

# **ndCONF Agent Development Studio**

## **User Guide**

for Linux installations

**NuDesign Technologies Inc.**

**2019**

## Table of Contents

1	Introduction.....	8
2	Installation Files.....	10
2.1	<SDK installdir>/ndconf.....	11
2.2	<SDK installdir>/ndconf/CodeGenerator.....	11
2.2.1	<SDK installdir>/ndconf/CodeGenerator/ndt-template.....	11
2.3	<SDK installdir>/ndconf/doc.....	11
2.4	<SDK installdir>/ndconf/mib2xml.....	12
2.4.1	<SDK installdir>/ndconf/mib2xml/MIB.....	12
2.4.2	<SDK installdir>/ndconf/mib2xml/out.....	12
2.5	<SDK installdir>/ndconf/ncclient.....	12
2.5.1	<SDK installdir>/ndconf/ncclient/config.....	12
2.5.2	<SDK installdir>/ndconf/ncclient/config/edit-config.....	12
2.5.3	<SDK installdir>/ndconf/ncclient/config/rpc.....	14
2.6	<SDK installdir>/ndconf/ndGarageMib.....	14
2.6.1	<SDK installdir>/ndconf/ndGarageMib/Linux.....	14
2.7	<SDK installdir>/ndconf/rcvlan.....	14
2.7.1	<SDK installdir>/ndconf/rcvlan/Linux.....	15
2.8	<SDK installdir>/ndconf/testagent.....	15
2.8.1	<SDK installdir>/ndconf/testagent/config.....	15
2.8.2	<SDK installdir>/ndconf/testagent/schema.....	15
2.9	<SDK installdir>/ndconf/sshd.....	15
2.10	<SDK installdir>/ndconf.....	16
2.10.1	<SDK installdir>/ndconf/include.....	16
2.10.2	<SDK installdir>/ndconf/lib64.....	16
3	Installation.....	17
3.1	SDK Requirements.....	17
3.2	Supported Distributions.....	17
3.3	Notes.....	17
4	Working with the included Source Projects.....	20
4.1	Building Projects.....	20
4.2	Configuring testagent for the Produced Shared Objects Libraries.....	21
4.3	Startup Data.....	22
5	Working with testagent.....	23
5.1	testagent Module Compliance.....	23
5.2	testagent Command Line Options.....	24
5.3	Using testagent.....	25
6	Working with the CLI.....	26
6.1	ndGarageMib Example.....	26
6.2	rcvlan Example.....	28
7	Working with ncclient.....	31

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

7.1	ndGarageMib Example.....	31
7.2	rcvlan based example.....	33
7.3	Notifications in ncclient.....	35
8	Working with SNMP.....	39
8.1	Notifications in MIBrowser.....	40
8.2	Putting it All together.....	41
9	CLI.....	44
9.1	Command completion.....	44
9.2	Commands and parameters.....	45
9.2.1	? or <prefix>?.....	45
9.2.2	about.....	45
9.2.3	cls.....	45
9.2.4	cmprfg.....	45
9.2.5	configure.....	45
9.2.6	datadll [load unload] <name>.....	46
9.2.7	exit.....	46
9.2.8	help.....	47
9.2.9	quit.....	47
9.2.10	script.....	47
9.2.11	session.....	47
9.2.12	show.....	47
9.2.13	sleep.....	48
9.3	Configuration Mode.....	49
9.3.1	commit.....	50
9.3.2	create.....	50
9.3.3	delete.....	50
9.3.4	diff.....	51
9.3.5	edit.....	51
9.3.6	exit.....	51
9.3.7	load.....	52
9.3.8	quit.....	52
9.3.9	run.....	52
9.3.10	save.....	53
9.3.11	set.....	53
9.3.12	show.....	53
9.3.13	top.....	54
9.3.14	up.....	54
10	ncclient - NetConf client.....	55
10.1	Commands and parameters.....	55
10.1.1	? or <prefix>?.....	55
10.1.2	about.....	56
10.1.3	cls.....	56
10.1.4	connect <host> [username].....	56

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

10.1.5	cancel-commit.....	57
10.1.6	close-session.....	58
10.1.7	create-subscription [startTime YYYY-MM-DDThh:mm:ssZ].....	58
10.1.8	commit.....	59
10.1.9	copy-config.....	59
10.1.10	delete-config.....	59
10.1.11	discard-changes.....	59
10.1.12	disconnect.....	59
10.1.13	edit-config.....	60
10.1.14	exec.....	61
10.1.15	exit.....	61
10.1.16	get.....	61
10.1.17	get-config.....	62
10.1.18	get-schema.....	62
10.1.19	help.....	63
10.1.20	hierarchy.....	63
10.1.21	kill-session.....	63
10.1.22	list.....	63
10.1.23	lock.....	64
10.1.24	partial-lock.....	64
10.1.25	partial-unlock.....	64
10.1.26	quit.....	64
10.1.27	rpc.....	64
10.1.28	script.....	65
10.1.29	show.....	65
10.1.30	unlock.....	65
10.1.31	validate.....	66
11	NACM - Access Control.....	67
11.1	Incoming RPC Message Validation.....	68
11.2	Outgoing <notification> Authorization.....	68
11.3	Data Node Access Validation.....	70
11.4	Debugging Access Control.....	72
12	SNMP.....	73
13	Code Generation.....	74
13.1	Code Generator.....	74
13.2	ndt.py Plugin.....	74
13.2.1	'ndt.py' Plugin Command Line Options.....	74
13.3	Generating Code with ndt.py.....	75
13.4	ndtGen.....	75
13.4.1	ndtGen Command Line Options.....	75
13.5	Generating Code with ndtGen.....	76
13.6	The Generated Project.....	76
13.7	Building the Project.....	77

