
| | 11 |
| <u>Adding and Removing Devices</u> | 12 |
| <u>Setting SNMP Device Type</u> | |
| | 14 |
| <u>Alarm Sources</u> | |
| | 15 |
| <u>Clearing An Alarm Condition</u> | |
| | 16 |
| <u>Finding Alarm Causes</u> | 17 |
| <u>Actions and Applets</u> | 20 |
| <u>Connecting to Web Managed Devices</u> | 21 |
| USER CREATED PLUG IN APPLETS | 22 |
| <u>Sample C++ Plugin Project “SampleDllPlugin”</u> | 22 |
| <u>Installing and Using “SampleDllPlugin” as a “Favorite”</u> | 22 |
| <u>Installing and Using “SampleDllPlugin” as a device “Action”</u> | 23 |
| <u>Using “SampleDllPlugin”</u> | |
| | 23 |
| <u>Building the sample</u> | 26 |
| EXPRESSION BASED CHARTING | 27 |
| <u>Expressions</u> | 28 |
| <u>Examples:</u> | 31 |
| <u>Creating an Utilization Chart on the first network interface</u> | 31 |
| <u>Creating a Packet Rate Chart on the first network interface</u> | 35 |
| <u>Creating Other Charts</u> | 35 |
| CONFIGURING ALARMS | 37 |
| <u>Alarms Filtering and Routing</u> | 37 |
| <u>SNMP Notifications and V3 Reports</u> | |
| | 38 |
| OTHER FEATURES | 41 |
| SNMP Row Management | 41 |
| <u>Creating a new Row</u> | 41 |

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



| | |
|--|---------------------------|
| Operating on multiple rows at once..... | 41 |
| Information Display Features..... | 43 |
| “Get Table” Pivoted View..... | 43 |
| “Ledger” Mode..... | 44 |

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



DOCUMENT HISTORY

| Date | Rev. | Remarks |
|------------|------|--|
| 2008-02-25 | A000 | Original release |
| 2008-02-28 | A001 | Minor reformatting, editing |
| 2008-04-07 | A002 | Reformat |
| 2009-03-29 | A003 | More reformatting |
| 2010-09-15 | A004 | Rebranding to MIBrowser Pro Add Row Management and Display sections |
| 2010-09-21 | A005 | Rename document |

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



What is Visual MIBrowser Pro™

NuDesign Visual MIBrowser Pro™ is a multi-protocol (SNMPv1/2/3 and HTTP) device management application from NuDesign Technology Inc. that provides enhanced capabilities over and above other products in this market place.

NuDesign Visual MIBrowser Pro™ is a product in the **NuDesign Technologies Visual MIBrowser** product family, which includes

- Visual MIBrowser™
- Visual MIBrowser Professional for HMS™
- Visual SNMP Traffic Monitor™.

In addition to the core functionality provided in the above products, added to this product line's features with this product are:

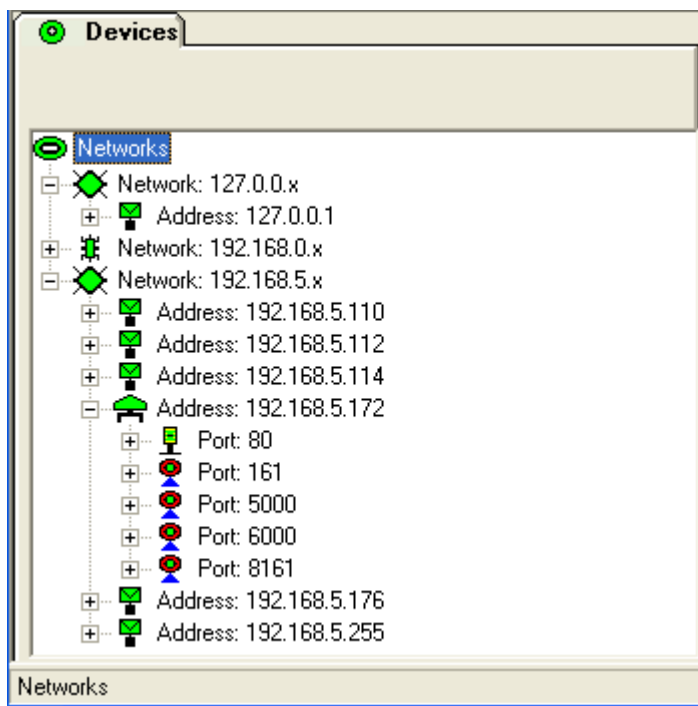
- A hierarchical devices management view, including alarm condition presentation and alarm source identification.
- The facilities to add your own “plug in” applications to extend Visual MIBrowser Pro's capabilities.
- Expression based graphing.
- An interface to web managed devices.
- Reorganized and improved alarm detection and logging.

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

Devices View

A Hierarchical Tree Layout

This view provides a hierarchical depiction of the devices Visual MIBBrowser Pro is configured to manage. Each major branch represents a network.

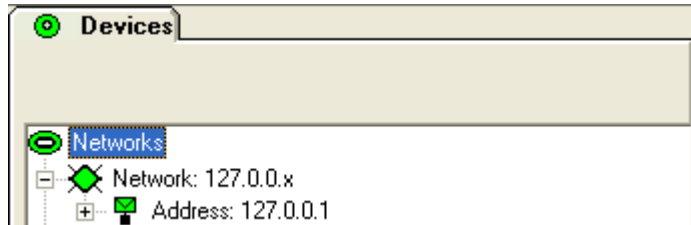


This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

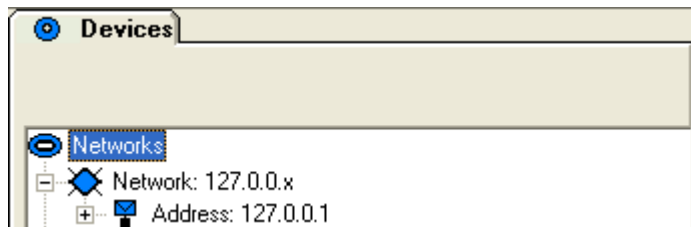
Color Coded Alarm Conditions

The icons and Device tab icon are color coded to provide an indication of the current network alarm condition.

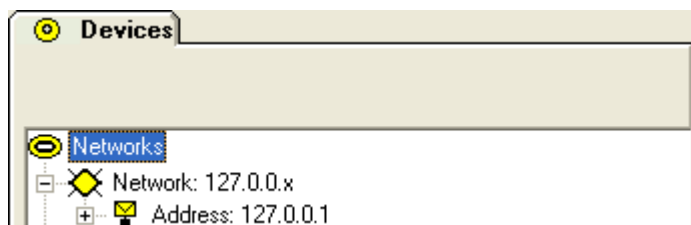
- Green icons indicate no alarms have been detected since start up or the last time the alarm status has been cleared.



- Blue icons indicate an alarm assigned an “*informative*” alarm level, has been detected.

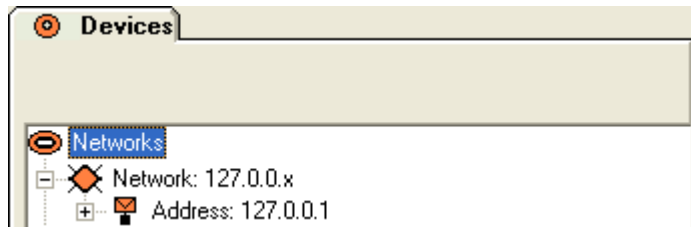


- Yellow icons indicate an alarm assigned a “*warning*” alarm level, has been detected.

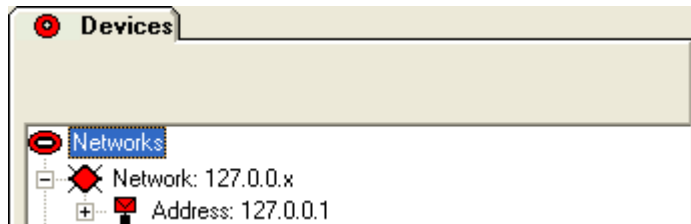


This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

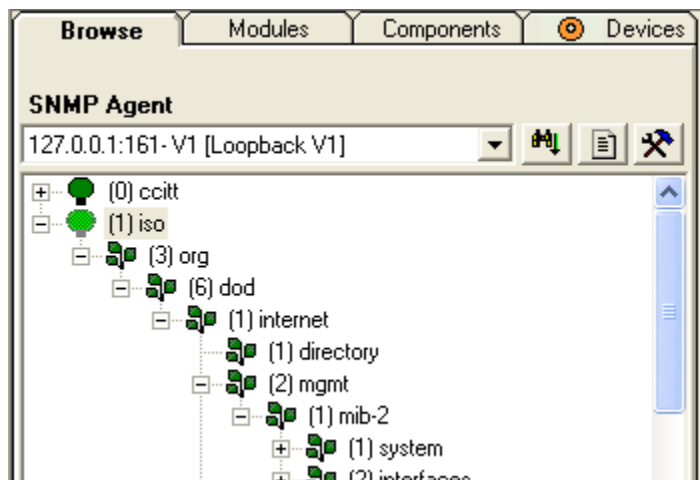
- Orange icons indicate an alarm assigned an “*error*” alarm level, has been detected.



- Red icons indicate an alarm assigned a “*critical*” alarm level, has been detected.



- The device tab depicts the current alarm state so that the alarm status can be detected when working in another tab view.



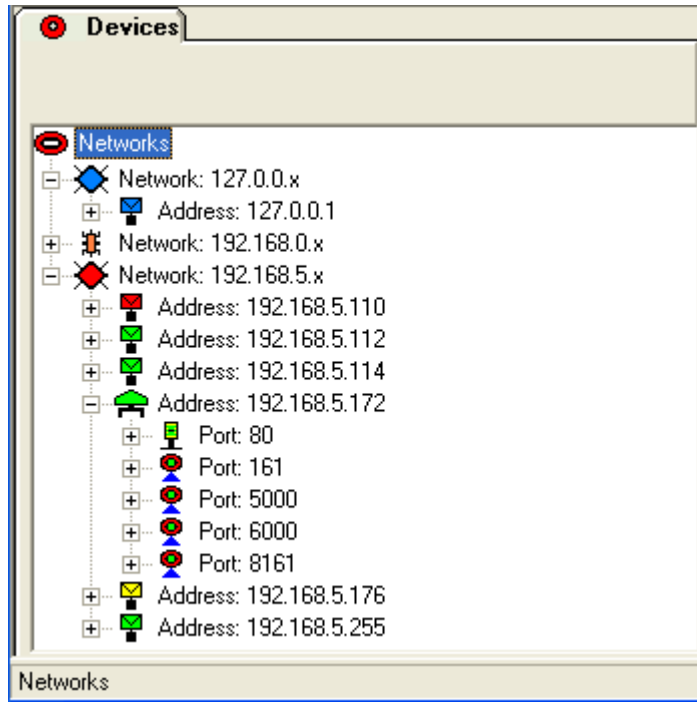
The above shows an *error* condition exists.

