Network Ferret ™

MIB Analysis

V19

Contents

1: What Is MIB Analysis

Network Ferret - MIB Analysis is essentially a MIB Browser on a MIB Dump rather than against a live device or simulation.

MIB Analysis scans MIB definitions and builds a tree just like every other MIB definition scanner out there.

NMS Shops Collect Dumps

Any NMS shop whether it is a product shop or a consulting shop is going to collect SNMP MIB Dumps. It is unavoidable.

Dumps are collected to load into a MIB Simulator and test NMS software. Dumps are collected to plan out new features. Dumps are collected when customers have bugs.

What MIB Analysis is Not

MA is not a perfect parser of the grammar of a MIB definition. Our goal was to read MIBs well enough to present the OIDs and descriptions and be able to use that to display the data in a MIB dump.

We have parsed over a thousand actual MIB definition files that humans read and it works pretty well. If you come across a real file that is not parsed correctly, please send it to us. support@altmantech.com.

Feel free to send us MIBs (or a link, or an RFC number) and we will incorporate them into the download site.

We have not tried every MIB dump format out there. MA will not work with MIB dumps that do partial translations of the OIDs into text. If we fail to read a dump format, send us a small sample and we will deal with it.

What Can MA Do For You?

- Present the data from a MIB Dump in a nice browsing format.
    - Export table data in CSV format
- Scan MIB definition files and present a tree format of the OIDs.
- Generate code to represent OID constants in programs.

2: Quick Start

Installation

Unzip the distribution and follow the instructions in the readme. It is necessary to know where the machine's Java installation is.

Sample Database

A sample database can be downloaded (See Chapter 3 – Importing/Exporting). It contains standard RFC MIBs plus some vendors such as Cisco and Juniper which also brought in some ISO and Experimental MIBs.

This would be the best option to determine if you like MA's features before spending the time parsing your own MIB files.

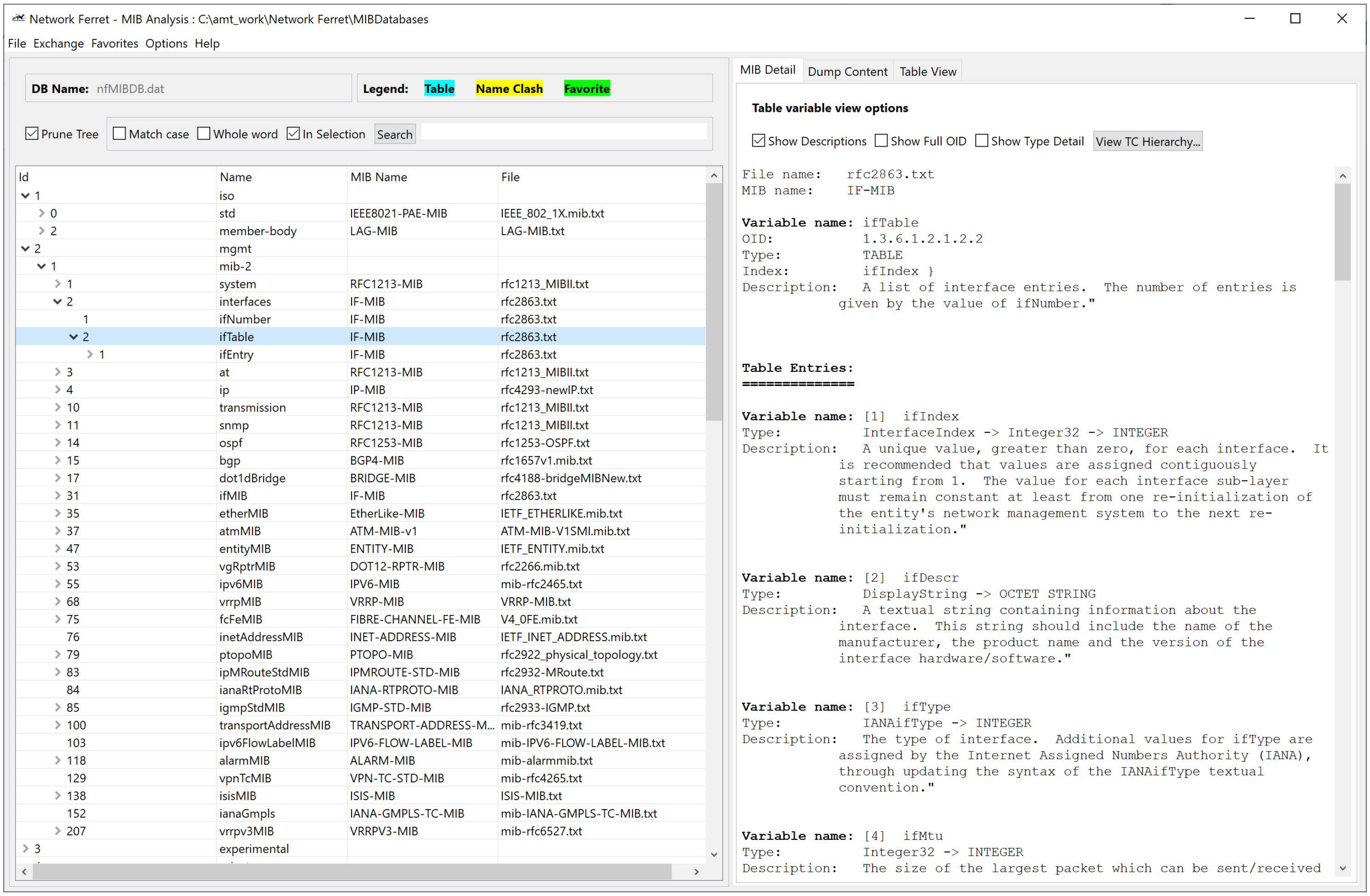
Optionally, you can parse your own MIB definition files (See Chapter 4).

Start MA

Start MDM and do File->Load a Database. Select the sample database that was downloaded or the database saved after scanning.

Browse to the well-known ifTable.

See graphic on next page


Network Ferret - MIB Analysis : C:\amt_work\Network Ferret\MIBDatabases
File Exchange Favorites Options Help
MIB Detail
Dump Content
Table View
Search
View TC Hierarchy...