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

13.8	Configuring testagent to use the Project's shared object output.....	77
14	Other Resources.....	78
14.1	Project Design Notes.....	78
14.2	Class Reference.....	78
14.3	man pages.....	78
15	Products for ndCONF Development.....	79
15.1	ndCONF Builder.....	79
15.2	ndCONF SDK.....	79
15.3	NETCONF, CLI, RESTCONF and SNMP Access Agents.....	80
16	About NuDesign.....	81
17	Appendix.....	82
17.1	'ND-GARAGE-MIB' Yang module.....	82
17.2	'ND-GARAGE-MIB' MIB module.....	86
17.3	'RAPID-CITY-MIB' (rcvlan) Yang Module.....	90
17.4	'RAPID-CITY-MIB' (rcvlan) MIB Module.....	97
17.5	Capabilities.....	118

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## DOCUMENT HISTORY

Date	Revision	Remarks
2017-12-10	.9	Imported from v1.12 Yang DEMO Quick Start document.
2018-01-03	1.0	Update and add SDK specific content.
2018-01-15	1.1	Add appendix sections for RAPID-CITY YANG and MIB modules. Miscellaneous edits
2018-01-23	1.2	Add 'Other Resources' and 'Startup Data' sections. Other miscellaneous updates.
2018-01-31	1.3	- change references to /usr/local/NuDesign/ to the SDK install directory. - change references to 'NcClient' and 'YANGDemoAgent' - add sub sections for <b>CLI</b> ndGarageMib and rcvlan examples. - add sub sections for <b>ncclient</b> ndGarageMib and rcvlan examples. - text regarding the derivation of <b>rcvlan.yang</b> and the related project.
2018-02-01	1.4	- minor edits
2018-02-09	1.5	- documentation changes relating to renaming and redefining to ndtCONF - revert lib64 and include references to fixed install location.
2018-02-14	1.6	- change footer. - add documentation about included design reference. - add documentation about included design notes for the two datadll projects.
2018-02-23	1.7	Add section to describe NuDesign Products for customizing ndCONF. - rename references of ngsettings.xml to ndconf.xml - rename reference of ngsettings_cli .xml to ndconf_cli.xml - miscellaneous updates
2018-02-28	1.8	- change image of directory structure. - correct section 2 sub headings. - minor revisions.
2018-08-01	1.9	- added - confirmed commit - writable running - partial locking
2018-08-20	2.0	- content updates
2018-09-06	2.1	- update partial-lock, partial-unlock
2018-09-18	2.2	- correct section numbering

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

2019-02-01	2.3	<ul style="list-style-type: none"><li>- change paths for updated layout of the SDK</li><li>- add CodeGen section for non-eval SDK</li><li>- restructure some of the headings.</li><li>- edits and corrections.</li><li>- update copyright notices</li></ul>
2019-02-01	2.4	<ul style="list-style-type: none"><li>- change demo-data-cfg.xml to project specific name</li><li>- several minor edits.</li></ul>

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 1 Introduction

This document describes the NuDesign ndCONF product. This document describes a complete management agent development product containing an extensible **YANG Datastore Server, NETCONF / YANG Datastore SDK** with include and library files, API documentation to the YANG Datastore object providers and to the protocol **Access Agents**. It also includes **source code** samples for demonstrating dynamically loadable Datastore data extensions. The product also contains a basic **CLI Access** agent, for operational and configuration aspects of the system. Multiple, dynamically loadable data extensions **are** supported by the **testagent**. The evaluation version is available for download from the NuDesign site at [www.ndt-inc.com](http://www.ndt-inc.com), as a Linux executable, installation package.

The evaluation version includes, a time limited **testagent** and includes an implementation for the standard YANG-module `ietf-interfaces`, plus two NDT code generated C++ source code projects<sup>1</sup>. The first is for a data model defined from the "ND-GARAGE-MIB" YANG module and the other one defined in the "**rcvlan**" YANG module. Both YANG documents were created from MIB modules, using NuDesign's Visual MIBuilder. These are the ND-GARAGE-MIB and (a) simplified RAPID-CITY-MIB (rcvlan) SNMP MIB Modules. In the case of the rcvlan example, the default MIB derived code has been modified to demonstrate how to transform the code into a more CLI centric data model. All modules are listed in the document Appendix.

Access to the agent is provided via an "Access Agent" (AA). An 'AA' is a Dynamic-Link Library loaded by the **testagent** at startup. Each "handles" a specific protocol (i.e. NetConf/SNMP/CLI/Web...). Protocol operations that manipulate data (read/write) are handled commonly by the data store in the **testagent**. In other words, all access to data is common, irrespective of the protocol used to access it. The evaluation version includes two additional access agents, one each for NETCONF and SNMP.

The **testagent** is a console application; the AA CLI provides command line handling for it.

**Note:** before you start the **testagent**, you need to stop any SNMP agents you may have running on your computer, as there may be a **UDP** port conflict with the **testagent**, due to the provided SNMP access.