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- Alarm condition coloring code is hierarchical. As alarms are detected, higher alarm levels are propagated to the branches above, ultimately to the top or root node.



In the above image, a *critical error* condition exists for address 192.168.5.110. The condition has been propagated to network 192.168.5.x and to top of the tree, “Networks”.

An *informative* alarm condition exists at address 127.0.0.1, which is propagated to network 127.0.0.x, but is not propagated above that point since a higher level condition exists for “Networks”

An *warning* alarm condition exists on network 192.168.0.x, but is not propagated above that point since a higher level condition exists for “Networks”

Icons Depict Device Types


The following icons are used to depict device service types and access method, whether the service is SNMP or web accessible. There are two states for each icon, open and closed. The open state is used when the tree below an icon is open, otherwise the closed state is used.


 - (open, closed) Depicts device is a workstation, access via SNMP.


 - (open, closed) Depicts device is a server, accessed via SNMP.

 - (open, closed) Depicts device is a router, accessed via SNMP.

 - (open, closed) Depicts device is a printer, accessed via SNMP.

 - (open, closed) Depicts an “other” kind of device, accessed via SNMP.

 - (open, closed) Depicts an unknown kind of device, accessed via SNMP.

 - (open, closed) Depicts an “other” kind of device, accessed via HTTP.

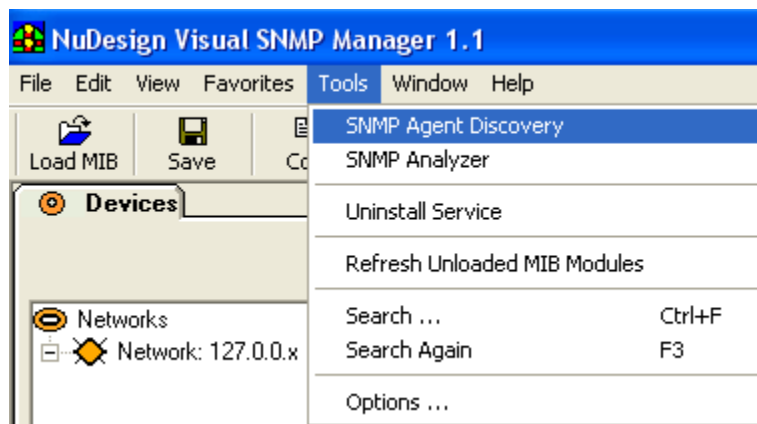
Note: The base of the icons for SNMP accessed devices is a small blue triangle whereas Web accessed device icons have a black line horizontal line.

Adding and Removing Devices

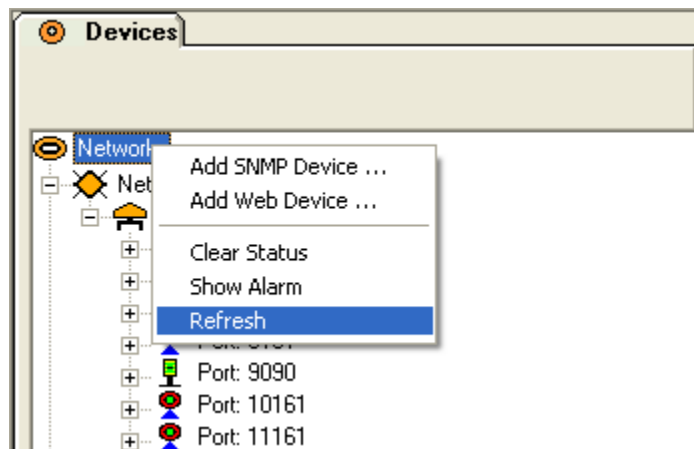
Each device that is depicted in the view is a result of configuration via several different means. Networks are added to the tree automatically as they are determined, based on the existence of configured agents in a network.

Configuration interfaces:

- Automatic discovery of SNMP agents, using the SNMP Agent Discovery Tool.

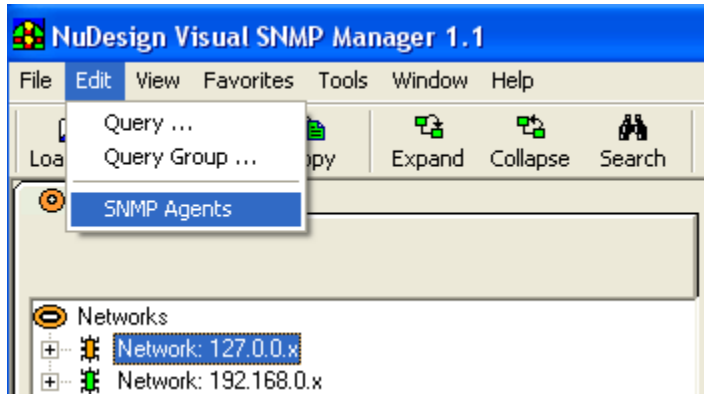


Once any discovered agents are saved, they can be added to the **Devices** tree with the Devices tree “Refresh” menu option. If the device is in a new network, a new network node will be added with the refresh.



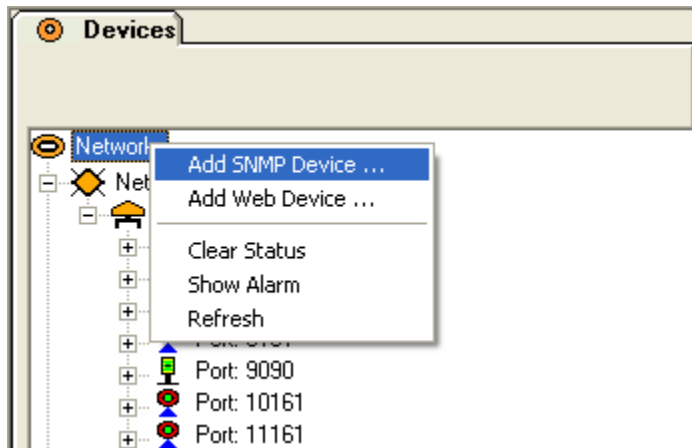
This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- Explicit creation of SNMP agents, using the Edit SNMP Agents interface.

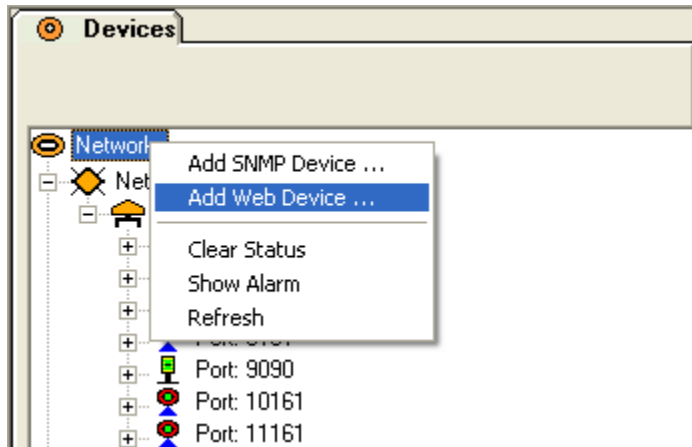


As with the sample above, a **Devices** tree “Refresh” will ensure the new device is displayed in the correct location in the Devices tree.

- Explicit addition of a SNMP Device to the **Devices** tree view from the “Networks” node mouse menu.

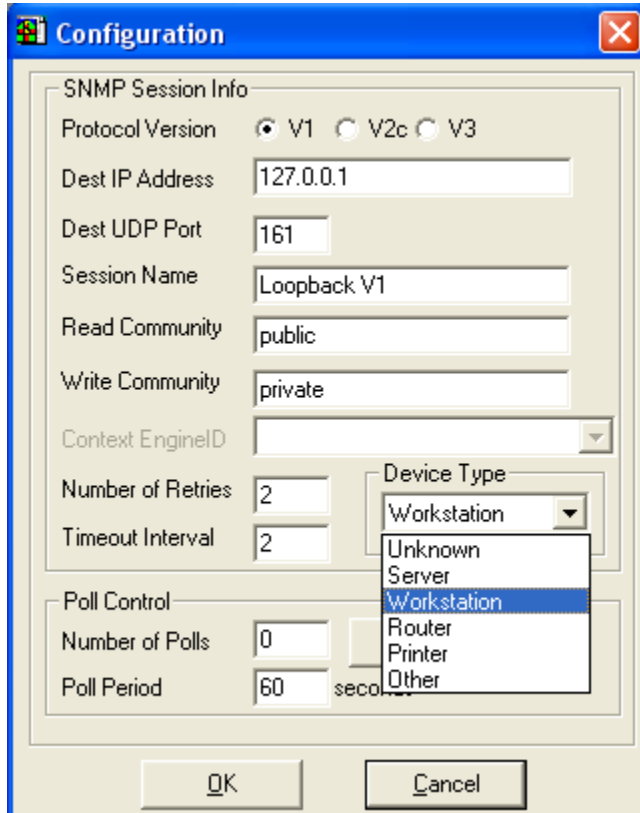


- Explicit addition of a Web Device to the **Devices** tree view from the “Networks” node mouse menu.



Setting SNMP Device Type

Device type is set for a SNMP device from the SNMP session **Configuration** dialog interface.



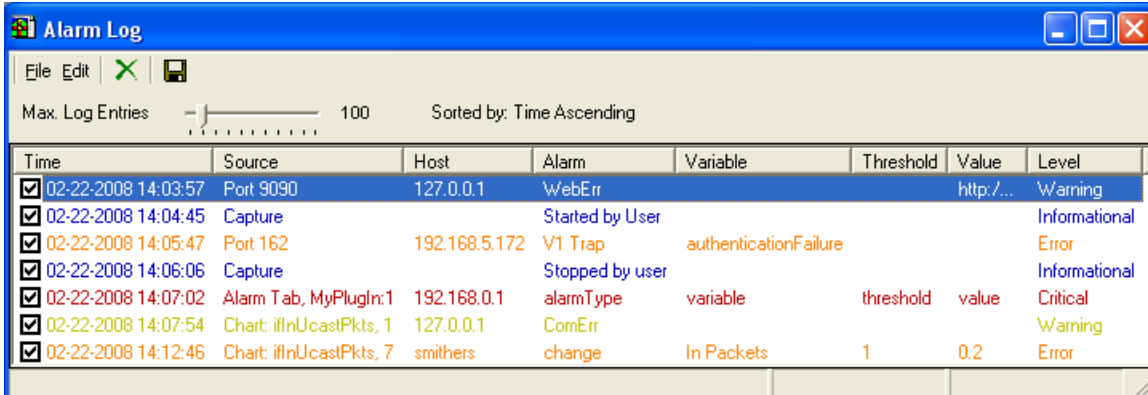
Alarm Sources

Alarms are detected from a number of sources:

- Locally detected via charts that have alarm detection configured. E.g. if an error rate chart violates an alarm detection setting, such as a threshold.
- Locally detected SNMP or Web communication errors.
- SNMP notifications or SNMP V3 Report messages that have been received.
- Plug in applications which explicitly forward alarm notifications to Visual MIBrowser Pro™.
- Start and stop events from the Analyzer, whether programmed or user initiated.

