

I. Understanding SNMP

Note: this document is meant as an extension to the *MonitorMagic Implementation Guide* and certain parts assume that you have read and performed all operations in this document.

Before we start configuring MonitorMagic, it is important to understand the basics of the SNMP protocol.

The SNMP protocol is widely used as a monitoring and management protocol and has recently been upgraded with enhanced security. MonitorMagic uses the SNMP version 1 implementation in its current version.

SNMP basically works in two ways, retrieving information (SNMP get) and sending information (SNMP trap). In this example, we will use the SNMP get operation. SNMP get needs only the IP address of the monitored host, a community string as security and the OID (object identifier) of the monitored resource. All OIDs are documented in so-called MIBs (Management Information Base), which can be found on the Internet. Since this example uses the Microsoft Lan Manager MIB, let's start by viewing the contents of this file:

```
LanMgr-Mib-II-MIB DEFINITIONS ::= BEGIN

    --
    -- Notes:
    --
    -- This MIB is documented in "LAN Manager 2.0 Management
    -- Information Base, LAN Manager MIB Working Group, Internet
    -- Draft: LanMgr-Mib-II" by Microsoft.
    --
    -- The Windows NT implementation currently does not support
    -- the following objects:
    --
    --     svSesNumConns
    --     svAuditLogSize
    --     wkstaErrorLogSize
    --     domLogonDomain
    --

    IMPORTS
        enterprises, OBJECT-TYPE, Counter
            FROM RFC1155-SMI
        DisplayString
            FROM RFC1213-MIB;

    lanmanager OBJECT IDENTIFIER ::= { enterprises 77 }
    lanmgr-2   OBJECT IDENTIFIER ::= { lanmanager 1 }

    -- lanmgr-2 Tree

    common    OBJECT IDENTIFIER ::= { lanmgr-2 1 }
    server     OBJECT IDENTIFIER ::= { lanmgr-2 2 }
    workstation OBJECT IDENTIFIER ::= { lanmgr-2 3 }
    domain     OBJECT IDENTIFIER ::= { lanmgr-2 4 }
```

To discover which OID you want to monitor, begin to read the MIB from the start. You will notice that this MIB imports several keywords from other MIBs or RFCs. In this case, the enterprises, OBJECT-TYPE and DisplayString are imported.

Later on, we see that the first entry in the MIB begins with lanmanager OBJECT IDENTIFIER ::=, followed by enterprises 77. That is our first part of the OID discovery. The next step is that we find out what the enterprises is, so we can complete this part of the OID. Enterprises comes from RFC 1155 as shown in the MIB, so let's have a look at that RFC (<http://www.faqs.org/rfcs/rfc1155.html>).

The root of every OID starts with Internet:

internet	OBJECT IDENTIFIER ::= { iso org(3) dod(6) 1 }
directory	OBJECT IDENTIFIER ::= { internet 1 }
mgmt	OBJECT IDENTIFIER ::= { internet 2 }
experimental	OBJECT IDENTIFIER ::= { internet 3 }
private	OBJECT IDENTIFIER ::= { internet 4 }
enterprises	OBJECT IDENTIFIER ::= { private 1 }

This concludes that "enterprises" makes up for 1.3.6.1.4.1.

If we continue reading the Microsoft Lan Manager MIB, we see that enterprises is followed by number 77, so our OID for the lanmanager enterprise reads 1.3.6.1.4.1.77.

2. SNMP object types

In SNMP, monitored resources can be represented in different ways, for instance using a textual string or a number. However, there are several important types to distinguish, and they have considerable impact on the configuration in MonitorMagic. To find out which type a certain OID is, look at the OID documentation in the MIB, then the SYNTAX field.

The most important type where MonitorMagic needs additional configuration is the COUNTER type. This type will only increase and at a certain point in to reset to 0 and start counting again. This type is widely used for resources such as bytes/second through a network interface. To get the actual value, you would have to get the COUNTER value at 2 points in time, and then detract them to get the increase over the period of time. MonitorMagic has a special feature to automate this calculation, which we will discuss shortly later on.