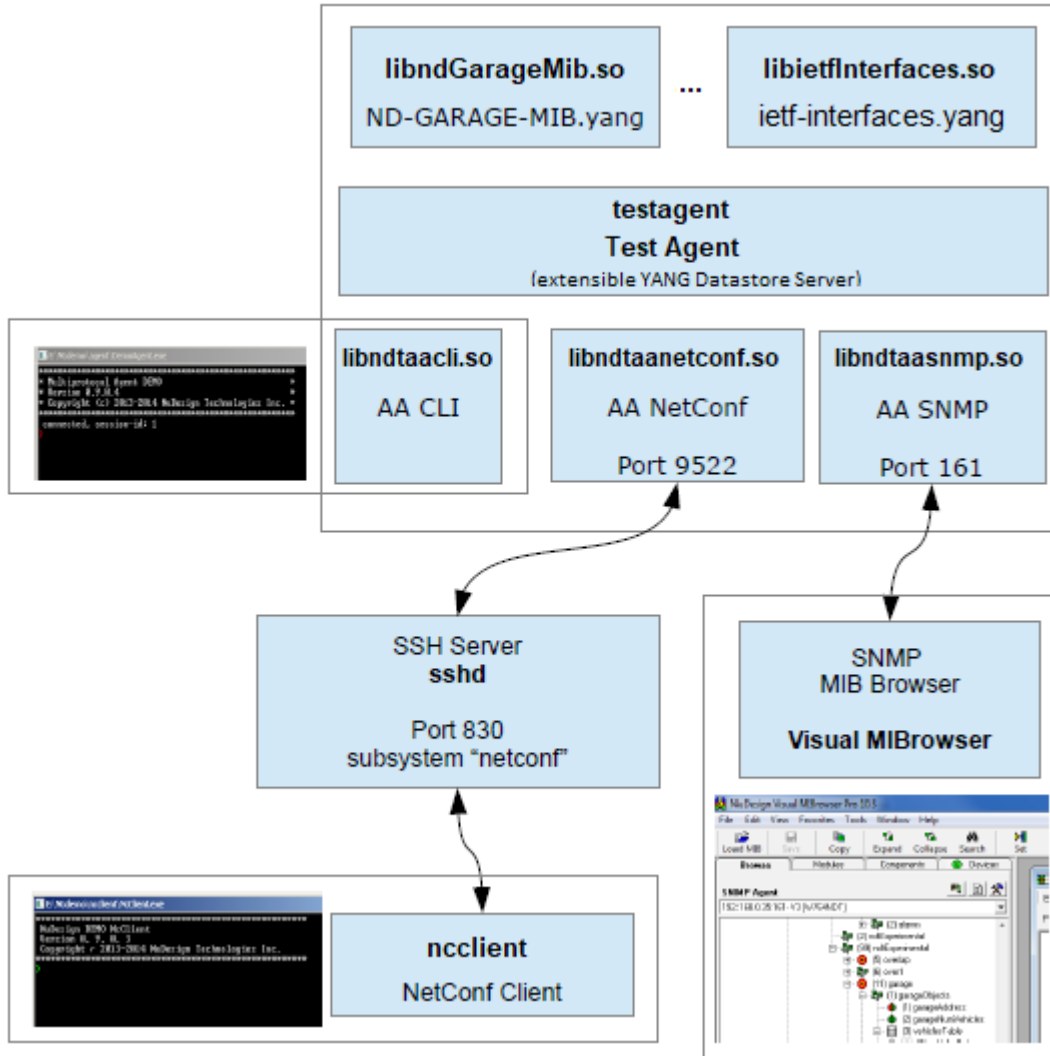
**ncclient** is a simple NetConf client console application; it may be used to access the **testagent** using the NETCONF protocol over **SSH**. NuDesign's **Visual MIBrowser Pro** is the SNMP client used within this document to present SNMP operations, though any SNMP manager could be used.

**Also Note:** in the diagram that follows, two data models are depicted as integrated with the **testagent**. These are provided for by '**libndGarageMib.so**' and '**libietfInterfaces.so.0**'. Another, **rcvlan**, is also provided and will be discussed later in the document.

**Lastly**, the **evaluation** version of the product expires **30 days** after installation. It also does not include any of the code generation facility available in the licensed version.

1 See ndCONF Builder

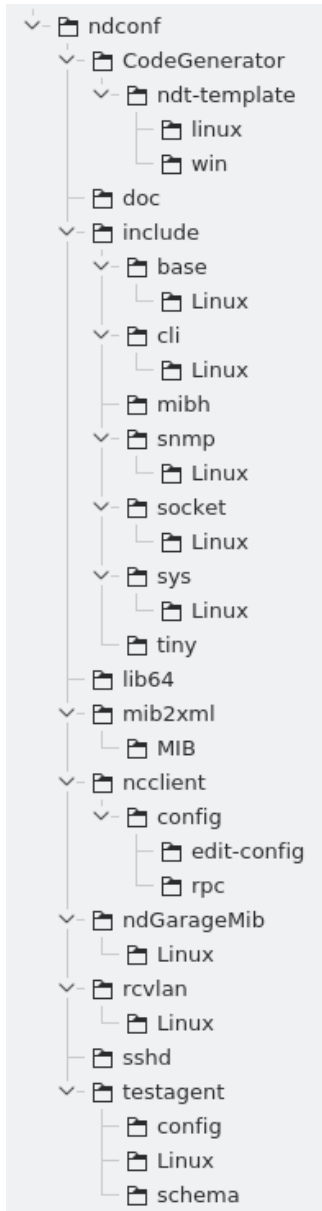




This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 2 Installation Files

By default, the installer will place **ndconf** under the directory **‘/opt/ndConf-x.xx’** (and **‘/opt/ndConfEval-x.xx’** for the evaluation version). **x.xx** is the version number being installed. As a convention, that location is identified below by **‘<SDK installdir>’**<sup>2</sup>.



<sup>2</sup> The **CodeGenerator** directory and files below exist only in the release version of the SDK.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 2.1 <SDK installdir>/ndconf

uninstall	Uninstall executable
install.log	A log of operations performed by the post installation script.

## 2.2 <SDK installdir>/ndconf/CodeGenerator

The directories below this point contain files relating to code generation.

ndt.py	pyang plugin for code generation.
pyang-1.7.5.tar.gz	pyang version 1.7.5 compatible with ndt.py.

### 2.2.1 <SDK installdir>/ndconf/CodeGenerator/ndt-template

The directories below this point contain platform specific code generation template files.

linux	Linux specific template files.
win	Windows specific template files.