You can determine the source of a particular alarm from the **Alarm Log**.

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



| Time | Source | Host | Alarm | Variable | Threshold | Value | Level |
|---------------------|-------------------------|---------------|-----------------|-----------------------|-----------|------------|---------------|
| 02-22-2008 14:03:57 | Port 9090 | 127.0.0.1 | WebErr | | | http://... | Warning |
| 02-22-2008 14:04:45 | Capture | | Started by User | | | | Informational |
| 02-22-2008 14:05:47 | Port 162 | 192.168.5.172 | V1 Trap | authenticationFailure | | | Error |
| 02-22-2008 14:06:06 | Capture | | Stopped by user | | | | Informational |
| 02-22-2008 14:07:02 | Alarm Tab, MyPlugIn:1 | 192.168.0.1 | alarmType | variable | threshold | value | Critical |
| 02-22-2008 14:07:54 | Chart: ifInUcastPkts, 1 | 127.0.0.1 | ComErr | | | | Warning |
| 02-22-2008 14:12:46 | Chart: ifInUcastPkts, 7 | smithers | change | In Packets | 1 | 0.2 | Error |

Analyzing the “Source”, “Host” and “Alarm” columns make the source of individual alarms evident.

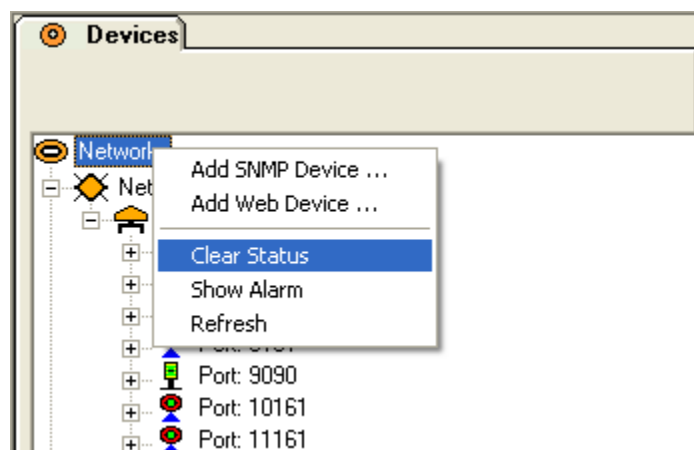
As in the **Devices** tree, log entries are color coded to indicate level (severity).

See the section “[Finding Alarm Causes](#)” on how to associate an alarm condition in the **Devices** tree to an alarm in the **Alarm Log**.

Clearing An Alarm Condition

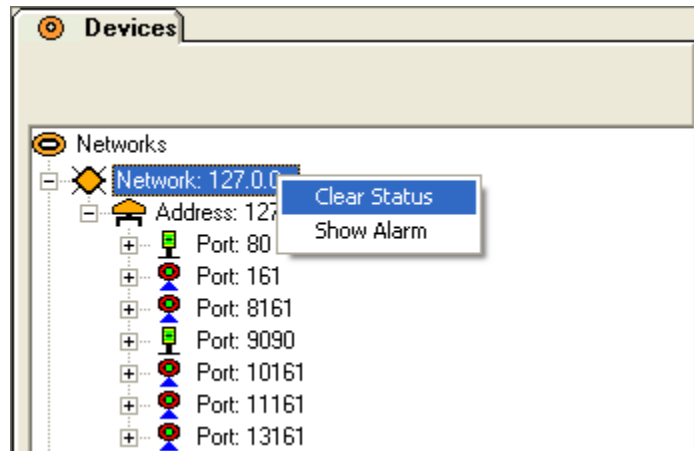
Alarm status is be cleared by selecting the “Clear Status” mouse menu item. Selection may be made from different points in the tree; the scope of the effect changes based on the point selected.

- Clearing the status of the entire device tree.

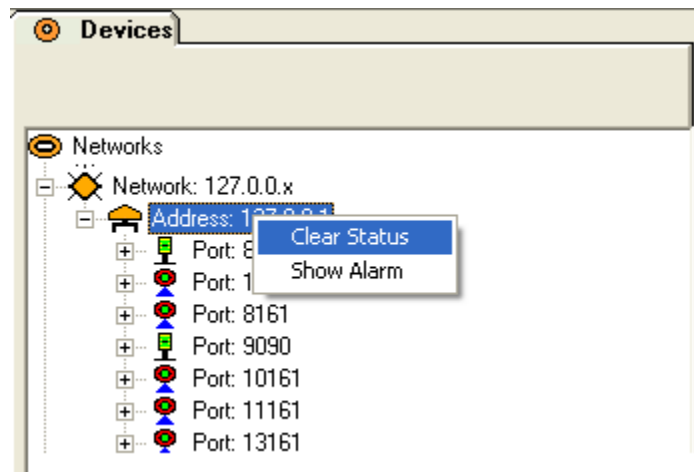


This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- Clearing the status for an individual network



- Clearing the status for an individual address

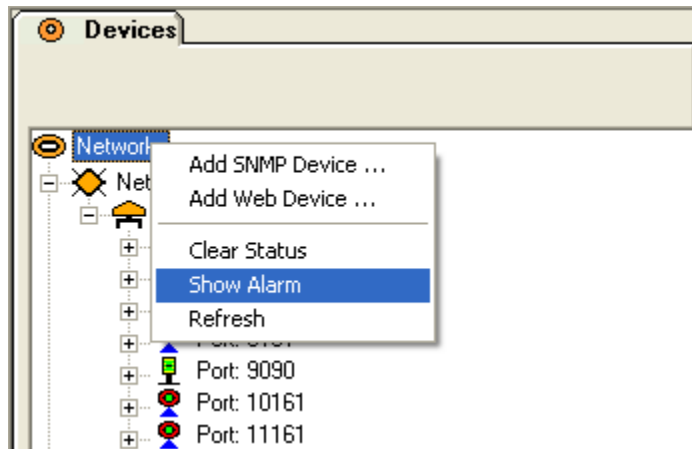


Finding Alarm Causes

Quick links back to the alarm log are available from various points in the Devices tree. Depending on the point of selection, you are taken to the individual alarm associated with the status of that node.

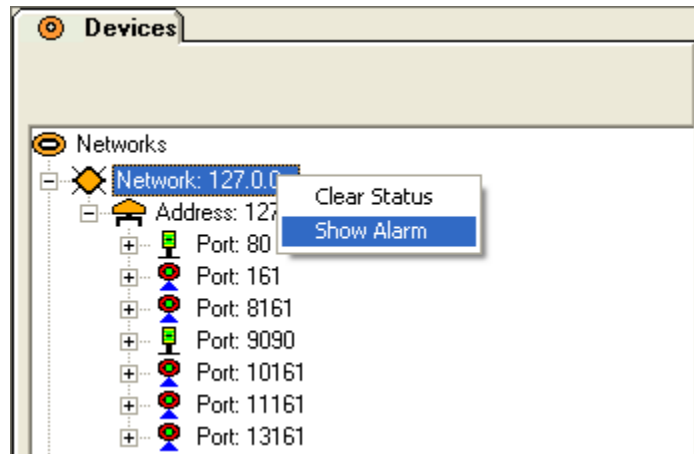
This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- To display the highest alarm detected for the entire device tree.

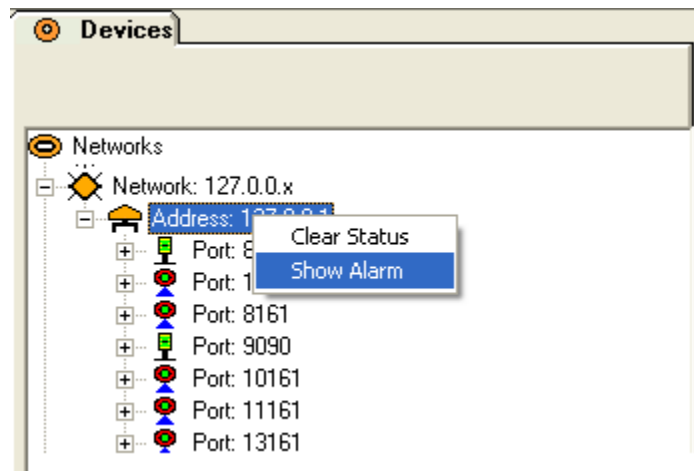


This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- To display the highest alarm detected for an individual network.



- To display the highest alarm detected for an individual address

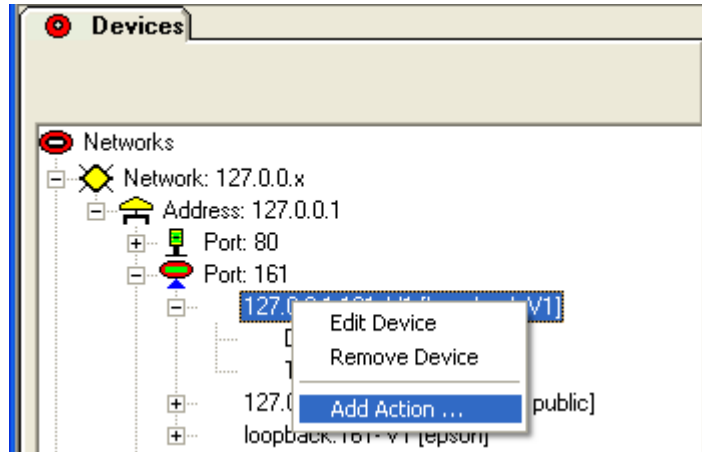


See [Alarm Sources](#) for more information on the Alarm Log.

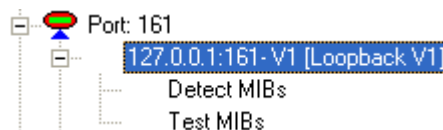
Actions and Applets

Actions are small “applications” (applets) that you can add to, delete from and initiate from the Devices view.