DB Name: nfMIBDB.dat

Legend: Table Name Clash Favorite

☑ Prune Tree ☐ Match case ☐ Whole word ☑ In Selection

Id	Name	MIB Name	File
1	iso		
0	std	IEEE8021-PAE-MIB	IEEE_802_1X.mib.txt
2	member-body	LAG-MIB	LAG-MIB.txt
2	mgmt		
1	mib-2		
1	system	RFC1213-MIB	rfc1213_MIBII.txt
2	interfaces	IF-MIB	rfc2863.txt
1	ifNumber	IF-MIB	rfc2863.txt
2	ifTable	IF-MIB	rfc2863.txt
1	ifEntry	IF-MIB	rfc2863.txt
3	at	RFC1213-MIB	rfc1213_MIBII.txt
4	ip	IP-MIB	rfc4293-newIP.txt
10	transmission	RFC1213-MIB	rfc1213_MIBII.txt
11	snmp	RFC1213-MIB	rfc1213_MIBII.txt
14	ospf	RFC1253-MIB	rfc1253-OSPF.txt
15	bgp	BGP4-MIB	rfc1657v1.mib.txt
17	dot1dBridge	BRIDGE-MIB	rfc4188-bridgeMIBNew.txt
31	ifMIB	IF-MIB	rfc2863.txt
35	etherMIB	EtherLike-MIB	IETF_ETHERLIKE.mib.txt
37	atmMIB	ATM-MIB-v1	ATM-MIB-V1SMI.mib.txt
47	entityMIB	ENTITY-MIB	IETF_ENTITY.mib.txt
53	vgRptrMIB	DOT12-RPTR-MIB	rfc2266.mib.txt
55	ipv6MIB	IPV6-MIB	mib-rfc2465.txt
68	vrrpMIB	VRRP-MIB	VRRP-MIB.txt
75	fcFeMIB	FIBRE-CHANNEL-FE-MIB	V4_0FE.mib.txt
76	inetAddressMIB	INET-ADDRESS-MIB	IETF_INET_ADDRESS.mib.txt
79	ptopoMIB	PTOPO-MIB	rfc2922_physical_topology.txt
83	ipMRouteStdMIB	IPMROUTE-STD-MIB	rfc2932-MRoute.txt
84	ianaRtProtoMIB	IANA-RTPROTO-MIB	IANA_RTPROTO.mib.txt
85	igmpStdMIB	IGMP-STD-MIB	rfc2933-IGMP.txt
100	transportAddressMIB	TRANSPORT-ADDRESS-M...	mib-rfc3419.txt
103	ipv6FlowLabelMIB	IPV6-FLOW-LABEL-MIB	mib-IPV6-FLOW-LABEL-MIB.txt
118	alarmMIB	ALARM-MIB	mib-alarmmib.txt
129	vpnTcMIB	VPN-TC-STD-MIB	mib-rfc4265.txt
138	isisMIB	ISIS-MIB	ISIS-MIB.txt
152	ianaGmpls	IANA-GMPLS-TC-MIB	mib-IANA-GMPLS-TC-MIB.txt
207	vrrpv3MIB	VRRPV3-MIB	mib-rfc6527.txt
3	experimental		

Table variable view options

☑ Show Descriptions ☐ Show Full OID ☐ Show Type Detail

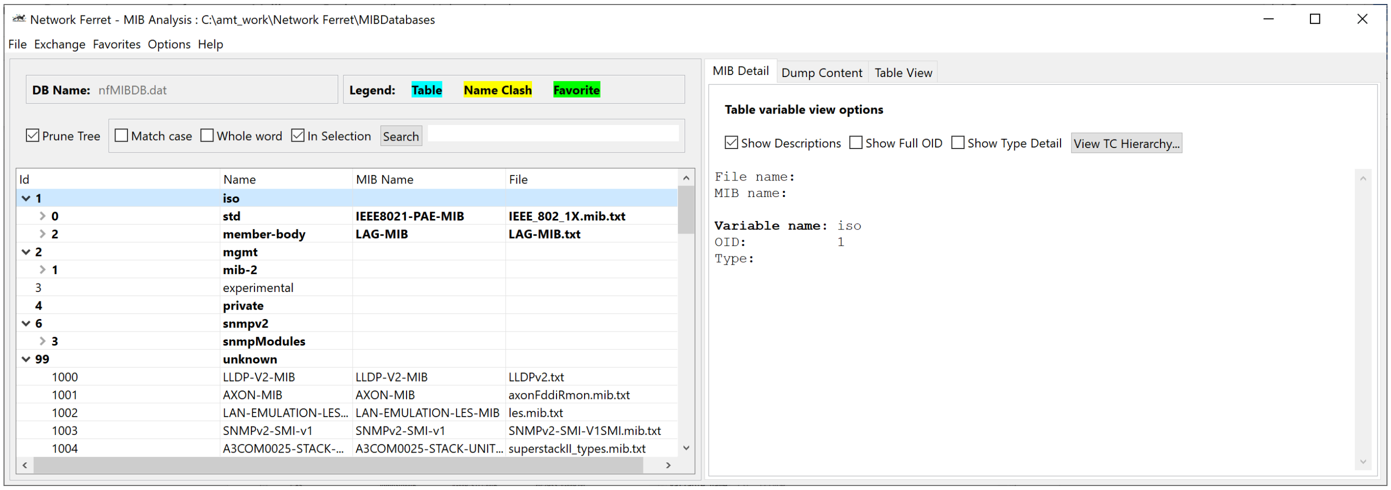
```
File name:   rfc2863.txt
MIB name:    IF-MIB

Variable name: ifTable
OID:           1.3.6.1.2.1.2.2
Type:          TABLE
Index:         ifIndex }
Description:   A list of interface entries.  The number of entries is
            given by the value of ifNumber."



Table Entries:
===============

Variable name: [1]  ifIndex
Type:          InterfaceIndex -> Integer32 -> INTEGER
Description:   A unique value, greater than zero, for each interface.  It
            is recommended that values are assigned contiguously
            starting from 1.  The value for each interface sub-layer
            must remain constant at least from one re-initialization of
            the entity's network management system to the next re-
            initialization."


Variable name: [2]  ifDescr
Type:          DisplayString -> OCTET STRING
Description:   A textual string containing information about the
            interface.  This string should include the name of the
            manufacturer, the product name and the version of the
            interface hardware/software."


Variable name: [3]  ifType
Type:          IANAifType -> INTEGER
Description:   The type of interface.  Additional values for ifType are
            assigned by the Internet Assigned Numbers Authority (IANA),
            through updating the syntax of the IANAifType textual
            convention."


Variable name: [4]  ifMtu
Type:          Integer32 -> INTEGER
Description:   The size of the largest packet which can be sent/received
```

This is pretty much what one would expect to see in a MIB definition browser. The OID tree on the left. The definition detail on the right.