## 2.3 <SDK installdir>/ndconf/doc

Errors.htm	HTML page documenting system error interpretation.
Main-pdf.html	Simple HTML front end to class library PDFs.
ndCONF Builder – CodeGen Tech Reference.pdf	Design notes for datadlls, based strictly on generated code and specifically the <b>ndGarageMIB</b> project.
ndCONF Builder – UsingExistingImpl Tech Reference.pdf	Design notes for datadlls where there is pre-existing implementation code and specifically the <b>rcvlan</b> project.
ndtbase.pdf	PDF document providing design documentation for the <b>ndtbase</b> class
ndtcli.pdf	PDF document providing design documentation for the <b>ndtcli</b> class
ndtmibh.pdf	PDF document providing design documentation for the <b>ndtmibh</b> class
ndtsnmp.pdf	PDF document providing design documentation for the <b>ndtsnmp</b> class
ndtsocket.pdf	PDF document providing design documentation for the <b>ndtsocket</b> class
ndtsys.pdf	PDF document providing design documentation for the <b>ndtsys</b> class
NuDesign ndCONF User Guide.PDF	This document.
NuDesign ndCONF Studio Eval License Agreement.pdf	Evaluation licensing information. This file exists only when the installation is for evaluations.
NuDesign ndCONF Studio License Reference.pdf	Licensing information.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

ReadMe – ndCONF SDK.txt	Text file containing SDK specific release information.
ReadMe – ndCONF Builder.txt	Text file containing Builder specific release information.

## 2.4 <SDK installdir>/ndconf/mib2xml

mib2xml	Executable to convert a MIB into a MIB info document
mib2xml.cfg	Configuration file for the mib2xml executable.

### 2.4.1 <SDK installdir>/ndconf/mib2xml/MIB

*.mib	MIB files
-------	-----------

### 2.4.2 <SDK installdir>/ndconf/mib2xml/out

*.xml	Output XML files
-------	------------------

## 2.5 <SDK installdir>/ndconf/ncclient

ncclient	NetConf client (command line)
----------	-------------------------------

### 2.5.1 <SDK installdir>/ndconf/ncclient/config

NcClient.xml	configuration file for <b>ncclient</b>
--------------	--

### 2.5.2 <SDK installdir>/ndconf/ncclient/config/edit-config

Repository of files that contain portion of xml encoding of the **edit-config** rpc.  
For example

```
> edit-config candidate 1
```

would generate the edit-config request for target=candidate, and the rest of the xml would be taken from file "1.xml" that resides in ...\\config\\edit-config folder. Content of these files should contain <default-operation>, <test-option>, <error-option> and/or <config> elements for valid edit-config request.

This is an example of the edit-config generated:

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
<rpc message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <candidate />
    </target>
    <default-operation>merge</default-operation>
    <test-option>set</test-option>
    <config>
      <interfaces xmlns:t="http://ndt-inc.com/ns/ndt-interfaces">
        <interface xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" operation="create">
          <name>if6</name>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

Note that the red portion is directly copied from file "1.xml".

### 2.5.3 <SDK installdir>/ndconf/ncclient/config/rpc

**ncclient** implements the 'rpc' command. E.g.

```
> rpc f1
```

This command would search for file "f1.xml" in "rpc" folder, and if found, read it and pass the content of this file to **testagent** for processing. An example of the content:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter type="subtree">
      <garage></garage>
    </filter>
  </get>
</rpc>
```

## 2.6 <SDK installdir>/ndconf/ndGarageMib

ndGarageMib-startup-cfg-demo.xml	Sample initialization data. This data may be copied to the <b>testagent</b> 's startup-cfg.xml file to auto initialize running data for this project. See the section 'Startup Data' below.
m2y_ndGarageMib.xml	MIB-to-YANG mapping file.
mib_ND_GARAGE-MIB.xml	MIB info file (generated by mib2xml tool)
NdGarageMib*.cpp	Project c++ source files.
NdGarageMib*.h	Project source include files.
NDGarageV2.mib	Mib file on which the projects is based.
ND_GARAGE-MIB.yang	YANG file (derived from above) on which the project is based.

### 2.6.1 <SDK installdir>/ndconf/ndGarageMib/Linux

makefile	Makefile for project
StdAfx.h	Platform specific project include file.

## 2.7 <SDK installdir>/ndconf/rcvlan

rcvlan-startup-cfg-demo.xml.xml	Sample initialization data. This data may be copied to the <b>testagent</b> 's startup-cfg.xml file to auto initialize running data for this project. See the section 'Startup Data' below.
m2y_rcvlan.xml	MIB-to-YANG mapping file.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

mib_RCVLAN .xml	MIB info file (generated by mib2xml tool)
rcvlan*.cpp	Project c++ source files.
rcvlan*.h	Project source include files.
rcvlan.mib	Mib file on which the projects is based.
rcvlan.yang	YANG file (derived from RAPID-CITY.YANG) on which the project is based.

### 2.7.1 <SDK installdir>/ndconf/rcvlan/Linux

makefile	Makefile for project
StdAfx.h	Platform specific project include file.

## 2.8 <SDK installdir>/ndconf/testagent

testagent	test agent
Main.cpp	Source file
teststart	Script to start <b>testagent</b> with the default parameters.

### 2.8.1 <SDK installdir>/ndconf/testagent/config

ndconf.xml	configuration file for test agent.
ndconf_cli.xml	configuration file for CLI Access agent
startup-cfg.xml	startup configuration data for the <b>testagent</b> .
m2y_*.xml	MIB-to-YANG mapping files
mib_*.xml	MIB info files (generated by mib2xml tool)

### 2.8.2 <SDK installdir>/ndconf/testagent/schema

Repository of YANG modules on which the included projects are dependent. The [get-schema](#) rpc method also looks here for YANG files.

## 2.9 <SDK installdir>/ndconf/sshd

ndncsub	Pipe process executable.
---------	--------------------------

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 2.10 <SDK installdir>/ndconf

This directory contains the include files and library file for building, the included programs and projects.

### 2.10.1 <SDK installdir>/ndconf/include

This directory contains a hierarchy of include files for building the included projects.

**Note:** The **makefile** for each project expects the includes be in this directory. If the includes are moved for any reason, then the **makefile** for each project will need to be **modified**.

### 2.10.2 <SDK installdir>/ndconf/lib64

This directory contains 64 bit shared object library files for building and executing the included programs.

**Note:** These executables are built expecting the libraries to be in the this directory. If the libraries are moved for any reason, then these libraries need to be available to the executables in this product, specifically **ncclient** and **testagent**.. This can be accomplished a number of ways on Linux, Including the following:

- copy, move or soft link them to one of the conventional library repositories, such as **/usr/lib** or **/usr/lib64**.
- Use **ldconfig**.
- export **LD\_LIBRARY\_PATH** with a value that includes a path to the new location of the **lib64** directory. This may be done from the command line, or in the user's **.bashrc**.



## 3 *Installation*

### 3.1 SDK Requirements

This SDK has a number of requirements for use. These are:

- a GNU C/C++ development environment, version 5.3 or greater, to build the projects.
- Open SSH server
- libssh2
- Openssl

These should all be available from your Linux distribution.

### 3.2 Supported Distributions

This Linux version of YANG Datastore Generator SDK has been verified to build the projects and operate on a variety of recent generation 64bit Linux distributions, including:

- Fedora 23 - 29
- Debian 9.3
- Ubuntu 18.04
- OpenSUSE 42.3, Leap 15.0

### 3.3 Notes

**Note 1:** Some of these notes may be Fedora 23-28 specific.

**Note 2:** most of the following steps require **root** level privilege.

**Note 3:** the server needs to be run as **root** (or **sudo**) to access privileged ports: (SNMP uses UDP port 161 by default, which is privileged on most Linux distributions.)

1. **chmod +x** the install **.bin** file you downloaded to make sure it is still executable after downloading. You will need to have root privileges to execute the **.bin** file to install the SDK. Select a location to install the SDK.
2. Make sure **sshd** is installed on your distribution. How to do this varies with the distribution. Once you have, you need to stop it to modify the configuration. You can use the following (as root):

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
# systemctl stop sshd
```