- Each device may have different “actions” associated with it. Actions may be added or removed by using the mouse menu.



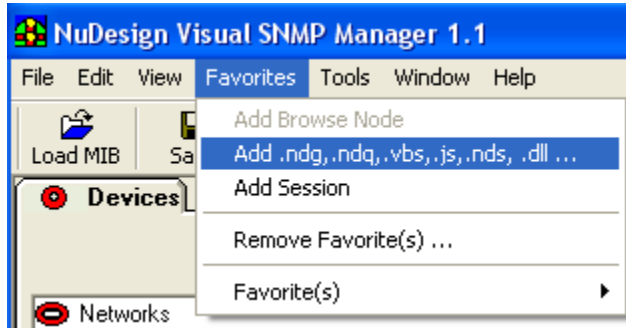
- Two default actions for SNMP based devices are created automatically. The first allows you to determine the MIBs a particular agent implements, based in the MIBs currently loaded into NuDesign Visual MIBrowser Pro™ (“**Detect MIBs**”) and the second to run a test of the each MIBs’ implementation (“**Test MIBs**”).



- A single (action) applet may be installed to multiple locations in the Devices tree.
- Actions initiated from the device view are done with the communication (session) context associated with the device service point node it is launched from.

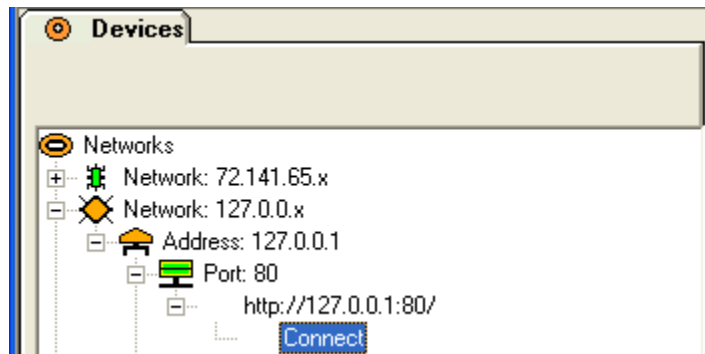
This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- Action applets may also be added as a “[Favorite](#)” under the main menu. When invoked from this facility, the action applet is invoked with the session context as the currently selected SNMP Agent. (see the Browse view)



Connecting to Web Managed Devices

You may connect to configured Web managed devices by:



Clicking any “Connect” node will start an internal web browser in the client area of Visual MIBrowser Pro™.

User Created Plug in Applets

- “Plug In” applets are small applications that enhance the functionality of Visual MIBrowser Pro™. Applets are added to Visual MIBrowser Pro as either an [“action”](#) or a [“favorite”](#).
- Plug In applets may be implemented as a user created C/C++ DLL that conforms to a simple interface that Visual MIBrowser Pro™ provides for SNMP services and alarm reporting.
- Applets may also be implemented in JScript or VB Script.
- Applets may also be a stored query (NDQ file) or query group (NDG file) based, saved from Visual MIBrowser Pro™.

Sample C++ Plugin Project “SampleDllPlugin”

A sample plug in is included with Visual MIBrowser Pro™ as a simple example of a multiple tab dialog, built as an external DLL. The DLL, “*SampleDllPlugin.dll*” is installed in the main installation directory, typically “C:\Program Files\NuDesign Tools\Visual MIBrowser Pro”.

Installing and Using “SampleDllPlugin” as a “Favorite”

To begin installation as a “Favorite”, select the main menu [“Favorites”](#) option. In the “Select Query, Group, Script or PlugIn File” browser menu that is provided, set the “Save as Type:” option to “PlugIn Files (*.dll)”. Navigate to the Visual MIBrowser Pro™ installation directory and select “*SampleDllPlugin.dll*”. Press the “Save” button to save the configuration to the **“Favorites”** list.

To use it, select the [Favorites](#) main menu option and select the “Favorite(s) >” sub menu. The plug in, with the complete path to it should be listed as an option in this sub menu. Select it to use it.

Remember, a plug in initiated as a “Favorite” is invoked with the SNMP communications options associated with the currently selected [“SNMP Agent”](#) in the **Browse** tab.

Installing and Using “SampleDllPlugin” as a device “Action”

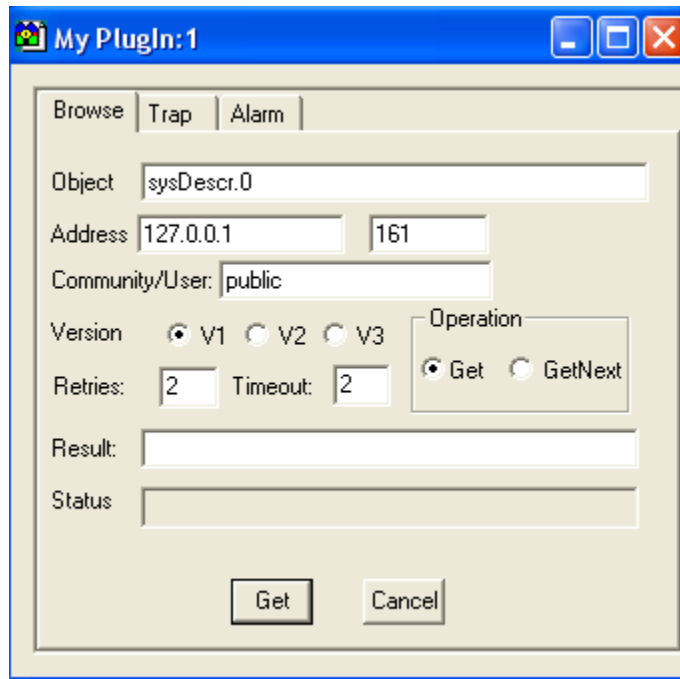
To begin installation as a device “action”, select the device you wish to associate the [action](#) with in the **Devices** tree. In the “Select Action File” browser menu that is provided, navigate to the Visual MIBrowser Pro™ installation directory and select “*SampleDllPlugin.dll*”. Press the “Open” button to commit the configuration to the selected device. The full path and plug in executables name should appear below the device’s branch.

To use the action, simply double click on the item in the action list. Note: you may activation an action multiple times.

Using “SampleDllPlugin”

There are three tabs that permit different modes of interaction with Visual MIBrowser Pro™.

- **Browse** tab. This tab’s interface allows you to access objects in the indicated agent.



You may alter the default communications parameters provided by altering the **Address** (& port), **Community/User**, **Version**, **Retries** or **Timeout** fields.

The **Object** field is used to specify which object to perform the operation on. To specify the object, you use the object name and instance form to specify objects for which the MIB has been loaded. E.g. “sysDescr.0” or “ifDescr.1”

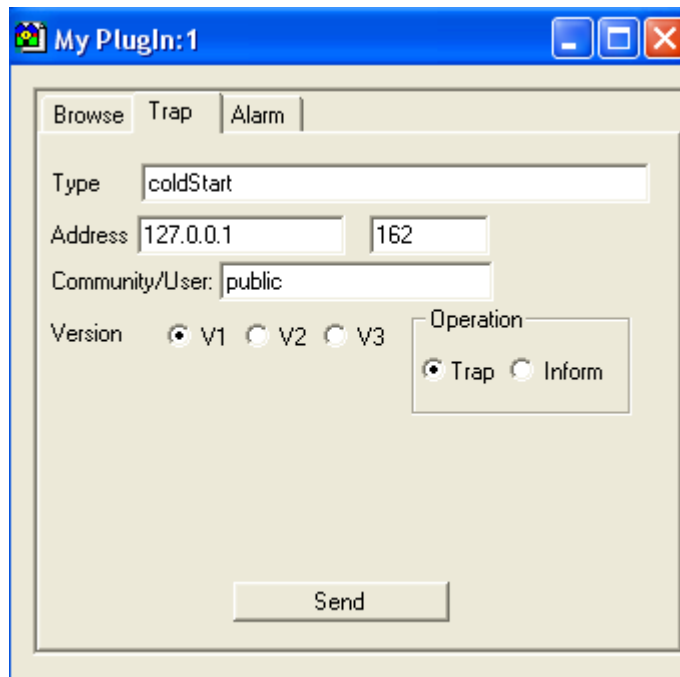
The **Operation** group permits the selection of either a **Get** or **GetNext** SNMP operation.

The **Result** will display the result if the operation was successful.

The **Status** field indicates the request status. The request has been successfully completed when the **Status** field displays “**Response Received**”. If a plug in doesn’t receive as response, the **Status** line will display “Request timed out”.

Note: if you have enabled [communication](#) error alarms, having a unsuccessful response will cause an entry to the [Alarm Log](#). In addition, if you have [forwarding](#) of alarms to the Device tree enable, you could see a state change in the Devices tree. (Depending on alarm priority, it may or may not reflect the change if there are already higher level indications for that device.)

- **Trap** tab. From this tab you may send SNMP Notification.



This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

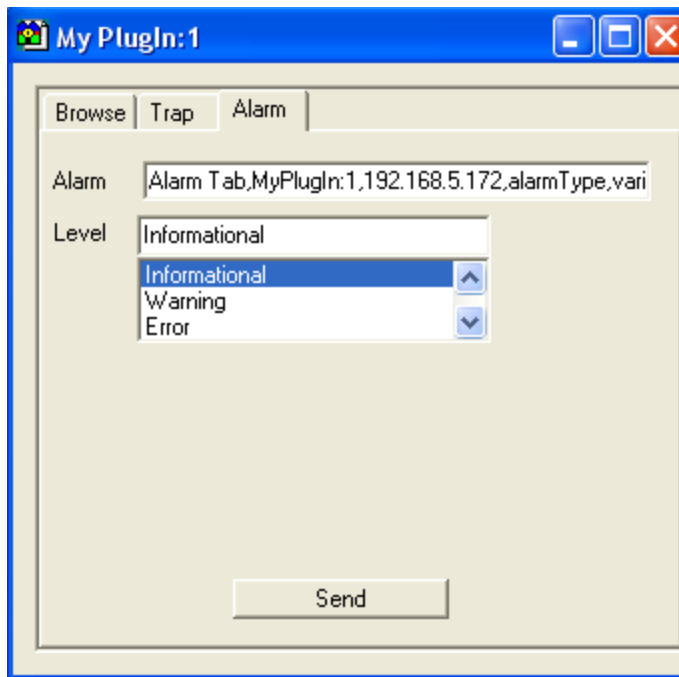
As with the previous tab, you may alter the communications parameters for the request.

The **Operation** group permits the selection of either a **Trap** or **Inform** Request SNMP operation.

Pressing “**Send**” will initiate the selected notification send operation.