Analyze A Dump

The distribution comes with a few small MIB dumps in the dumps subdirectory. Select **File->Analyze A MIB Dump** and choose one of the dumps.

The dump will be scanned and any branches that contain data will become **BOLD**.


Network Ferret - MIB Analysis : C:\amt_work\Network Ferret\MIBDatabases
File Exchange Favorites Options Help
MIB Detail
Dump Content
Table View
View TC Hierarchy...
Search

DB Name: nfMIBDB.dat

Legend: Table | Name Clash | Favorite

☑ Prune Tree ☐ Match case ☐ Whole word ☑ In Selection

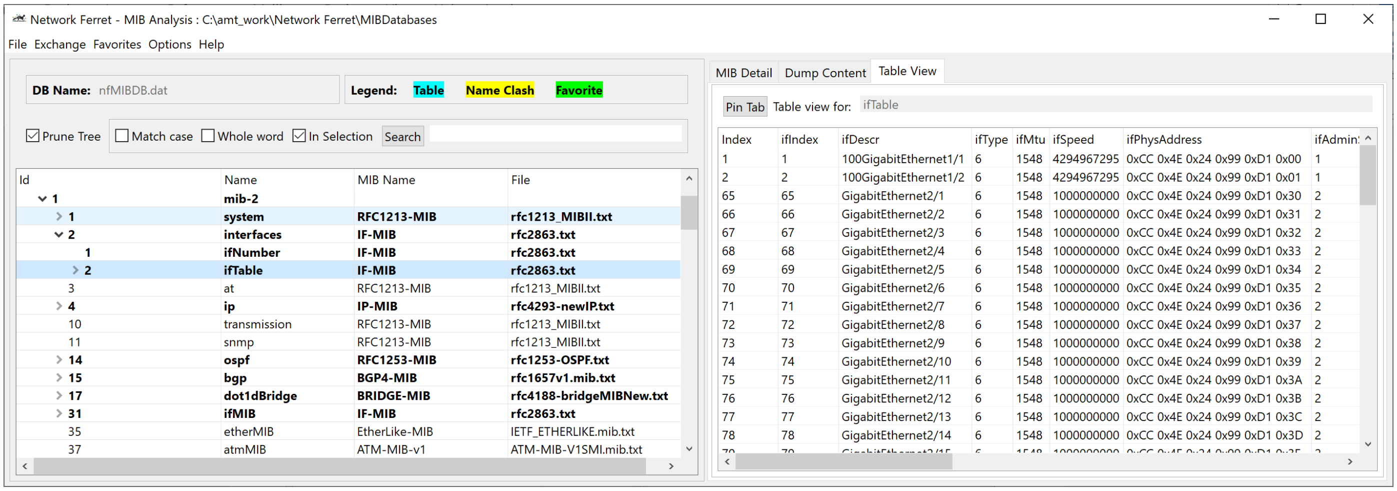
Id	Name	MIB Name	File
1	**iso**		
0	**std**	**IEEE8021-PAE-MIB**	**IEEE_802_1X.mib.txt**
2	**member-body**	**LAG-MIB**	**LAG-MIB.txt**
2	**mgmt**		
1	**mib-2**		
3	experimental		
4	**private**		
6	**snmpv2**		
3	**snmpModules**		
99	**unknown**		
1000	LLDP-V2-MIB	LLDP-V2-MIB	LLDPv2.txt
1001	AXON-MIB	AXON-MIB	axonFddiRmon.mib.txt
1002	LAN-EMULATION-LES...	LAN-EMULATION-LES-MIB	les.mib.txt
1003	SNMPv2-SMI-v1	SNMPv2-SMI-v1	SNMPv2-SMI-V1SMI.mib.txt
1004	A3COM0025-STACK-...	A3COM0025-STACK-UNIT...	superstackII_types.mib.txt

Table variable view options

☑ Show Descriptions ☐ Show Full OID ☐ Show Type Detail

```
File name:
MIB name:

Variable name: iso
OID:           1
Type:
```

Browse back to the well-known ifTable. Select the Table View tab on the right. There is a nice presentation of the ifTable data.


Network Ferret - MIB Analysis : C:\amt_work\Network Ferret\MIBDatabases
File Exchange Favorites Options Help
MIB Detail
Dump Content
Table View
Pin Tab
Search