3. Locating the OID for your resource

Let's go back to the Microsoft Lan Manager MIB and go through its contents. When scrolling through the file, we at some point get to the content as shown below:

```
svShareTable OBJECT-TYPE
    SYNTAX SEQUENCE OF SvShareEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "The table of shares on this server."
    ::= { server 27 }

svShareEntry OBJECT-TYPE
    SYNTAX SvShareEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A table entry corresponding to a single share on this server."
    INDEX { svShareName }
    ::= { svShareTable 1 }

SvShareEntry ::= SEQUENCE {
    svShareName
        DisplayString ,
    svSharePath
        DisplayString ,
    svShareComment
        DisplayString
}

svShareName OBJECT-TYPE
    SYNTAX DisplayString (SIZE (1..12))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The name of the share."
    ::= { svShareEntry 1 }

svSharePath OBJECT-TYPE
    SYNTAX DisplayString (SIZE (1..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The local name of this shared resource."
    ::= { svShareEntry 2 }

svShareComment OBJECT-TYPE
    SYNTAX DisplayString (SIZE (0..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "A comment associated with this share."
    ::= { svShareEntry 3 }
```

In this example, I would like to get the svShareName resource, which represents the name of a particular share. When looking through this section, you will notice that the svShareName is actually an entry in a table, named svShareTable. This table has several entries which all have a column named svShareName, and we're interested to get all share names on a server.

To get the OID for this resource, we start at the table root and work our way up to the top of the MIB. This table is represented by "server 27". When looking at the top of the MIB, we see:

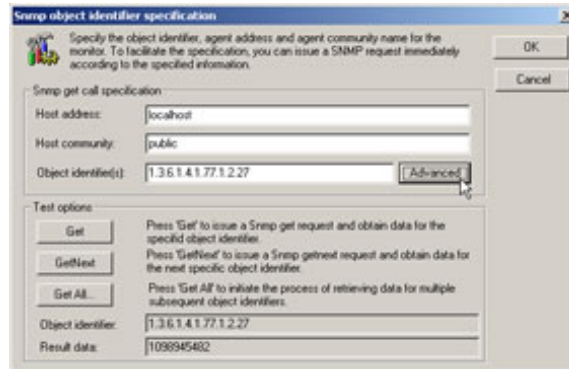
```
lanmgr-2 OBJECT IDENTIFIER ::= { lanmanager 1 }
server OBJECT IDENTIFIER ::= { lanmgr-2 2 }
```

Conclusion: the table root OID = 1.3.6.1.4.1.77.1.2.27

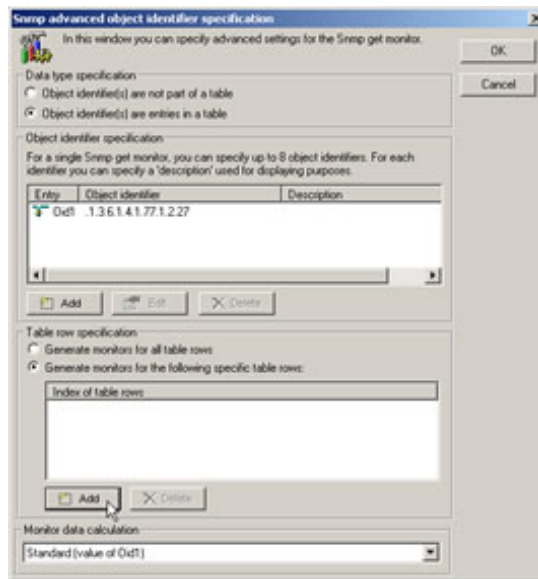
4. Creating an SNMP policy in MonitorMagic

In the previous chapters we have found out what we want to monitor and where to start. We can now use MonitorMagic to put everything together.

Create a policy called "SNMP Shares". In this policy, create a new "Snm Get" monitor. In the following window, click **Add** to define a new SNMP OID. Specify the host IP address, community string and the OID found in the previous chapter, as shown below.