If you have a notification receiver (Trap Rx window) active, given the communication parameters indicated in the image above, you will see the notification received.

- **Alarm** tab. The interface sends an alarm from the plug in to the [Alarm Log](#).



Pressing “Send” cause the alarm to be sent.

The format of the alarm is a simple CSV string. The string is separated into the component parts and display in the alarm log.

You can try altering the string content to experiment. The format is as follows.

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



- Field 1 & 2 are presented in the “**Source**” column.
- Field 3 is presented in the “**Host**” column.
- Field 4 is presented in the “**Alarm**” column.
- Field 5 is presented in the “**Variable**” column.
- Field 6 is presented in the “**Threshold**” column.
- Field 7 is presented in the “**Value**” column.
- Field 8 is presented in the “**Level**” column.

Building the sample

Located in the “SampleDllPlugin” directory in the installation directory.

The following is a list of the significant file:

- NDPlugInDefs.h, plug in interface definition include file.
- NDPlugInIf.cpp, plug in call back implementation file.
- NDPlugInIf.h, plug in call back include file.
- SampleDllPlugin.cpp, standard wizard generated application implementation file.
- SampleDllPlugin.dsp, Visual Studio 6 project file.
- SampleDllPlugin.dsw, Visual Studio 6 project workspace file.
- SampleDllPlugin.h, standard wizard generated include file
- SampleDllPlugin.rc, project resource file.
- SampleDllPlugin.sln, Visual Studio 2005 Solution file.
- SampleDllPlugin.vcproj, VC++ Project file.
- SampleDllPluginDialog.cpp, dialog and primary implementation file.
- SampleDllPluginDialog.h, dialog and primary include file.

Expression Based Charting

SNMP variables don't always provide the basis for the creating the charts or detecting the alarms you need.

As an example, the MIB II variable, *ifInOctets* provides a total count of the raw bytes that have been received on a particular network interface. This normally restricts a user to charting the rate at which packets are arriving on a particular interface. A more conventional metric for network usage is "utilization". To make this determination, you need to take into consideration the maximum capacity of a particular network interface. MIB II provides such a variable, called *ifSpeed*. Unfortunately there is no variable called *ifUtilization* available.

Note: an expression for the variable *ifUtilization* on interface 'n' for a given at time 't', given the interval t-1 to t, could be expressed as follows:

$$\text{IfUtilization.n(t)} = \frac{8 * (\text{ifInOctets.n(t)} - \text{ifInOctets.n(t-1)})}{\text{ifSpeed.n(t)}}$$

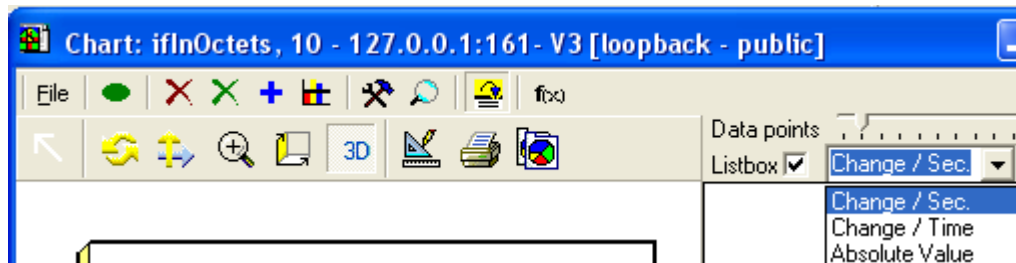
Visual MIBrowser Pro™ provides you with the ability to create Charts and detect alarm conditions on "synthetic" variables, such as above.

Without expressions, by default, Visual MIBrowser Pro™ provides the same SNMP charting facilities as other products in the family. That is, it will plot a SNMP variable as:

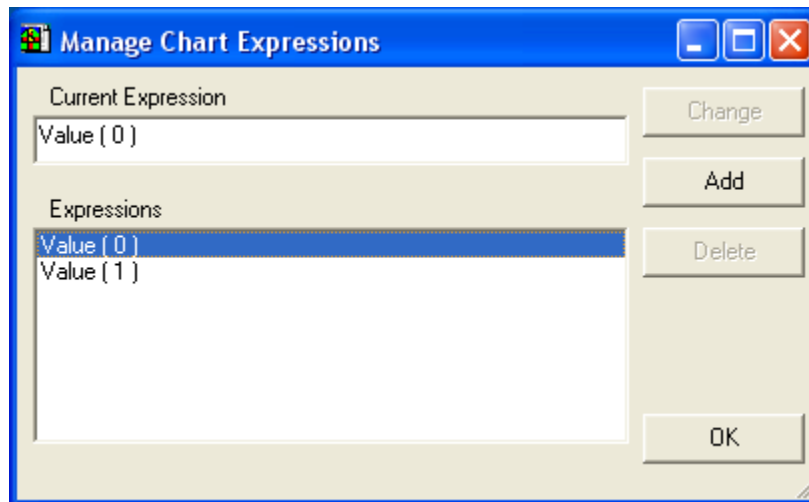
- **Absolute Value**, provides the raw value returned by the agent.
- **Change / Time**, provides the difference in value from the last sample.
- **Change / Sec.**, provides the difference in value from the last sample, normalize to a second (the default behavior).
(See the image below)

It also normalizes jitter in graphing to overcome variable latency times typical in a networked environment.

By pressing the "f(x)" button on a Chart toolbar,



you are presented an expression input Interface something like the following.



Expressions

The charting expression evaluator supports full VB script expressions, which also includes any mathematical functions VB scripting supports.

E.g. Abs(), Cos(), Log(), etc.

In addition, Visual MIBrowser Pro™ provides two functions for accessing SNMP data in the current response from and agent. These are

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- **Value(n)**. This function returns the 'n'th data item (variable) from the current response from the agent. This function only applies to the non-synthetic series (graphs) and corresponds exactly to the data item number in the SNMP variable list in the request. The data is affected by the selection for the chart, regarding the default processing of the data. E.g. "**Change / Sec.**" (see above) would cause 'Value()' to return the difference in value per second.
- **RawValue(n)**. This function returns the 'n'th data item (variable) from the current response from the agent. This function only applies to the non-synthetic series (graphs) and corresponds exactly to the data item number in the SNMP variable list in the request. The data is **not** affected by the selection for the chart, regarding default processing of the data. I.e. it always returns exactly the value provided from the response from the agent. This is particularly useful when you have the chart set to provide **Change / Time** or **Change / Sec.**, but you need access to a variable without have the change processing applied. E.g. *ifSpeed*.

In addition, Visual MIBrowser Pro™ provides several functions for returning, primarily statistical information from other series (an individual graph) in the same chart.

These are:

- **AvgValue(n)**, returns the average value in the 'n'th series (graph) in the same chart.
- **MinValue(n)** , returns the minimum value in the 'n'th series (graph) in the same chart.
- **MaxValue(n)** , returns the maximum value in the 'n'th series (graph) in the same chart.
- **CountValue(n)** , returns the count of values in the 'n'th series (graph) in the same chart.

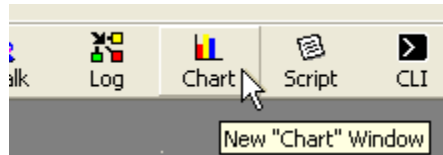


- **FirstValue(n)** , returns the first value in the 'n'th series (graph) in the same chart.
- **LastValue(n)** , returns the last value in the 'n'th series (graph) in the same chart.
- **TotalValue(n)** , returns the sum of the values in the 'n'th series (graph) in the same chart.

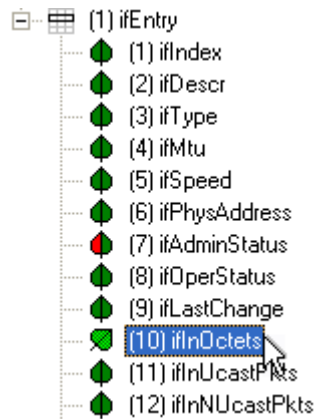
Examples:

Creating an Utilization Chart on the first network interface

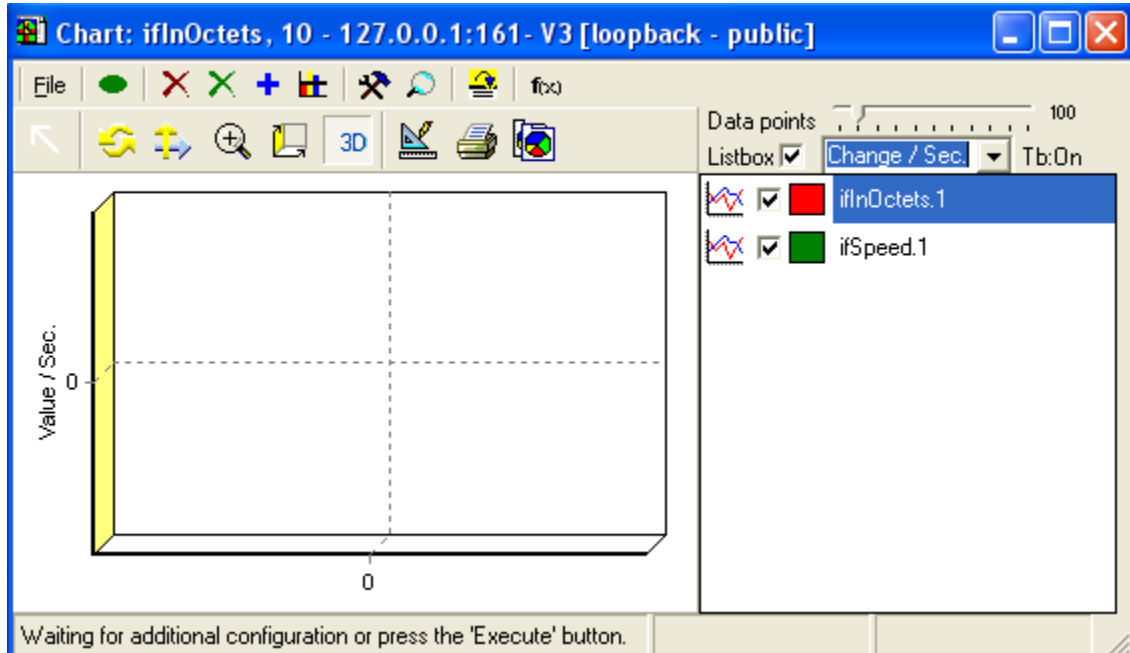
- To start, create a blank chart window by pressing the Chart button on the main tool bar.



Select the “Browse” tab and in succession, open the “*mib-2*”, “*interfaces*”, “*ifTable*”, and “*ifEntry*” nodes. Select the “*ifInOctets*” node.