DB Name: nfMIBDB.dat

Legend: Table | Name Clash | Favorite

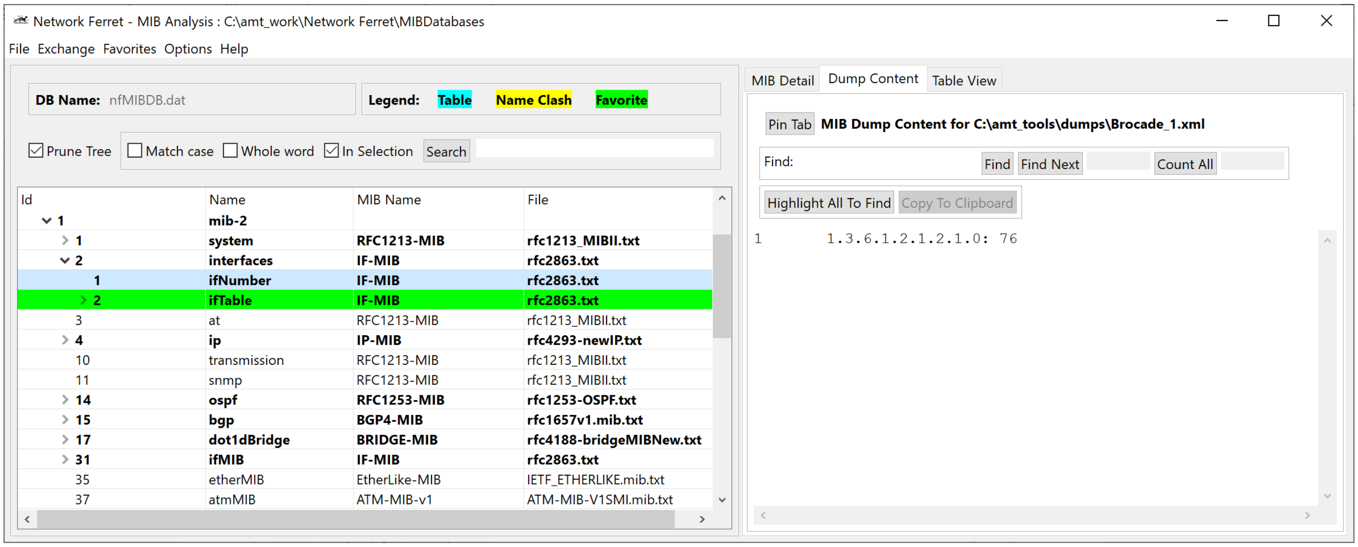
☑ Prune Tree ☐ Match case ☐ Whole word ☑ In Selection

Id	Name	MIB Name	File
1	**mib-2**		
1	**system**	**RFC1213-MIB**	**rfc1213_MIBII.txt**
2	**interfaces**	**IF-MIB**	**rfc2863.txt**
1	**ifNumber**	**IF-MIB**	**rfc2863.txt**
2	**ifTable**	**IF-MIB**	**rfc2863.txt**
3	at	RFC1213-MIB	rfc1213_MIBII.txt
4	**ip**	**IP-MIB**	**rfc4293-newIP.txt**
10	transmission	RFC1213-MIB	rfc1213_MIBII.txt
11	snmp	RFC1213-MIB	rfc1213_MIBII.txt
14	**ospf**	**RFC1253-MIB**	**rfc1253-OSPF.txt**
15	**bgp**	**BGP4-MIB**	**rfc1657v1.mib.txt**
17	**dot1dBridge**	**BRIDGE-MIB**	**rfc4188-bridgeMIBNew.txt**
31	**ifMIB**	**IF-MIB**	**rfc2863.txt**
35	etherMIB	EtherLike-MIB	IETF_ETHERLIKE.mib.txt
37	atmMIB	ATM-MIB-v1	ATM-MIB-V1SMI.mib.txt

Table view for: ifTable

Index	ifIndex	ifDescr	ifType	ifMtu	ifSpeed	ifPhysAddress	ifAdmin:
1	1	100GigabitEthernet1/1	6	1548	4294967295	0xCC 0x4E 0x24 0x99 0xD1 0x00	1
2	2	100GigabitEthernet1/2	6	1548	4294967295	0xCC 0x4E 0x24 0x99 0xD1 0x01	1
65	65	GigabitEthernet2/1	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x30	2
66	66	GigabitEthernet2/2	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x31	2
67	67	GigabitEthernet2/3	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x32	2
68	68	GigabitEthernet2/4	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x33	2
69	69	GigabitEthernet2/5	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x34	2
70	70	GigabitEthernet2/6	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x35	2
71	71	GigabitEthernet2/7	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x36	2
72	72	GigabitEthernet2/8	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x37	2
73	73	GigabitEthernet2/9	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x38	2
74	74	GigabitEthernet2/10	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x39	2
75	75	GigabitEthernet2/11	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x3A	2
76	76	GigabitEthernet2/12	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x3B	2
77	77	GigabitEthernet2/13	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x3C	2
78	78	GigabitEthernet2/14	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x3D	2
79	79	GigabitEthernet2/15	6	1548	1000000000	0xCC 0x4E 0x24 0x99 0xD1 0x3E	2

Individual variables such as ifNumber or specific table columns can be seen on the Dump Content tab.


Network Ferret - MIB Analysis : C:\amt_work\Network Ferret\MIBDatabases
File Exchange Favorites Options Help
Legend: Table Name Clash Favorite
☑ Prune Tree □ Match case □ Whole word ☑ In Selection Search
MIB Detail
Dump Content
Table View
Pin Tab
Find:
Find
Find Next
Count All
Highlight All To Find
Copy To Clipboard