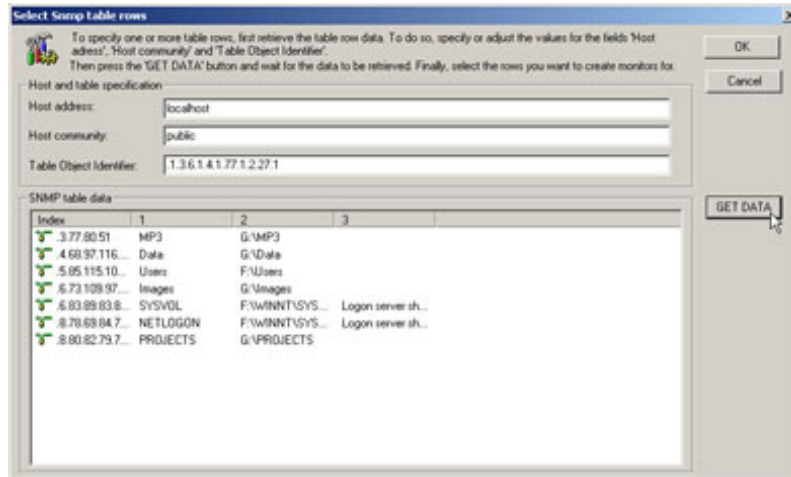


Click the **Advanced** button to access the advanced SNMP configuration. In this window, select the options as shown below:

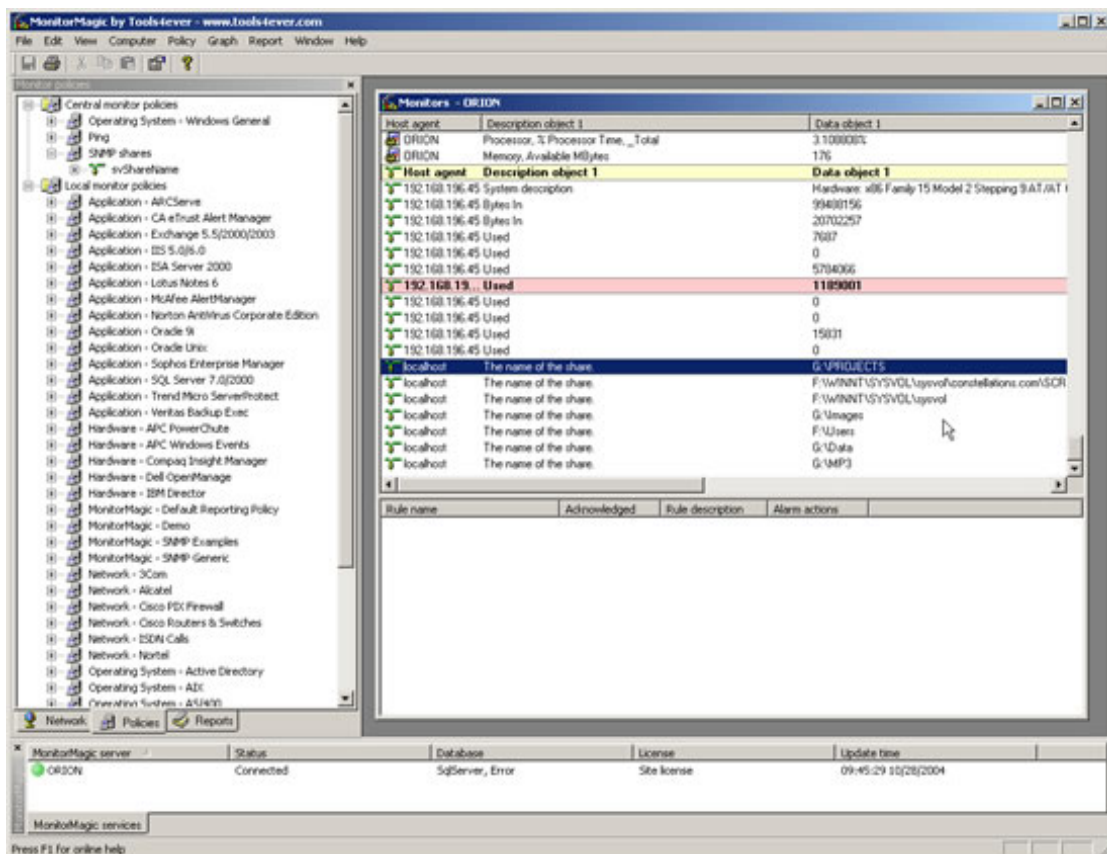


In the next dialog, we can test the results of the SNMP query on the svShareTable resource.