Now press the ‘+’ button on the Chart toolbar. In the dialog the pops up, select the first interfaces’ index, typically “.1” and press ‘OK’. Now go back the Browse tree again, and select “*ifSpeed*” and again press the ‘+’ button and select the same index. You should see something like the following.



- Now press the '**f(x)**' button on the chart tool'. You see the interface in the [Manage Chart Expression](#) dialog.
- Type then follow text into the **Current Expression** field of the Manage Chart Expression dialog:

$$100 * 8 * \text{Value}(0) / \text{RawValue}(1)$$

Where:

- **Value(0)** is the difference in octets (bytes) for *ifInOctets.1* per second.
- **RawValue(1)** is the current value of *ifSpeed.1* in bits / second.

The expression is multiplied by eight in order to turn *ifInOctets.1* into the number of bits (which is the units *ifSpeed.1* is in). Multiplying by 100 is to express the result as a percentage.

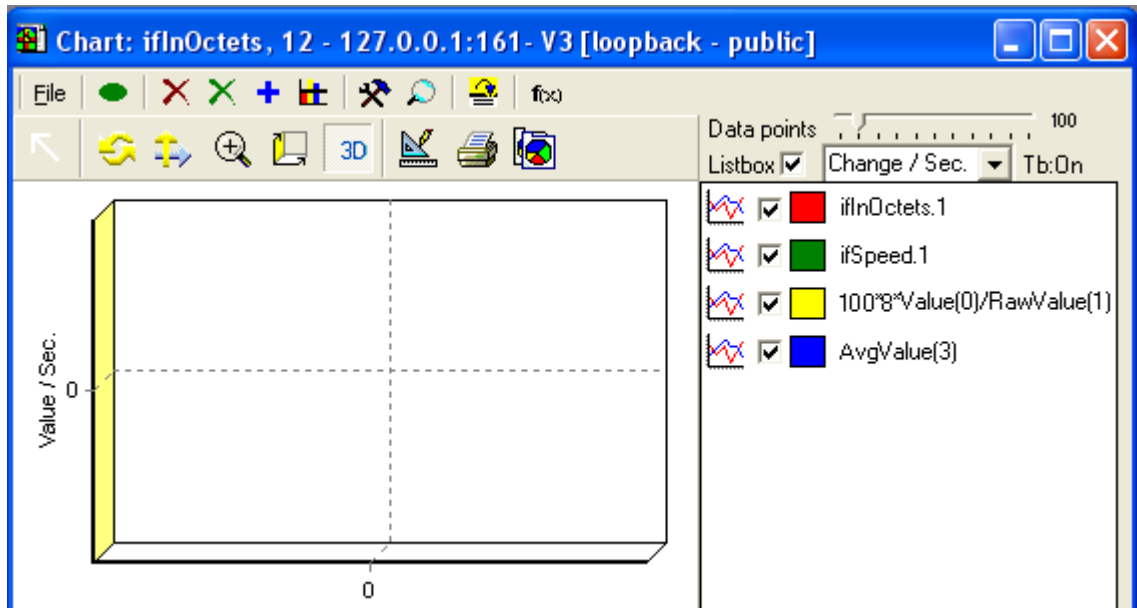
To save the expression to the chart, press the '**Add**' button.


- To create an average utilization series (graph) type:

$$\text{AvgValue}(3)$$

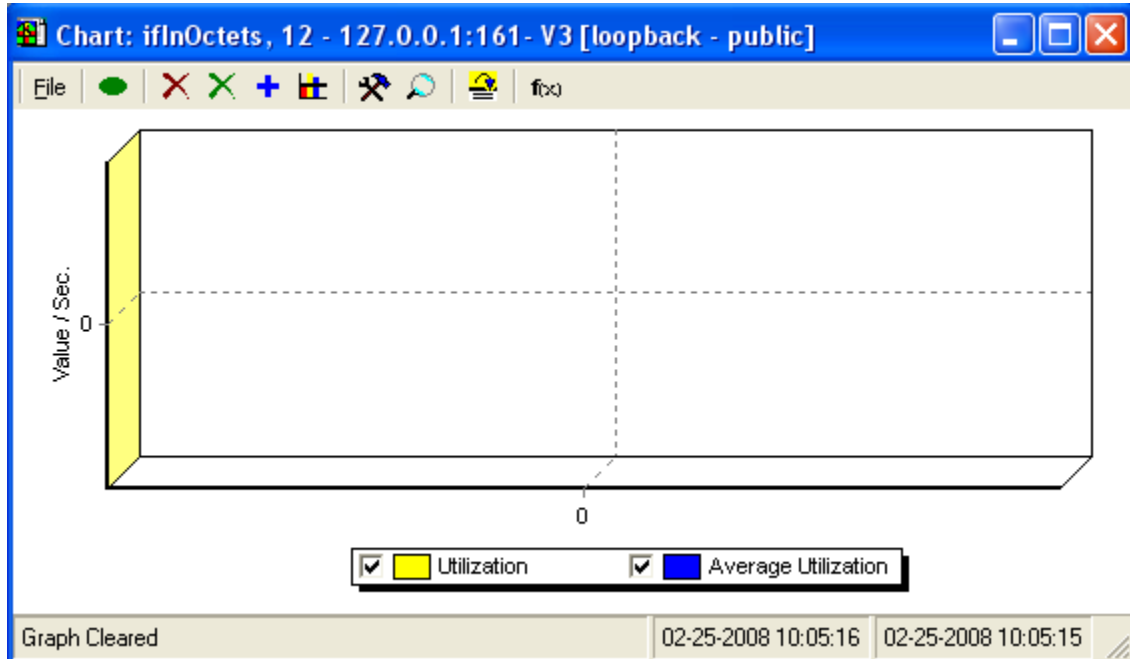
into the **Current Expression** field. (This means show the current average of series (graph) 3 at each point) To save this expression, press the 'Add' button.



- Press the 'OK' button.
- You should now see a chart that looks like the following....

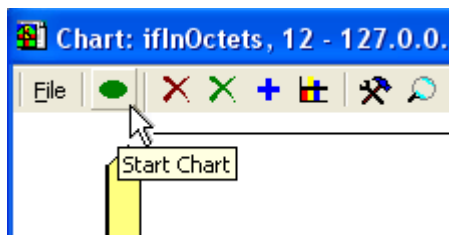


- Uncheck the ifInOctets.1 and ifSpeed.1 entries in the series list box. This will hide the graphs for the raw variables from which **Utilization** and **Average Utilization** are derived.
- Double click on '100*8*Value(0)/RawValue(1)' and change this field to 'Utilization'.
- Double click on 'AvgValue(3)' and change this field to 'Average Utilization'
- Uncheck 'Listbox' to hide the series list box view
- Press the  button to hide the secondary tool bar.

The chart should now look something like this.



- You can change the chart's sampling rate by changing the '**Poll Period**' field in the configuration dialog for the chart. To open the configuration dialog, press the  tool bar button on the chart. ([configuration dialog](#))
- You can also set up alarm condition detection, but pressing the 'Action or [Favorite](#)), by pressing the "**File | Save Query**" menu option from the chart.
- Lastly, press the start button to initiate Chart executing.



This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

Creating a Packet Rate Chart on the first network interface

Creating a packet rate chart is very similar to the above. Follow the example above, however:

- Instead of selecting '*ifInOctets.1*', select '*ifInUcastPkts.1*'. This is a count of packets that been directed to this interface.
- Instead of selecting '*ifSpeed.1*', select '*ifInNUcastPkts.1*'. This is a count of packets that have been broadcast or multicast to the network that this interface has received.
- Open up the “**Manage Chart Expression**” dialog and in the ‘**Current Expression**’ Field, enter the expression:

`Value(0) + Value(1)`

- If you want to track the average packet rate, you could add a fourth expression '`AvgValue(3)`'.
- Press '**OK**' to close the dialog and follow the same steps outlined in the previous example. (You might want to label the graphs differently.)

Creating Other Charts

There are many graphs of this nature you could configure charts for (and alarm condition detection on). The charts that you may wish to create will depend on the MIBs that are implemented by the agent.

Continuing with MIB-II/RFC1213 derived objects as we have in the above examples, here are a couple that you could created.

- **Error rate.** For a particular interface, there are two variables *ifInErrors.(n)* and *IfOutErrors.(n)*
- **Total Error.** The error rate of all network interfaces. The expression would involve the summation of all *ifInErrors* and or *ifOutErrors*.
- If transmit and receive on a network interface share the media, then *ifOutOctets* would be in the expression given in the [Utilization](#) example.

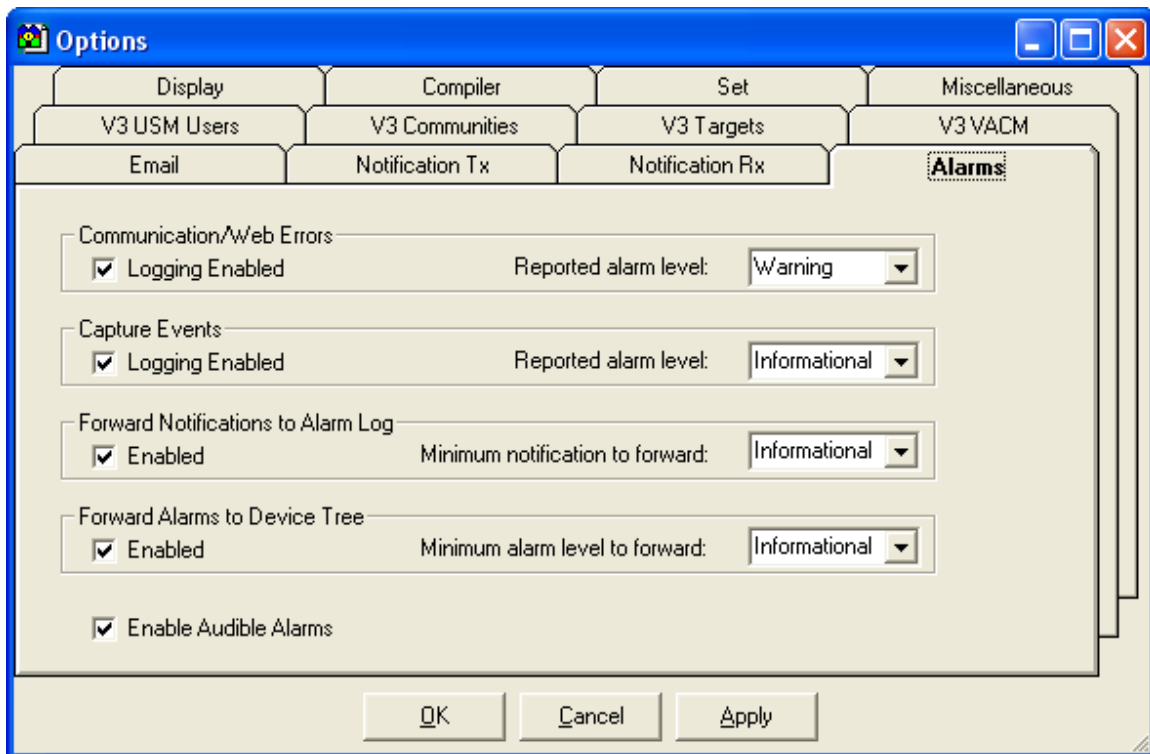


- Similarly if transmit and receive on a network interface share the media, then ifOutUcastPkts & ifOutNUcastPkts would be in the expression given in the [Packet Rate](#) example.