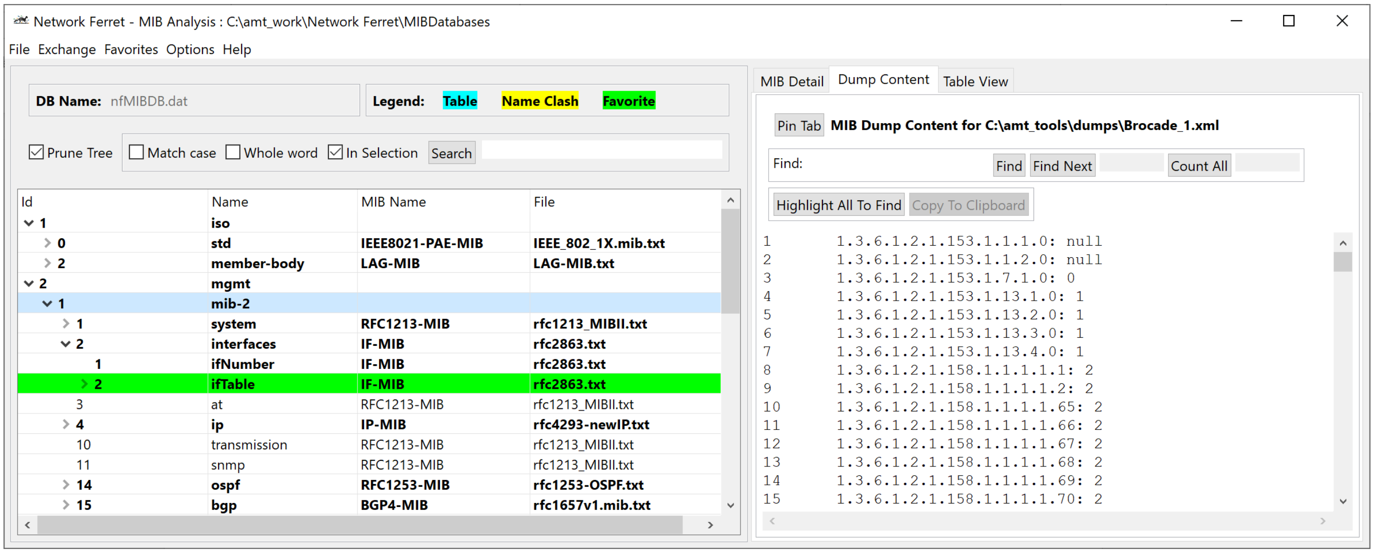
DB Name: nfMIBDB.dat

Id	Name	MIB Name	File
1	mib-2		
1	system	RFC1213-MIB	rfc1213_MIBII.txt
2	interfaces	IF-MIB	rfc2863.txt
1	ifNumber	IF-MIB	rfc2863.txt
2	ifTable	IF-MIB	rfc2863.txt
3	at	RFC1213-MIB	rfc1213_MIBII.txt
4	ip	IP-MIB	rfc4293-newIP.txt
10	transmission	RFC1213-MIB	rfc1213_MIBII.txt
11	snmp	RFC1213-MIB	rfc1213_MIBII.txt
14	ospf	RFC1253-MIB	rfc1253-OSPF.txt
15	bgp	BGP4-MIB	rfc1657v1.mib.txt
17	dot1dBridge	BRIDGE-MIB	rfc4188-bridgeMIBNew.txt
31	ifMIB	IF-MIB	rfc2863.txt
35	etherMIB	EtherLike-MIB	IETF_ETHERLIKE.mib.txt
37	atmMIB	ATM-MIB-v1	ATM-MIB-V1SMI.mib.txt

MIB Dump Content for C:\amt_tools\dumps\Brocade_1.xml

```
1.3.6.1.2.1.2.1.0: 76
```

Dump content for which there is no MIB definition will be attached to the nearest parent branch. For example, MIB2 has some data attached to it for this sample dump. These happen to be the spdMIB (153) and the dot3OamMIB (158)


Network Ferret - MIB Analysis : C:\amt_work\Network Ferret\MIBDatabases
File Exchange Favorites Options Help
Legend: Table Name Clash Favorite
☑ Prune Tree □ Match case □ Whole word ☑ In Selection Search
MIB Detail
Dump Content
Table View
Pin Tab
Find:
Find
Find Next
Count All
Highlight All To Find
Copy To Clipboard

DB Name: nfMIBDB.dat

Id	Name	MIB Name	File
1	iso		
0	std	IEEE8021-PAE-MIB	IEEE_802_1X.mib.txt
2	member-body	LAG-MIB	LAG-MIB.txt
2	mgmt		
1	mib-2		
1	system	RFC1213-MIB	rfc1213_MIBII.txt
2	interfaces	IF-MIB	rfc2863.txt
1	ifNumber	IF-MIB	rfc2863.txt
2	ifTable	IF-MIB	rfc2863.txt
3	at	RFC1213-MIB	rfc1213_MIBII.txt
4	ip	IP-MIB	rfc4293-newIP.txt
10	transmission	RFC1213-MIB	rfc1213_MIBII.txt
11	snmp	RFC1213-MIB	rfc1213_MIBII.txt
14	ospf	RFC1253-MIB	rfc1253-OSPF.txt
15	bgp	BGP4-MIB	rfc1657v1.mib.txt

MIB Dump Content for C:\amt_tools\dumps\Brocade_1.xml

```
1.3.6.1.2.1.153.1.1.1.0: null
1.3.6.1.2.1.153.1.1.2.0: null
1.3.6.1.2.1.153.1.7.1.0: 0
1.3.6.1.2.1.153.1.13.1.0: 1
1.3.6.1.2.1.153.1.13.2.0: 1
1.3.6.1.2.1.153.1.13.3.0: 1
1.3.6.1.2.1.153.1.13.4.0: 1
1.3.6.1.2.1.158.1.1.1.1.1: 2
1.3.6.1.2.1.158.1.1.1.1.2: 2
1.3.6.1.2.1.158.1.1.1.1.65: 2
1.3.6.1.2.1.158.1.1.1.1.66: 2
1.3.6.1.2.1.158.1.1.1.1.67: 2
1.3.6.1.2.1.158.1.1.1.1.68: 2
1.3.6.1.2.1.158.1.1.1.1.69: 2
1.3.6.1.2.1.158.1.1.1.1.70: 2
```

On the main window Options menu, there is an option to set the maximum characters per table cell. Normally this is not a problem. However, there are cells that are super long such as VLAN membership bit maps. Scrolling gets cumbersome. Setting a maximum number of characters per cell alleviates this.

3: Importing/Exporting

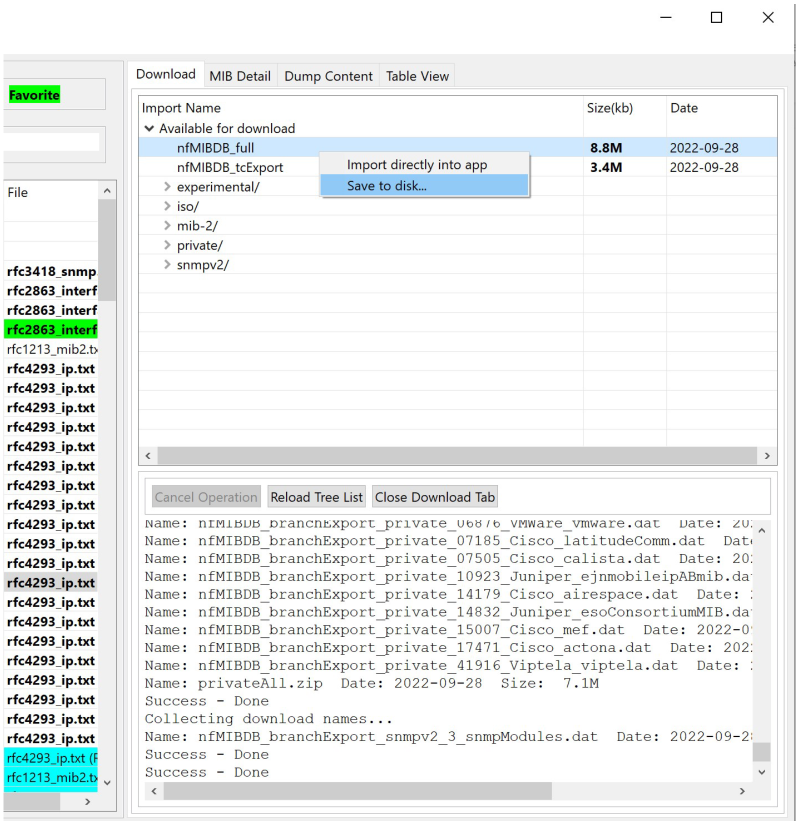
Importing

Download Imports From AMT

AMT has scanned a number of MIBs which are available for import. We will be happy to add more to the site. Send MIB files or a link to support@altmantech.com.

From the main window menu select **Exchange->Import From Download Site**. The Download tab will open in the right pane and it will automatically download the list of possible imports from the AMT site.

At the first level will be the full database. **nfMIBDB_full.** Select the file, right-click, select **Save To Disk** and then choose a directory. After download is complete, **unzip** the file and then select **File->Load Database**.



Import Name	Size(kb)	Date
Available for download		
nfMIBDB_full	**8.8M**	2022-09-28
nfMIBDB_tcExport	**3.4M**	2022-09-28
experimental/		
iso/		
mib-2/		
private/		
snmpv2/		

```
Name: nfMIBDB_branchExport_private_06876_VMWare_vmware.dat  Date: 20
Name: nfMIBDB_branchExport_private_07185_Cisco_latitudeComm.dat  Date
Name: nfMIBDB_branchExport_private_07505_Cisco_calista.dat  Date: 20
Name: nfMIBDB_branchExport_private_10923_Juniper_ejnmobileipABmib.da
Name: nfMIBDB_branchExport_private_14179_Cisco_airespace.dat  Date:
Name: nfMIBDB_branchExport_private_14832_Juniper_esoConsortiumMIB.da
Name: nfMIBDB_branchExport_private_15007_Cisco_mef.dat  Date: 2022-0
Name: nfMIBDB_branchExport_private_17471_Cisco_actona.dat  Date: 202
Name: nfMIBDB_branchExport_private_41916_Viptela_viptela.dat  Date:
Name: privateAll.zip  Date: 2022-09-28  Size:  7.1M
Success - Done
Collecting download names...
Name: nfMIBDB_branchExport_snmpv2_3_snmpModules.dat  Date: 2022-09-2
Success - Done
Success - Done
```

Any other downloads can either be **saved to disk** or **imported directly** into the loaded database (which could be a new, empty database). Imports will replace existing branches. No attempt is made to merge the data.

AMT has created some collections of downloads. The big main branches will have files at the top of their list called branchAll. This is a collection of all branches in that folder.

There are some special collections for vendors. Cisco, for example. There is an import called **00000_Cisco_cisco** which is ONLY the Cisco branch. However, the Cisco MIB package download includes a number of other private branches. They are all included in the file **Cisco_allVendors**.

Import From File System

Under the main window Exchange menu there are options to import TCs and branches from the file system. Branch imports can be either individual branches or zip files containing a collection of branches.

Any import will replace an existing branch if it exists. No attempt is made to merge data.

TC imports will ALWAYS be merged.

Exporting

Export The Main Branches

From the main window Exchange menu there are options to export the main branches or all of the main branches. A file will be created for each sub-branch. For example, under Private, there will be a file for Cisco, Juniper, etc. The files are all created in the same directory.

No zip files are exported. Collections need to be created manually.

Export A Single Branch

From the right-click menu in the MIB Tree, an individual branch can be exported.

Only the first level below the main branches can be selected.

Export A Single Branch Plus TCs

From the right-click menu in the MIB Tree, an individual branch plus Textual Conventions can be exported.

Use this when you want to merge a new vendor branch into an existing MIB database.

1 – Select File->New Database
2 – Scan the MIBs you want to scan.
3 – Highlight the vendor branch under Private, right-click and select Export Branch + TCs.

This will export the MIB definitions AND the Textual Conventions as a single file. Import this file into an existing database. Save the database.

The alternative is to export the MIB Branch as one file and export the TCs as another file and then import them both.

Most of the exports on the download site were done WITHOUT including TCs. Instead, there is one large TC import file with all of them.

4: Scanning MIB Definitions

Databases

MA databases can be kept anywhere. They do not need to be in the directory that was scanned.

The database does maintain file names so if the database is moved to a location that does not have access to the files (like the sample database that can be downloaded), the user will not be able to view the MIB definition files.

Deleting A Branch

MA allows the deletion of branches. This allows the cleaning up of a database of things you don't want to see. Keep in mind, that this cleanup will have to be repeated if the database is rescanned. It might be better to either remove those MIB files or put entries into the nfMIBDB_scanIgnore.txt file.

Scan A Directory

Select **File->Scan A Directory**, choose the directory to be scanned and the Scanner Window will open.

Scanning will recursively traverse the directory hierarchy. At AMT, we download zip packages from vendors that contain files other than MIB definition files. The file nfMIBDB_scanIgnore.txt in the distribution, lists files extensions that are ignored. You can add more.