In this dialog, make sure to append .1 after the OID to make sure that MonitorMagic starts at the table root for each entry and click **GET DATA**. You will see results similar to the data shown below.

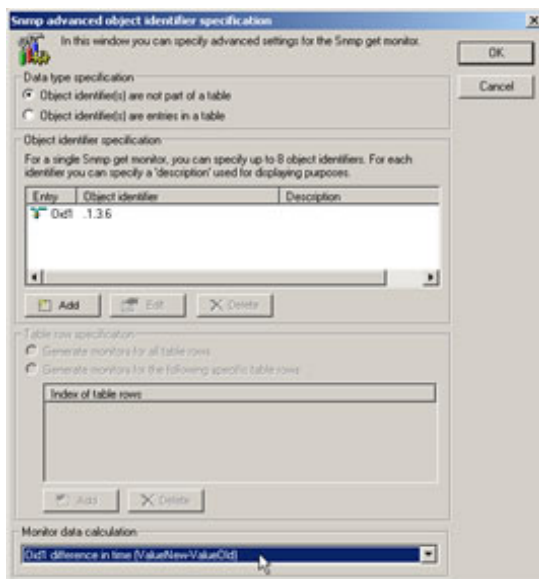


We see that the second column contains the name and corresponds with the information from the MIB, where svShareName was also the second column. We are now sure that we can monitor this resource correctly, so click **OK** to exit this window. In the next dialog, toggle the switch "**Generate monitors for all rows**", and then click the **Edit** button in the Object Identifier specification frame. In this dialog, you can enter a description for the svShareName resource, which will be displayed in the global alarm window and in e-mail notifications, so it is important to enter a clear description. Tools4ever recommends copying and pasting the information from the MIB, which would result in: "The name of the share." Click OK to return to the SNMP monitor specification, and then click OK to return to the list of SNMP monitors. Modify the name to "svShareName", and then click OK to confirm the new policy. Now, apply the policy by dragging it to a custom SNMP host in the network tree.



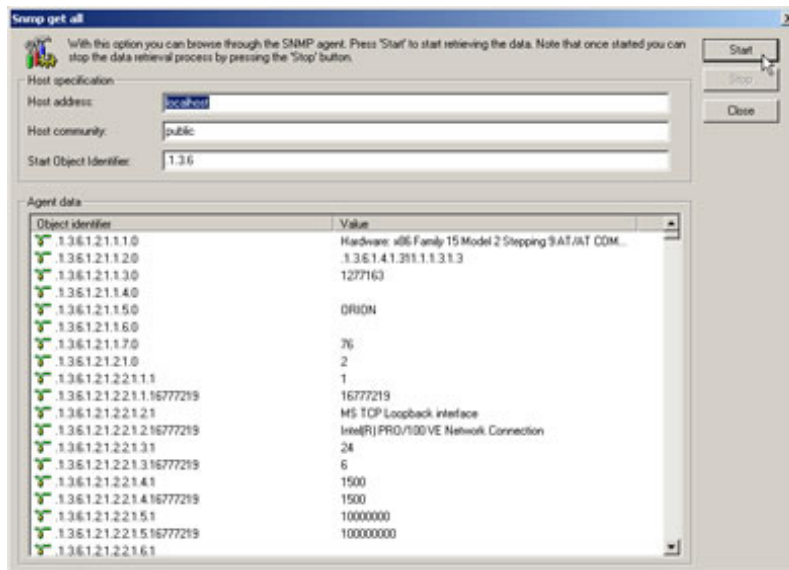
5. Configuration for COUNTER data types

When monitoring resources with a COUNTER data type, MonitorMagic requires extra configuration as shown in the screenshot below. When you enable this calculation method, MonitorMagic will automatically retrieve the contents of the OID you selected, wait until the second evaluation has passed and then perform the calculation. The result will be stored in the variable `%RESULT_DATA_DATA_CALCULATED_VALUE%`, while the original data is stored in the `%RESULT_DATA_DATA_OBJECT_X%`.