Configuring Alarms

Alarms Filtering and Routing

Alarms filtering and routing is managed from the main menu option, “**Tools | Options ...**” dialog. Select the “**Alarms**” dialog tab. When you do so you should see the following.



- The **Communications/Web Errors** area controls how SNMP and HTTP communication problems are managed. If the “**Logging Enabled**” is checked, then any communication problems are forwarded to the “**Alarm Log**” with the alarm level indicated by the current selection in the “**Report alarm level:**” list box.
- The **Capture Events** area controls how Start and Stop capture events from the Analyzer are managed. If the “**Logging Enabled**” is checked, then these events are forwarded to the “**Alarm Log**” with the alarm level indicated by the current selection in the “**Report alarm level:**” list box.

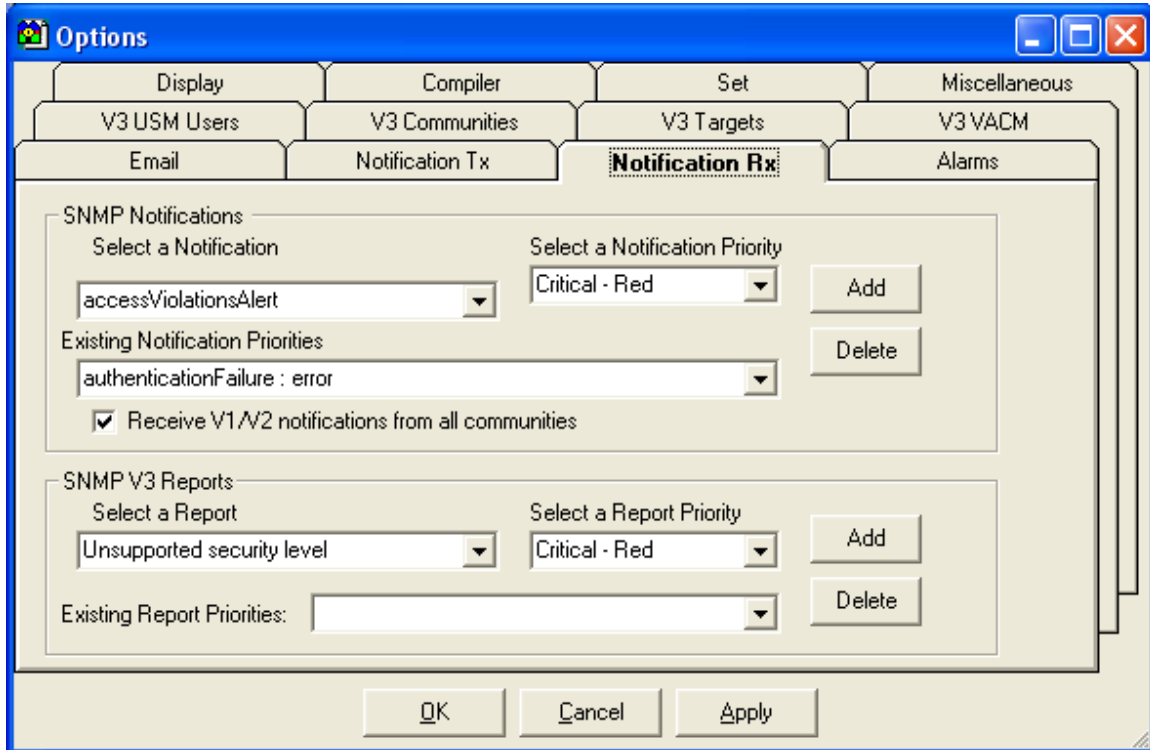
This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

- The [Notifications to Alarm Log](#) area controls how SNMP notification events received via a **Notification Receiver** are managed. SNMP V3 report messages, received as a consequence of any communications between Visual MIBrowser Pro and any agent is also control by this area. (See [Notifications and V3 Reports](#) for more information on configuring these events.) If “**Enabled**” is checked, then these events are forwarded to the “**Alarm Log**”, provided that the configured notification alarm level meets or exceeds the alarm level indicated by the current selection in the “**Minimum notification to forward:**” list box.

The **Forward Alarms to Device Tree** area controls whether alarm indications that are placed in the “Alarm Log” are also forwarded to the Devices tree. If the “**Enabled**” is checked, then these events are forwarded, provided that the configured notification alarm level meets or exceeds the alarm level indicated by the current selection in the “**Minimum notification to forward:**” list box.

SNMP Notifications and V3 Reports

Notifications and SNMP V3 message report alarm levels are configured from the main menu “**Tools | Options ...**” dialog, from the “**Notifications Rx**” tab.



The dialog is separated into two areas:

- The “**SNMP Notifications**” area controls the mapping of received SNMP notifications by any operational **Notification Receiver** to a certain alarm level.

Note: To have a SNMP Notification, such as a Trap cause an alarm condition to be evident in the Devices tree, it must be assigned some elevated alarm status. In addition, the level that it is assigned must be high enough to exceed the limits to both the “[Notifications to Alarm Log](#)” and “[Forward Alarms to Device Tree](#)” filters.

All notification definitions detected in Visual MIBrowser Pro™’s loaded MIBs are place in the “**Select a Notification**” list box. Items are moved to the “**Existing Notification Priorities**” list boxed once a particular notification is assigned an alarm level (priority).

The “**Select a Notification Priority**” contains the list of all possible alarm levels (priorities).

An alarm level (priority) association is made by selecting a notification from the “**Select a Notification**” list box, selecting an alarm level from the “**Select a Notification Priority**” and pressing the “**Add**” button.

An existing alarm level (priority) association may be deleted by selecting it from the “**Existing Notification Priorities**” list box and pressing “**Delete**”.

Once such an association is made, when any active Notification Receiver receives it, it is processed as an alarm of the associated level.

- The “**V3 SNMP Reports**” area controls the mapping of any received SNMP V3 report message.

All the available types of V3 SNMP report messages are placed in the “**Select a Report**” list box. Items are moved to the “**Existing Report Priorities**” list box once a particular report is assigned an alarm level (priority).

The “**Select a Report Priority**” contains the list of all possible alarm levels (priorities).

An alarm level (priority) association is made by selecting a report from the “**Select a Report**” list box, selecting an alarm level from the “**Select a Report Priority**” and pressing the “**Add**” button.

An existing alarm level (priority) association may be deleted by selecting it from the “**Existing Report Priorities**” list box and pressing “**Delete**”.

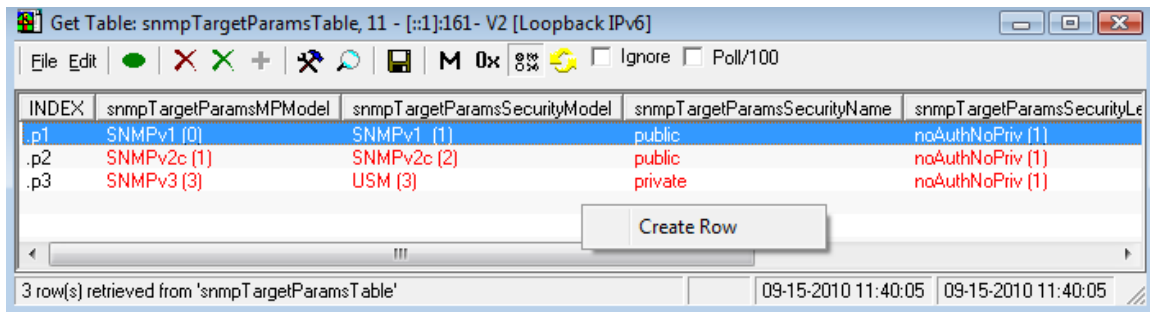
Other Features

SNMP Row Management

Several row management operations available in the “**Get Table**” window interface to simplify adding, delete and changing tables controlled by **RowStatus** variables. These features are available from the right mouse click menu in a **Get Table** window.

Creating a new Row

To create a row from a **Get Table** window, right mouse click on an area below any information that may already be visible in the window. When you do, a menu as depicted below is displayed. Selecting this menu option initiates the create row sequence.

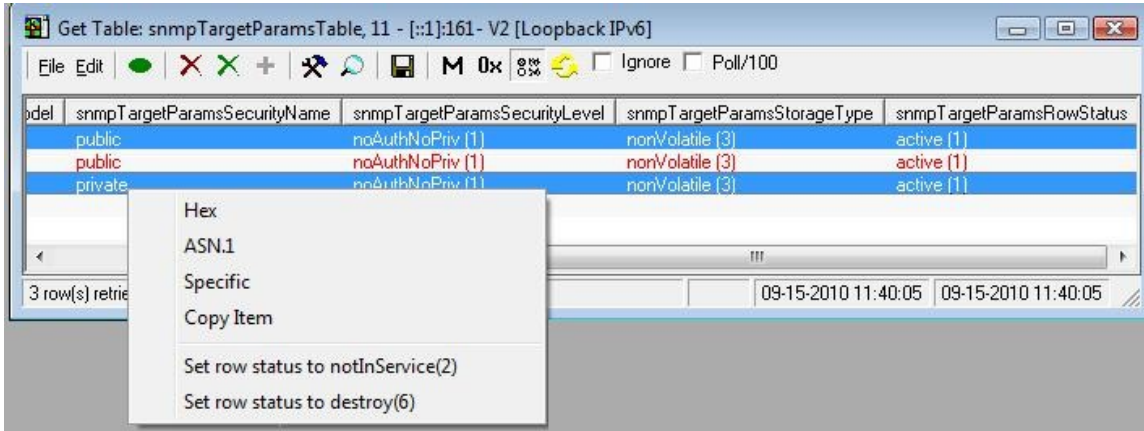


Operating on multiple rows at once

By selecting multiple rows from a **Get Table** window, it is possible to operate on all selected rows with a single operation, provided the values of all **RowStatus** variables are the same.