3. Add entries for 'ndncsub' to **/etc/ssh/sshd\_config** directly below appropriate lines (commented or not)

```
Port 830
Subsystem netconf <SDK install directory>/sshd/ndncsub
```

4. Ensure **sshd** starts at startup (it may be disabled by default, depending on distribution)

**note: systemctl** may not be available on all distributions, if not you can likely skip this step)

```
# systemctl enable sshd
```

5. Enable **sshd** to use port 830 (for netconf). This is probably not necessary on many distributions, but is required on SELinux distributions, such as Fedora & RedHat.

```
# semanage port -a -t ssh_port_t -p tcp 830
```

6. Make sure sshd is restarted with the new configuration items. How depends on the distribution.

```
# systemctl restart sshd
```

or

```
# service sshd restart
```

or

```
# /etc/init.d/ssh restart
```

If all else fails, kill and start '**sshd -D**' manually.

Do a '**netstat -a -t**' to ensure port 830 is open.

7. If the firewall is running, open firewall ports for remote access (may not be necessary on all distributions)
  - A) Install "Firewall configuration tool" (optional)
  - B) Open "Firewall configuration tool" (optional)
  - C) add port to firewall exceptions (make it "permanent", if you want it to power up with these)

```
SNMP      udp    161-162
NETCONF   ssh    tcp 830
```

Alternatively, for testing, you can disable the firewall. This can be done (on Fedora) with

```
# systemctl stop firewalld
```

To have it not start again on startup, you can do:

```
# systemctl disable firewalld
```

Now you should be ready to run the **testagent** and the NETConf Client, **ncclient**.

8. To run the **testagent** using **teststart**. (See the **man** pages for **testagent** and **teststart**.) Change the directory to **<SDK installdir>/ndconf/testagent**, then type

```
# ./teststart
```

This starts **testagent** with the default parameters to run it from this directory.

- 9) run the demo **netconf** client. (See the **man** pages for **ncclient** noting in particular the library requirements.) Change the directory to **<SDK installdir>/ndconf/ncclient**, then type:

```
# ./ncclient
```

## 4 Working with the included Source Projects

This install contains three C++ projects. The first two projects were generated with NuDesign's ndCONF Builder product³. The latter is a simple console, data server application, implementing NetCONF, SNMP and CLI from the resources provided by this installation

<b>ndGarageMib</b>	A YANG data provider library project, implementing the <b>ND-GARAGE-MIB.yang</b> module, which is a direct MIB-to-YANG conversion from the <b>ND-GARAGE-MIB.mib</b> module.
<b>rcvlan</b>	A YANG data provider library project, implementing <b>rcvlan.yang</b> . <b>rcvlan.yang</b> is a subset derived from the <b>RAPID-CITY.yang</b> module, which is a direct MIB-to-YAMG conversion of a subset of the <b>RAPID-CITY</b> enterprise MIB. This subset MIB contains just two related MIB tables.  This project demonstrates the consolidation of those two MIB tables into nested YANG lists.
<b>testagent</b>	The source project for the <b>testagent</b> .

Each is located in the SDK directory of the same name.

### 4.1 Building Projects

To build these projects, change directory to the **Linux** sub-directory in each project and typing make.

There are a number of targets and build options for these projects. Targets are provided on the make command line. Options may be provided via the environment or on the make command line. These are:

**clean** This target indicates that the project should be "cleaned". This deletes all object files and libraries or the executable, depending on the project.