6. Using the GET ALL feature

To perform diagnostics, MonitorMagic can get all the SNMP information from a device using the GET ALL feature. To use this, access the Get All window, specify a host address, community string and start OID. To get all the SNMP information, use 1.3.6 as start OID. To get only private enterprises information, use 1.3.6.1.4.1 as start OID. As you can see below, the SNMP information starts with 1.3.6.1.2.1.1.1.0, which is the root of the MIB-II convention. Every known SNMP supported device must return this basic information, such as host name, description and uptime. The 1.3.6.1.2.1.1.1.0 OID is an excellent OID to see if a device supports SNMP at all, since this must return a value at all times.



7. Multiple OIDs in a single monitor

In some cases, there is a need to monitor multiple OIDs in a single monitor. For instance, you want to monitor the bandwidth consumption on a network interface. When you receive an alarm action, you want to know which interface was responsible for a certain amount of traffic. So there's already a problem, since the value and the name of the interface are 2 OIDs.

Let's have a look at the policy "**MonitorMagic – SNMP examples**". The first monitor, the Disk usage table, contains multiple OIDs for a single monitor. Go to the properties of the monitor, click the **Edit** button and then access the **Advanced** properties. You will see a list of 3 OIDs listed: Used, Size and Name.

This means the following:

Value for OID1 will be stored in %RESULT_DATA_DATA_OBJECT_1%

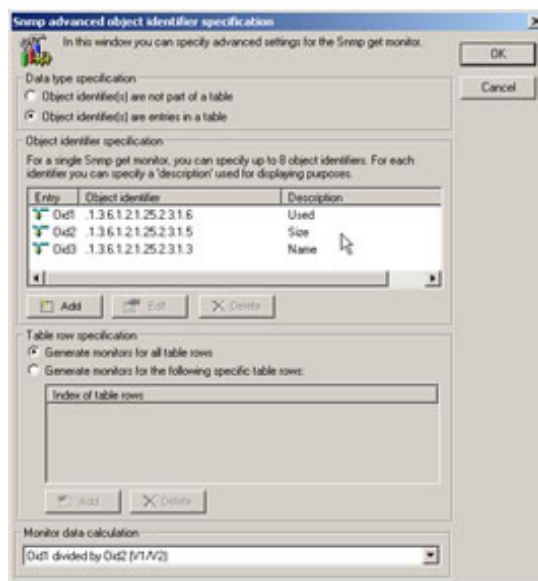
Description for OID1 will be stored in %RESULT_DATA_DESCRIPTION_OBJECT_1%