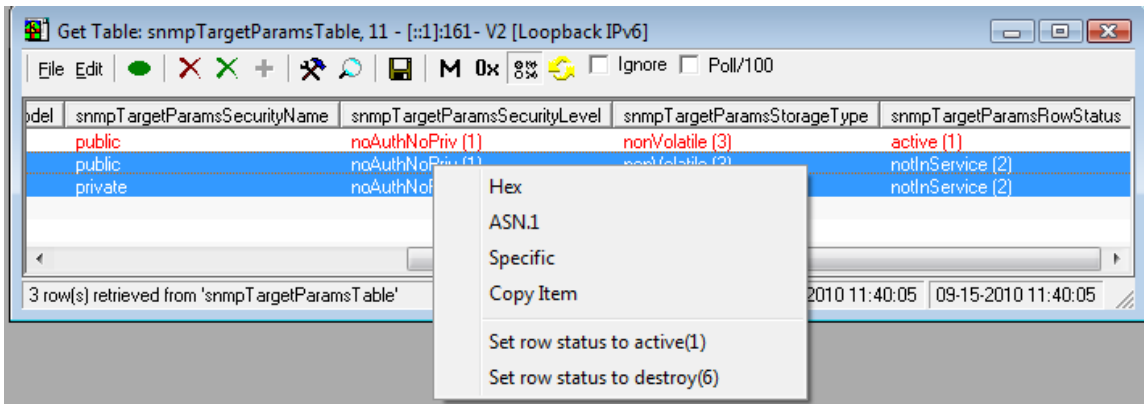
The options available from the right mouse click is context sensitive to the values of the **RowStatus** variables. For instance, if all **RowStatus** variables have the value **active (1)**, then the clicking on one of the selected rows presents the following menu options.

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



Clicking the menu option ‘Set row status to **notInService (2)**’ would set all selected rows to this value.

Whereas, if all **RowStatus** variables have the value **notInService (2)**, then the following menu would be presented.



Clicking the menu option ‘Set row status to **active (1)**’ would set all selected rows to this value.

In either case, you could select the ‘Set row status to **destroy (6)**’ option in which case all the rows selected would be deleted from the table.

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.

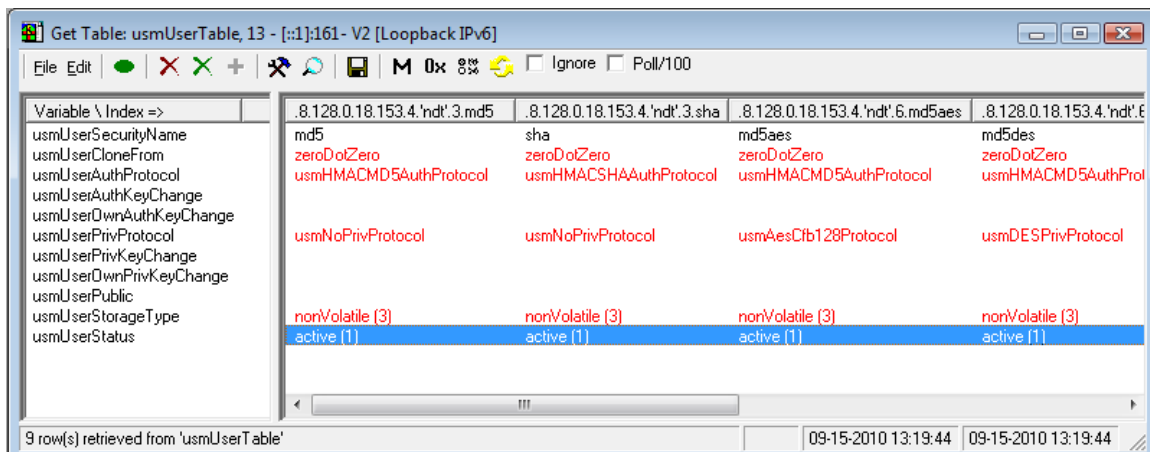
Information Display Features

“Get Table” Pivoted View


In this view, the conventional **Get Table** view is pivoted so that table “columns” are presented as row. The left area of the view depicts the variable names in the “column” and the header area of the table presents the index of the row.

In addition, the left area is separated by a “splitter” bar with that separates the pan regions of the left and right side of the table. This is so that the variable names can remain visible even though data is panned in the right.

Below depicts a typical view of this table.



| Variable \ Index => | .8.128.0.18.153.4.'ndt'.3.md5 | .8.128.0.18.153.4.'ndt'.3.sha | .8.128.0.18.153.4.'ndt'.6.md5aes | .8.128.0.18.153.4.'ndt'.6.md5des |
|-------------------------|-------------------------------|-------------------------------|----------------------------------|----------------------------------|
| usmUserSecurityName | md5 | sha | md5aes | md5des |
| usmUserCloneFrom | zeroDotZero | zeroDotZero | zeroDotZero | zeroDotZero |
| usmUserAuthProtocol | usmHMACMD5AuthProtocol | usmHMACSHAAuthProtocol | usmHMACMD5AuthProtocol | usmHMACMD5AuthProtocol |
| usmUserAuthKeyChange | | | | |
| usmUserOwnAuthKeyChange | | | | |
| usmUserPrivProtocol | usmNoPrivProtocol | usmNoPrivProtocol | usmAesCfb128Protocol | usmDESPrivProtocol |
| usmUserPrivKeyChange | | | | |
| usmUserOwnPrivKeyChange | | | | |
| usmUserPublic | | | | |
| usmUserStorageType | nonVolatile (3) | nonVolatile (3) | nonVolatile (3) | nonVolatile (3) |
| usmUserStatus | active (1) | active (1) | active (1) | active (1) |

This view is entered by pressing the  button in a **Get Table** toolbar. Pressing it again reverts it to the default view.



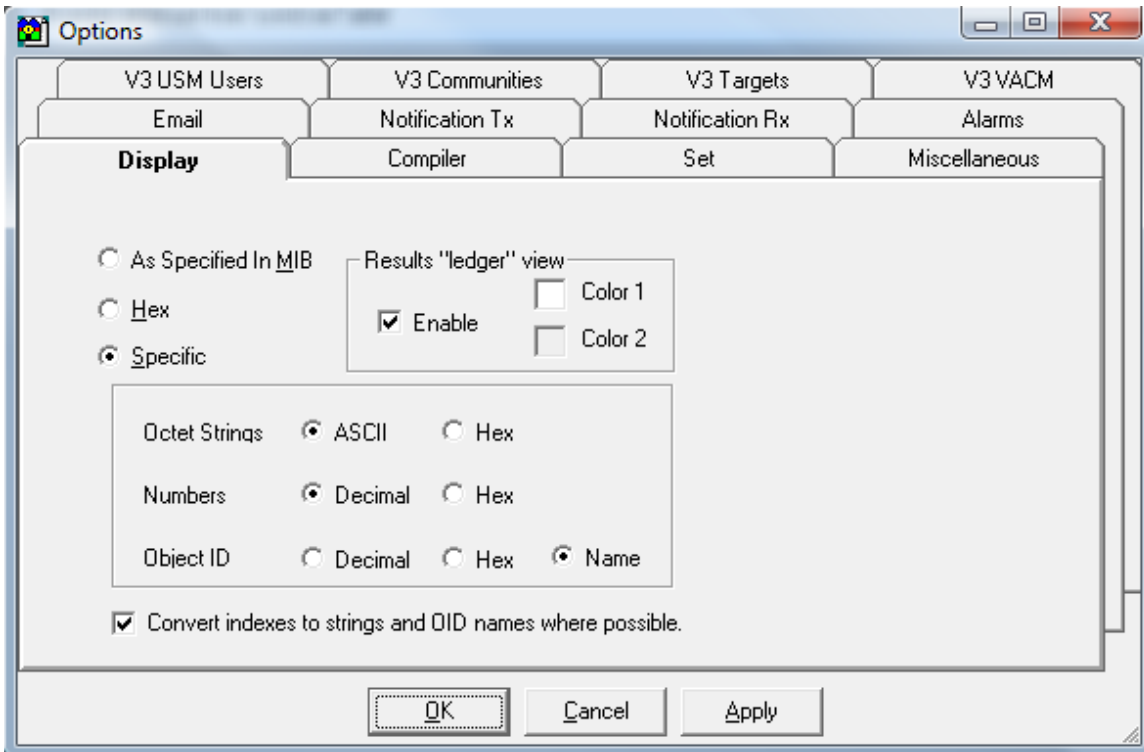
“Ledger” Mode

This mode alternates the background color between two colors of your choosing. This can be beneficial viewing information in a wide presentation. (The mode is set as a default starting with MIBrowser Pro 9.0. The default colors are white and light gray)

Below depicts a table of information with the default colors.

| INDEX | usmUserSecurityName | usmUserCloneFrom | usmUserAuthProtocol | usmUserPrivProtocol | usmUserStorageType | usmUserStatus |
|------------------------------------|---------------------|------------------|------------------------|----------------------|--------------------|---------------|
| 8.128.0.18.153.4.'ndt'.3.md5 | md5 | zeroDotZero | usmHMACMD5AuthProtocol | usmNoPrivProtocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.3.sha | sha | zeroDotZero | usmHMACSHAAuthProtocol | usmNoPrivProtocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.6.md5aes | md5aes | zeroDotZero | usmHMACMD5AuthProtocol | usmAesCib128Protocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.6.md5des | md5des | zeroDotZero | usmHMACMD5AuthProtocol | usmDESPrivProtocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.6.public | public | zeroDotZero | usmNoAuthProtocol | usmNoPrivProtocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.6.shaaes | shaaes | zeroDotZero | usmHMACSHAAuthProtocol | usmAesCib128Protocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.6.shades | shades | zeroDotZero | usmHMACSHAAuthProtocol | usmDESPrivProtocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.9.md5nopriv | md5nopriv | zeroDotZero | usmHMACMD5AuthProtocol | usmNoPrivProtocol | nonVolatile (3) | active (1) |
| 8.128.0.18.153.4.'ndt'.9.shanopriv | shanopriv | zeroDotZero | usmHMACSHAAuthProtocol | usmNoPrivProtocol | nonVolatile (3) | active (1) |

Colors are controlled from the Display tab of the Option dialog, depicted below.



This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.



The “Enable” checkbox controls whether this mode is enabled or not. You change the colors when, this mode is enabled, by double clicking on the either the color boxes or the color labels.

This document contains confidential and proprietary information. Reproduction and or disclosure through any means is prohibited unless expressed, written consent of authorized representative of NuDesign Technologies Inc. is obtained.