**MODE** This variable selects whether the build is a '**debug**' or '**release**' build. The default value is '**debug**'. To create a release version of the project you would set the value to '**release**'. **E.g.**

³ For the purposes of your evaluation, you may contact us about generating a project for you, from your YANG document.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
# make clean
# make MODE=release
```

The first line “cleans” the project and the second builds a release version of the project.

**NUDESIGNDIR** This variable specifies where the SDK include and library files are located. The default is `<SDK installdir>/ndconf` . If the files under this directory have been move to another directory, then this variable will need to specified in the **makefile**.

## 4.2 Configuring testagent for the Produced Shared Objects Libraries

Two of the source projects produce a “data dll”. These are the

1. **ndGarageMib**
2. **rcvlan**

projects. Both of these produce a “shared object” run-time loadable library. On Linux, the name of the library file will be ‘lib’ + project name + “.so” and while be located in the **Linux** sub-directory of the project. E.g. **libndGarageMib.so**.

To have the **testagent** use them, you need to to make a configuration change to the **testagent**. This change is made in the **ndconf.xml**. This file is located in the **testagent** “**config**” directory. By default this directory is “**./testagent/config**”, though this can be modified by the **testagent** command line options.

Once you find the **ndconf.xml**, open the file in a text editor. Locate the **<datadlls>** section file. Within this section are individual **<datadll></datadll>** entries. In the default file, there will be an existing entry in this section for the ‘if’ (interfaces) datadll, which looks as follows:

```
<datadll name="if" dll="libietfInterfaces.so.0" ></datadll>
```

There are two or three fields in a **<datadll> </datadll>** specification. These provide specific configuration information about a particular **datadll**. These fields are as follows.

names	This is the name that is given to the <b>datadlls</b> . This is used in load, unload & status operations.
dlls	This is the actual file name of the <b>datadlls</b> . E.g. <b>libietfInterfaces.so.0s</b> .
paths	(optional). This is the path to the <b>datadlls</b> . When not specified, the path is assumed to be <b>&lt;SDK installdir&gt;/ndconf/lib64</b> . If the <b>datadlls</b> ’s path is not this, then you need to specify it.

This manual and associated NuDesign Software are provided on an “As Is” basis. By using them, you are consenting to the Terms of the “NuDesign Software License Agreement” associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

E.g.

```
<datadll name="garage" dll="libndGarageMib.so" path="../../ndGarageMib/Linux" ></datadll>
```

### 4.3 Startup Data

The file, **startup-cfg.xml**, may be used to provide initial data to the **testagent**, on startup. Each source project includes a file named '**<project name>-startup-cfg-demo.xml**' that contains some sample initialization data.

Once the **testagent** has been configured to load either or both of the datadll projects, you can use the contents of these files to provide initial data for that datadll.

E.g. Adding initialization data for **ndGarageMib** to **startup-cfg.xml**.

From the default **startup-cfg.xml**, at the bottom is the close tag **</config>**. Depicted as follows.

```
</nacm>
</config>
```

Insert the entire content from ndGarageMib's **ndGarageMib-startup-cfg-demo.xml**, between the tags (shown above), as depicted below:

```
</nacm>
<garage xmlns="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB">
  <garageObjects>
    <garageAddress>10 street</garageAddress>
    <garageCOLevelRisingThreshold>50</garageCOLevelRisingThreshold>
    <garageCOLevelFallingThreshold>10</garageCOLevelFallingThreshold>
  </garageObjects>
  <vehiclesEntry>
    <vehicleIndex>1</vehicleIndex>
    <vehicleLicencePlate>123 ABC</vehicleLicencePlate>
    <vehicleModel>Maserati Quattroporte</vehicleModel>
  </vehiclesEntry>
  <vehiclesEntry>
    <vehicleIndex>2</vehicleIndex>
    <vehicleLicencePlate>345 XYZ</vehicleLicencePlate>
    <vehicleModel>Jaguar F-Type</vehicleModel>
  </vehiclesEntry>
</garage>
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

</config>

## 5 Working with testagent

### 5.1 testagent Module Compliance

**testagent** implements the standard YANG modules:

ietf-inet-types  
ietf-yang-types  
SNMPv2-TC  
ietf-netconf  
ietf-netconf-monitoring  
ietf-netconf-acm  
ietf-writable-running  
ietf-confirmed-commit  
ietf-partial-lock

**ietfInterfaces** data dll implements module:

ietf-interfaces

**ndtaasmp** implements YANG modules:

- SNMPv2-MIB
- ietf-snmp

with the following submodules:

ietf-snmp-common  
ietf-snmp-engine  
ietf-snmp-target  
ietf-snmp-notification  
ietf-snmp-proxy  
ietf-snmp-community  
ietf-snmp-usm  
ietf-snmp-vacm

## 5.2 testagent Command Line Options

```
testagent [-c configDir] [-s schemaDir] [-b bakDir] [-l logDir] [-w] [-a]
```

<b>configDir</b>	folder with agent configuration files, default is config <b>sub-folder</b> of the folder where testagent is located.
<b>schemaDir</b>	folder with YANG modules. Agent searches this folder while processing "get-schema" rpc. The default is the schema <b>sub-folder</b> of the folder where testagent is located.
<b>bakDir</b>	folder where backup files are stored, default is <b>./bak</b> .
<b>logDir</b>	folder where log files are stored, default is <b>./log</b> .
<b>-w</b>	create a new log file on each agent start, this is default. File name format is testagent-xxx.log where xxx is encoded date/time in format YYMMDDhhmmss. E.g. testagent-151206191025.log
<b>-a</b>	a single log file named testagent.log is created, log output is appended

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



## 5.3 Using testagent

To start **testagent**, type **testagent** and any appropriate parameters, in a terminal window.

E.g.

```
./testagent -c ./config -s ./schema -l . -a
```

(See the previous page for more information about command line parameters.)