Value for OID2 will be stored in %RESULT_DATA_DATA_OBJECT_2%

Description for OID2 will be stored in %RESULT_DATA_DESCRIPTION_OBJECT_2%

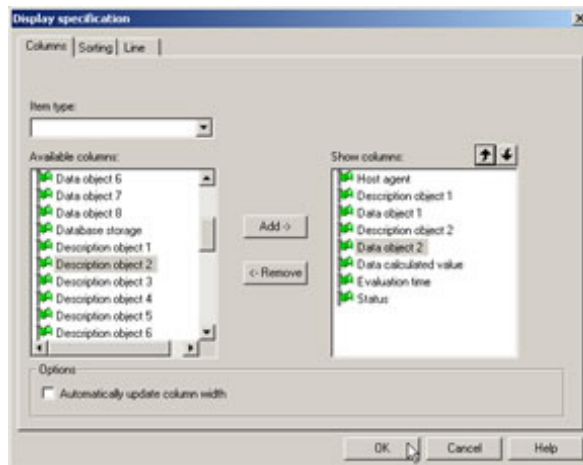
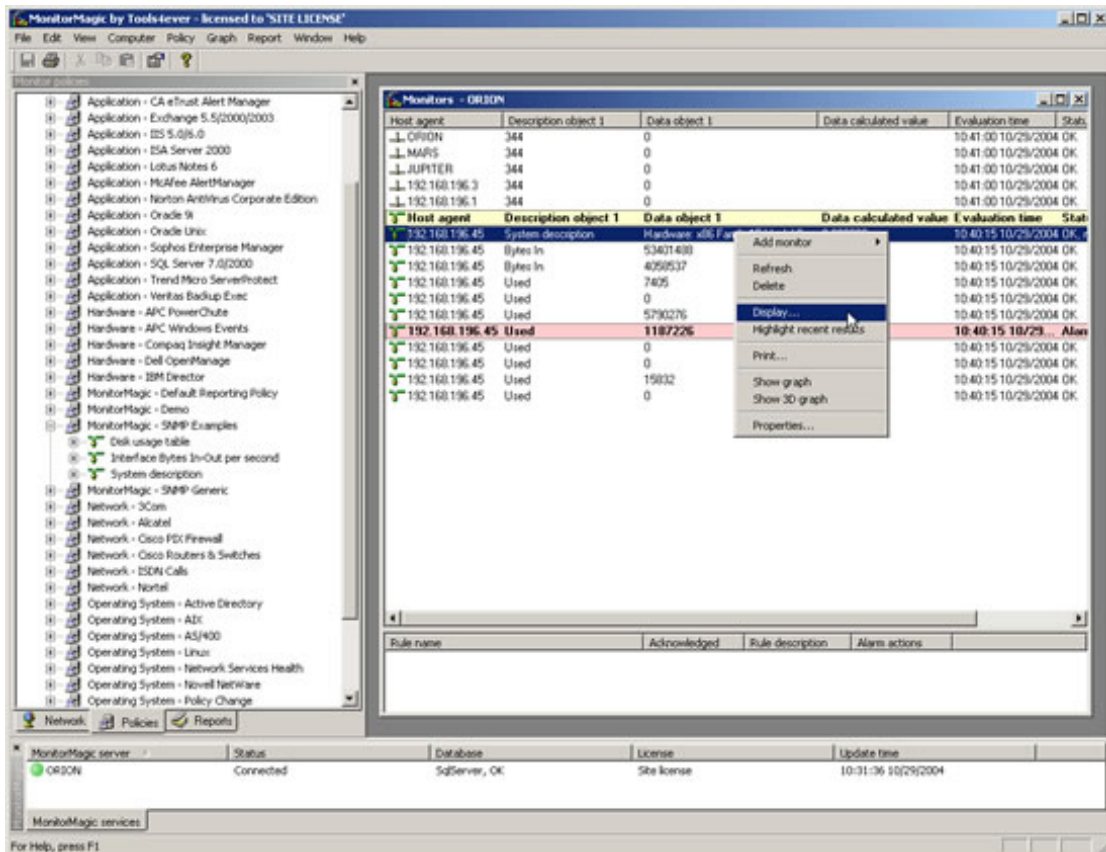
Value for OID3 will be stored in %RESULT_DATA_DATA_OBJECT_3%

Description for OID3 will be stored in %RESULT_DATA_DESCRIPTION_OBJECT_3%



When applying this policy on a "Manual SNMP host", notice that by default the monitor window will only display the first description and data object, along with the data calculated value. The data calculated value contains the result of any additional calculation such as when using a COUNTER type OID, as discussed earlier. By default, the data calculated value contains the same value as the first data object, converted to a numeric value. You will also notice that the data calculated value for the second and third monitor is empty. This is because these monitors use the COUNTER type and need a second evaluation to determine the change since the last evaluation. When you refresh these monitors, the data calculated value will appear.

To access the other data and description objects, select an SNMP monitor, right-click and select **Display**, as shown below. Now add the data and description objects in the right column.



The final result will look similar to the screenshot below:

