Network SNMP Connector

SNMP supported devices

- Name: **SNMP**
- Version: 2.6.0.0
- Protocol: SNMP
- Interface: TCP/IP or UDP
- Runtime: .NET 2.0 (Multiplatform)
- Configuration:
    - Devices / Protocols

Supported Operands
Nodes Configuration
Points Configuration
Receiving Traps and Informs
Driver Revision History

Overview

Communication Driver Information	
Driver name	SNMP
Assembly Name	T.ProtocolDriver.SNMP
Assembly Version	2.6.0.0
Available for Linux	True
Devices supported	SNMP supported devices
Manufacturer	N/A (Not Applicable)
Protocol	SNMP
Interface	TCP/IP or UDP
PC Requirements	Ethernet port

Supported Operands

Get Command

Command	Description
OctetString	FS Device DataType ASCII
ObjectIdentifier	FS Device DataType ASCII
TimeTicks	FS Device DataType ASCII
Integer32	FS Device DataType Long
Gauge32	FS Device DataType DWord
Counter32	FS Device DataType DWord

Set Command

Command	Description
OctetString	FS Device DataType ASCII
Integer32	FS Device DataType Long
Counter32	FS Device DataType DWord

Nodes Configuration

Station Configuration

- ***<IP address>***: IP address of the equipment in the network
- ***<Port>***: Port of the equipment in the network
- ***<MIB>***: MIB file of equipment. If the MIB file is not specified the SNMPv2-MIB file will be used. From FrameworX 10.1.6a, a MIB file chosen in the Designer on Windows is also found when the solution runs on Linux, when the file is in the product's Mibs folder.
- ***<Version>***: Version of the SNMP
- ***<Authentication>***: Authentication MD5 or SHA-1. Only for Version 3.
- ***<Passphrase>***: Passphrase when Authentication is enabled.
- ***<Privacy>***: Privacy DES or AES. Only for Version 3 and Authentication is enabled.
- ***<Passphrase>***: Passphrase when Privacy is enabled.
- ***<UserName>***: UserName for version 3.

Points Configuration

The syntax for the SNMP communication points is*: **[Community name]; <OID>***

Where:

- ***[Community name]*** can be:
    - Public (default)
    - Private
- ***<OID>***: indicates the Object Identifier (OID) in the verbose notation. You can use the TreeView to select the OID from MIB.

E.g.: **.iso.org.dod.internet.mgmt.mib-2.transmission**

Receiving Traps and Informs

Besides reading values on a schedule, the connector can receive the traps and informs that network devices send on their own. Receiving is off by default: channels without a TrapPort work as before.

Channel Configuration

- ***<TrapPort>***: UDP port that receives the traps and informs of the channel's nodes. 162 is the standard SNMP trap port. 0 (default) turns receiving off.

- The sender is recognized by the **IP address** of the node. A node with IP address 0.0.0.0 accepts notifications from any sender.
- Versions 1 and 2c are accepted when the **Community** matches the node. Version 3 is accepted when the **UserName** matches the node and the message carries at least the Authentication and Privacy configured on the node.
- Version 2c and version 3 informs are confirmed back to the device.
- Only one program per computer can listen on a UDP port. On Windows, the SNMP Trap service may already use port 162: stop it or choose another port. On Linux, a port below 1024 needs root or the CAP_NET_BIND_SERVICE capability. Notifications are received over IPv4.

Points Configuration

Start the point address with **trap:** to read the latest notification received from the node. Points with this prefix are read only. In the address editor, check *Trap* to select an object from the MIB or one of the entries below.

Address	Value
trap:<OID>	Value of that variable in the latest notification. The OID can be numeric (e.g. trap:1.3.6.1.4.1.99999.1) or in the verbose notation, as in the polled points.
trap: @TrapOID	OID that identifies the notification.
trap: @Source	IP address of the sender.
trap:@Time	Time the notification was received (UTC).
trap:@Count	Number of notifications received from the node since the runtime started.

When several notifications arrive between two scans, the point keeps the latest value and **trap:@Count** counts all of them. While the port cannot be opened, the trap: points have Bad quality and the polled points of the same node keep working. Editing the channel while the solution runs restarts it, and the count starts again from zero.

Driver Revision History

SNMP Revision History	
Version	**Notes**
2.5.0.7	Initial release on new documentation standards.
2.6.0.0	Receives SNMP traps and informs: new channel option TrapPort and point addresses with the trap: prefix. Channels without TrapPort behave as before.

In this section...