On startup **testagent** executes the following steps:

- read settings (from ./config/agent/ndconf.xml), here is an excerpt:

```
<ndt>
  <agents>
    <agent name="cli" dll="libndtaaccli.so"></agent>
    <agent name="nc" dll="libndtaanetconf.so" ></agent>
    <agent name="snmp" dll="libndtaasnmp.so"></agent>
  </agents>
  <datadlls>
    <datadll name="garage" dll="libndGarageMib.so" ></datadll>
    <datadll name="if" dll="libietfInterfaces.so" ></datadll>
  </datadlls>
```

- load Access Agents (listed in section **<agents>**)

- load data dlls (listed in section **<datadlls>**, in this example ndGarageMib.so) and ie, initialize and register modules implemented in each dll

- load **startup-cfg.xml** into RUNNING configuration data store

- copy RUNNING to CANDIDATE

- initialize Access agents (at this point running configuration contains parameters required by AA to be initialized properly)

- refresh state data (get initial state of objects that are not part of configuration, e.g. counters)

- record start time

- start Access agents

- if the **libndtaaccli.so** is successfully loaded, control is passed to it:

```
*****
NuDesign YANG Datastore Test Server
Version 1.0
Copyright 2014-2017 NDT
*****
> |
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

otherwise it displays:

```
Type 'q' for exit ...
```

In other words, other Access agents may be running without **libndtaaccli.so**, until 'q' is pressed.

**testagent** creates verbose log files. The name of the log file is "<executable-name>.log", e.g. "testagent.log". By default, log files are created in subdirectory ". /log".

## 6 Working with the CLI

The CLI emulates "Junos" and is started in operational mode, indicated by the "> " prompt. To get a list of available command type "?" (question mark).

```
> ?
Possible completions:
about
cls
cmpcfg
configure
datadll
exit
help
quit
script
session
show
> █
```

Execute the "configure" command to enter configuration mode, indicated by the "[edit ]" line above the prompt, which changes to "# ")

```
> configure[Enter]
Candidate configuration is now locked.
[edit ]
# █
```

In configuration mode user can modify data in the "candidate" datastore (which is locked by the CLI during execution of the "configure" command). Once the "candidate" is configured, to apply the changes to the "running" configuration, the user must execute the "commit " command.

### 6.1 ndGarageMib Example

Here is an example of how to create a row in the vehiclesTable:

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

[edit ]
# show garage[Enter] <--- display current config
// config-candidate //
garageObjects
  garageAddress 10 Street
  garageCOLevelRisingThreshold 50
  garageCOLevelFallingThreshold 10
vehiclesEntry
  vehicleIndex 1
  vehicleLicencePlate 123 ABC
  vehicleModel Maserati Quattroporte
vehiclesEntry
  vehicleIndex 2
  vehicleLicencePlate 345 XYZ
  vehicleModel Jaguar F-Type
[edit ]
# create garage vehiclesEntry 3[Enter] <--- create new entry
ok
[edit ]
# show garage[Enter] <--- display current config
// config-candidate //
garageObjects
  garageAddress 10 Street
  garageCOLevelRisingThreshold 50
  garageCOLevelFallingThreshold 10
vehiclesEntry
  vehicleIndex 1
  vehicleLicencePlate 123 ABC
  vehicleModel Maserati Quattroporte
vehiclesEntry
  vehicleIndex 2
  vehicleLicencePlate 345 XYZ
  vehicleModel Jaguar F-Type
vehiclesEntry <--- new row created, note that vehicleLicencePlate
  vehicleIndex 3 <--- and vehicleModel is not created yet
[edit ]
# create garage vehiclesEntry 3 vehicleLicencePlate "555 QQQ"[Enter] <--- create vehicleLicencePlate
ok
[edit ]
# create garage vehiclesEntry 3 vehicleModel "Audi TT"[Enter] <--- create vehicleModel
ok
[edit ]
# show garage[Enter] <--- display current config
// config-candidate //
garageObjects
  garageAddress 10 Street
  garageCOLevelRisingThreshold 50
  garageCOLevelFallingThreshold 10
vehiclesEntry
  vehicleIndex 1
  vehicleLicencePlate 123 ABC
  vehicleModel Maserati Quattroporte
vehiclesEntry
  vehicleIndex 2
  vehicleLicencePlate 345 XYZ
  vehicleModel Jaguar F-Type
vehiclesEntry
  vehicleIndex 3
  vehicleLicencePlate 555 QQQ
  vehicleModel Audi TT
[edit ]
# commit[Enter] <--- commit
ok

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

[edit ]
# show running[Enter]          <--- show running configuration
// config-running //
garage
  garageObjects
    garageAddress 10 Street
    garageCOLevelRisingThreshold 50
    garageCOLevelFallingThreshold 10
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
  vehiclesEntry
    vehicleIndex 2
    vehicleLicencePlate 345 XYZ
    vehicleModel Jaguar F-Type
  vehiclesEntry
    vehicleIndex 3
    vehicleLicencePlate 555 QQQ
    vehicleModel Audi TT[edit ]
#