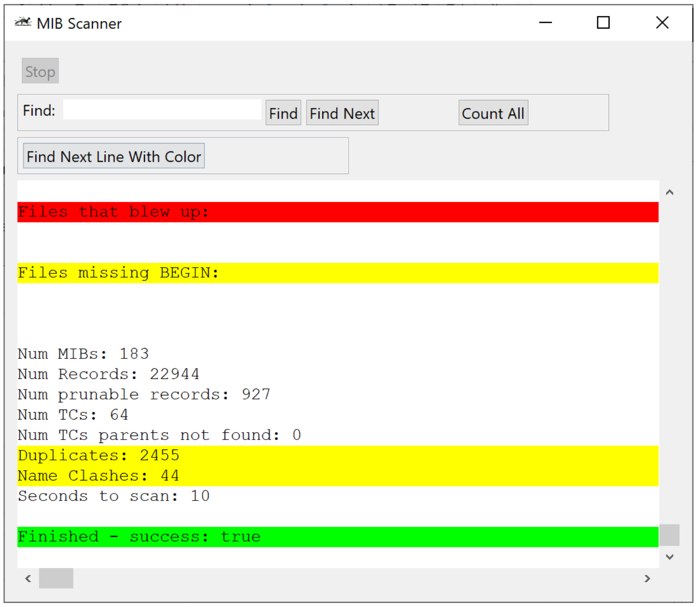
Any directory ending with "_NFDS" (Network Ferret Don't Scan) will be ignored.

Any file larger than 5,000,000 bytes will be ignored. We have seen a couple of valid files just over 2MB. 5MB seems like a safe limit. This is to prevent accidentally loading some massive text file that isn't really a MIB definition.

Any file that has binary data at the start will be ignored.

We have noticed that the painting of the window can take significant CPU so if scanning a massive hierarchy (thousands of files), minimize the window and check periodically to see if it is finished.

A finished scan will look like the graphic below. Hopefully with zero files that blew up, zero files missing BEGIN, no duplicates, no name clashes and a Green – success line.



```
Files that blew up:


Files missing BEGIN:



Num MIBs: 183
Num Records: 22944
Num prunable records: 927
Num TCs: 64
Num TCs parents not found: 0
Duplicates: 2455
Name Clashes: 44
Seconds to scan: 10

Finished - success: true
```

Exceptions

If a file blows up, that is AMT's fault. Please send us the file so we can correct the error.

Vendors include many text files that are not MIB definitions. **Files Missing BEGIN** is just to let you know they are there. This could also be a parsing error with MIB Analysis.

TCs without parents indicates either a missing MIB definition file with the parent TC or some parsing error with MIB Analysis.

Duplicates often occur because multiple versions of a vendor MIB set are downloaded. Also, older MIBs tended to redefine many of the standard variables higher up the tree.

Name Clashes occur when the same OID is found in different MIBs with different names. This is usually a typo, or one vendor bought another and they edited the private branch name, or the vendor changed a top level name and did not change it in all files.

Create a New Database

Scanning a directory automatically creates a new database. The only time to manually create a new database is if you want to build a database from imports.

Select **File->New DB**.

Import .dat and .zip files. See Chapter 3 – Importing/Exporting.

Adding to an Existing Database

The only way to add to an existing database is via imports. Scanning automatically creates a new database.

5: Analyzing A Dump

The **Quick Start** chapter describes how to analyze a dump file. In this chapter we will describe some details about the dump files.

Text Dumps

MIBAnalysis has not been tested with every MIB dump format out there.

It looks for the following:

“OID=” is snmpwalk format. Those four characters are stripped from the line.

Does the line start with a dot (.) or digit?

Grab the OID by reading dots and digits.

For non-table data, the line is simply taken and displayed as is in the Dump Content tab.

For table data, MA looks for the first non-digit, non-dot (.) character after the OID. Then it skips any white space. Then it takes the rest of the line as the value for that OID.

Remember that this is strictly for human consumption so there is no attempt to translate/interpret the data.

XML Dumps

MA expects to find "<?xml" on the first line of an XML file. MA will load the XML and build the object tree.

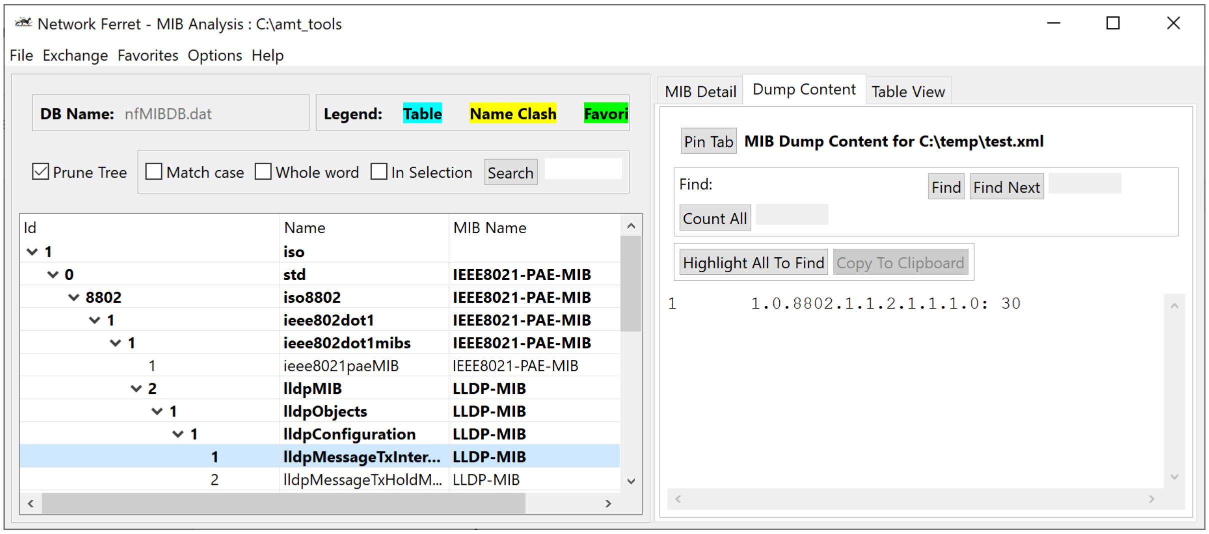
AMT works with ireasoning as our MIB simulator so MA expects that format (mostly) for the file. If there are other XML formats for MIB dumps, send us a sample. support@altmantech.com.

MA expects the MIB instance data to be at the third level in the object tree.

```
<Level 1 object – name irrelevant>
     <Level 2 object – name irrelevant>
          <Level 3 object – name irrelevant oid=”1.2.3”>
               <Value><![CDATA[someData]]></Value>
          </Level 3 object>
          The rest of the file is a list of Level 3 objects
```

For example (paste this into a text file and Analyze it):

```
<?xml version="1.0"?>
<SnmpSimulatorData>
<Instances readCommunity="public">

<Instance oid=".1.0.8802.1.1.2.1.1.1.0" valueType="Integer">
          <Value><![CDATA[30]]></Value>
</Instance>

</Instances>
</SnmpSimulatorData>
```



DB Name: nfMIBDB.dat

Legend: Table Name Clash Favori

☑ Prune Tree ☐ Match case ☐ Whole word ☐ In Selection

Id	Name	MIB Name
∨ 1	**iso**	
∨ 0	**std**	**IEEE8021-PAE-MIB**
∨ 8802	**iso8802**	**IEEE8021-PAE-MIB**
∨ 1	**ieee802dot1**	**IEEE8021-PAE-MIB**
∨ 1	**ieee802dot1mibs**	**IEEE8021-PAE-MIB**
1	ieee8021paeMIB	IEEE8021-PAE-MIB
∨ 2	**lldpMIB**	**LLDP-MIB**
∨ 1	**lldpObjects**	**LLDP-MIB**
∨ 1	**lldpConfiguration**	**LLDP-MIB**
1	**lldpMessageTxInter...**	**LLDP-MIB**
2	lldpMessageTxHoldM...	LLDP-MIB

MIB Dump Content for C:\temp\test.xml

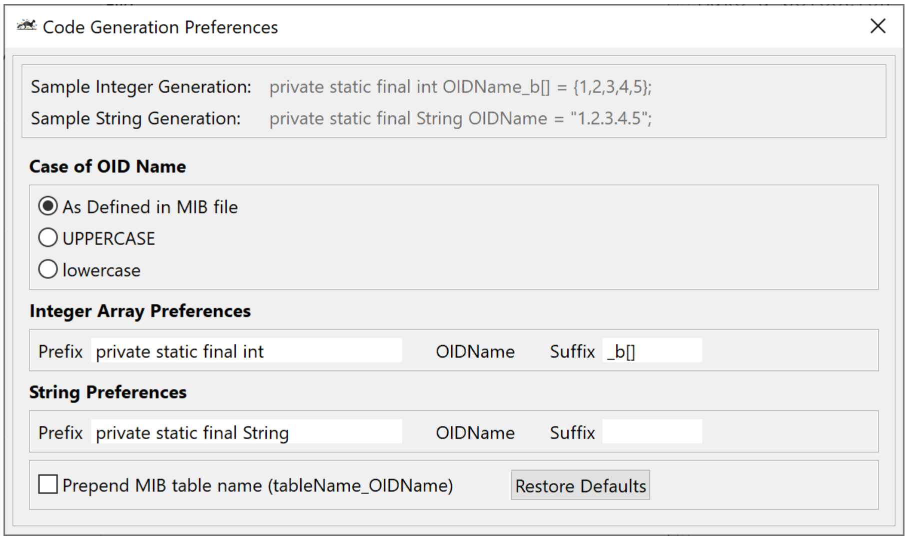
```
1       1.0.8802.1.1.2.1.1.1.0: 30
```

6: Generating Code

We are programmers here at AMT so we added a little feature to generate OID variable constants.

Options

From the main window menu select **Options->Code Generation Options**.



Code Generation Preferences

Sample Integer Generation: private static final int OIDName_b[] = {1,2,3,4,5};

Sample String Generation: private static final String OIDName = "1.2.3.4.5";

Case of OID Name

● As Defined in MIB file
○ UPPERCASE
○ lowercase

Integer Array Preferences

Prefix: private static final int — OIDName — Suffix: _b[]

String Preferences

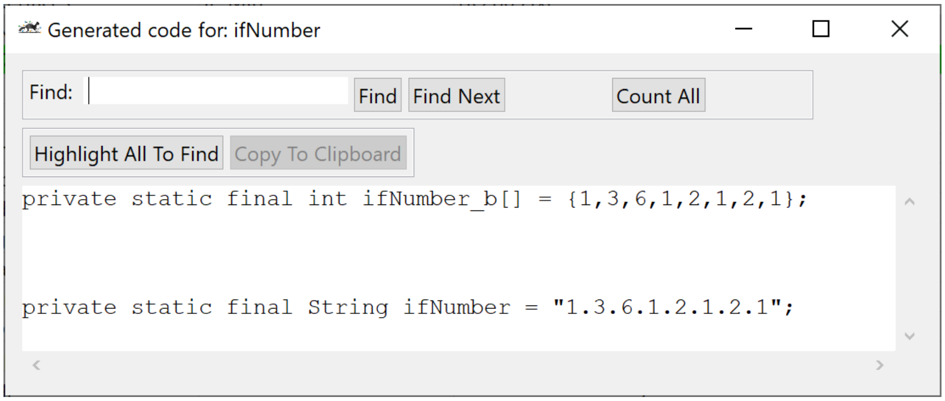
Prefix: private static final String — OIDName — Suffix:

□ Prepend MIB table name (tableName_OIDName)

The Samples at the top of the dialog will update as changes are made in the dialog. Values are stored in the Java VM Preferences cache so they will be there the next time MA starts.

Sample Code

Right-click on a variable or table and select **Generate Code**.



Generated code for: ifNumber

```
private static final int ifNumber_b[] = {1,3,6,1,2,1,2,1};


private static final String ifNumber = "1.3.6.1.2.1.2.1";
```

7: Favorites

Favorites work like browser favorites. Right-click on a row in the MIB tree and select Add or Remove from favorites.

When a favorite is selected from the Favorites menu in the main window menu, the tree will highlight the favorite if it exists in the tree.

Favorites can be imported and exported from the main window menu.