```

## 6.2 rcvlan Example

Here is the example how to create vlan:

```

[edit ]
# show vlanModule[Enter]      <--- display current config
// config-candidate //
vlan
  vlanId 1
  vlanName vlan#1
  stgId 0
  vlanType 1
  vlanPortMembers 43:04:00:40
  vlanProtocolId 1
  ip
    ifIndex 1
    addr 192.168.0.11
    netMask 255.255.0.0
    bcastAddrFormat 0
    reasmMaxSize 1234
    macOffset 0
    vrfId 15
[edit ]
# edit vlanModule[Enter]     <--- navigate to vlanModule branch
ok
[edit vlanModule]
# create vlan 2[Enter]      <--- create vlan, id# = 2
ok
[edit vlanModule]
# commit[Enter]             <--- try to commit, it will fail as we did not configure
rpc-error                  <--- all elements
  error-type application
  error-tag operation-failed
  error-severity error
  error-app-tag invalid-data
  error-path /nc:config-running/nc:config/rcv:vlanModule/rcv:vlan[2]
  error-message Y: too few elems
# edit vlan 3[Enter]       <--- navigate to vlan 3

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

ok
[edit vlanModule vlan "3"]
# set vlanName "vlan no3"[Enter]          <--- set vlanName
ok
[edit vlanModule vlan "3"]
# set stgId 0[Enter]                       <--- set stgId
ok
[edit vlanModule vlan "3"]
# set vlanType byPort[Enter]              <--- set vlanType
ok
[edit vlanModule vlan "3"]
# set vlanPortMembers 1/1/1-2/1/3[Enter]  <--- set vlanPortMembers
ok
[edit vlanModule vlan "3"]
# set vlanProtocolId 1[Enter]             <--- set vlanProtocolId
ok
[edit vlanModule vlan "3"]
# create ip 1|1.2.3.4[Enter]              <--- create row in ip-addr table, key = ifIndex|ipAddr
ok
[edit vlanModule vlan "3"]
# edit ip 1|1.2.3.4[Enter]                <--- navigate to ip
ok
[edit vlanModule vlan "3" ip "1|1.2.3.4"]
# set netMask 255.255.128.0[Enter]        <--- set netMask
ok
[edit vlanModule vlan "3" ip "1|1.2.3.4"]
# set bcastAddrFormat 0[Enter]            <--- set bcastAddrFormat
ok
[edit vlanModule vlan "3" ip "1|1.2.3.4"]
# set reasmMaxSize 1234[Enter]            <--- set reasmMaxSize
ok
[edit vlanModule vlan "3" ip "1|1.2.3.4"]
# set macOffset 0[Enter]                  <--- set macOffset
ok
[edit vlanModule vlan "3" ip "1|1.2.3.4"]
#set vrfId 15[Enter]                      <--- set vrfId
ok
[edit vlanModule vlan "3" ip "1|1.2.3.4"]
#up[Enter]                                 <--- move up to vlan 3
[edit vlanModule vlan "3"]
#show[Enter]                               <--- display vlan 3
// config-candidate //
vlanId 3
vlanName vlan no3
stgId 0
vlanType byPort
vlanPortMembers 1/1/1-2/1/2
vlanProtocolId 1
ip
ifIndex 1
addr 1.2.3.4
netMask 255.255.128.0
bcastAddrFormat 0
reasmMaxSize 1234
macOffset 0
vrfId 15
[edit vlanModule vlan "3" ]
# commit[Enter]                            <--- commit
ok
[edit vlanModule vlan "3" ]
# quit[Enter]
Candidate configuration is now unlocked.
> show running[Enter]                      <--- show running configuration

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
// config-running //  
...  
vlanId 3  
vlanName vlan no3  
stgId 0  
vlanType byPort  
vlanPortMembers 1/1/1-2/1/2  
vlanProtocolId 1  
ip  
  ifIndex 1  
  addr 1.2.3.4  
  netMask 255.255.128.0  
  bcastAddrFormat 0  
  reasmMaxSize 1234  
  macOffset 0  
  vrfId 15  
> █
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 7 Working with *ncclient*

**ncclient** is a command line NetConf client. Command line handling is the same as in the CLI AA. The difference is that **ncclient** does not have a configuration mode. Rather, commands are an implementation of NetConf rpc commands, defined in the `ietf-netconf` Yang module. A few commands (E.g. `edit-config`) expect xml as an argument/payload, and it would be impractical to provide such input in a command line. So the argument for these commands is the name of the file that contains part of the xml string required.

To start **ncclient**, from `<SDK install_dir>/ndconf/ncclient`, type `./ncclient` in a terminal window<sup>4</sup>.

The **ncclient** 'connect' command establishes communication with the **testagent**. If the "connect" succeeds, **ncclient** sends a "hello" message (shown in light blue below), and reads a "hello" from server:

```
> connect 127.0.0.1 <a valid user name here>[Enter]
Password: *****
Fingerprint: A0 E0 6A E9 E1 29 85 3E 77 37 AB 20 97 88 03 88 83 5B 78 ED
Authentication methods: publickey,password,keyboard-interactive
Sending NETCONF client <hello>
<?xml version="1.0" encoding="UTF-8"?>
<hello>
<capabilities>
<capability>
urn:ietf:params:netconf:base:1.0
</capability>
<capability>
urn:ietf:params:netconf:base:1.1
</capability>
</capabilities>
</hello>
]]>]]>
Reading NETCONF server <hello>
  capabilities
    capability urn:ietf:params:netconf:base:1.0
  ...
> ]
```

At this point, a session is created, and the user can issue netconf commands.

### 7.1 ndGarageMib Example

Here is an example how to create a row in the vehiclesTable:

```
> get-config candidate xpath //garage[Enter] <--- get the candidate configuration
data
  garage
...
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
  vehiclesEntry
```

4

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

    vehicleIndex 2
    vehicleLicencePlate 345 XYZ
    vehicleModel Jaguar F-Type
  vehiclesEntry
    vehicleIndex 3
    vehicleLicencePlate 555 QQQ
    vehicleModel Audi TT
> lock candidate[Enter]          <--- lock configuration
ok
> list edit-config c1[Enter]     <--- display content of file to be used as argument
<default-operation>merge</default-operation>
<test-option>set</test-option>
<config>
  <t:garage xmlns:t="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB" >
    <t:vehiclesEntry xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="create" >
      <t:vehicleIndex>4</t:vehicleIndex>
      <t:vehicleLicencePlate>777 SWE</t:vehicleLicencePlate>
      <t:vehicleModel>Koenigsegg One:1</t:vehicleModel>
    </t:vehiclesEntry>
  </t:garage>
</config>

> edit-config candidate c1[Enter] <--- execute edit-config rpc
ok
> lock running[Enter]           <--- lock configuration
ok
> commit[Enter]                 <--- commit changes to running configuraton
ok
> unlock running[Enter]         <--- unlock configuration
ok
> unlock candidate[Enter]       <--- unlock configuration
ok
> get xpath //garage[Enter]     <--- get running configuration
data
  garage
...
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
    vehicleStatus active
  vehiclesEntry
    vehicleIndex 2
    vehicleLicencePlate 345 XYZ
    vehicleModel Jaguar F-Type
    vehicleStatus active
  vehiclesEntry
    vehicleIndex 3
    vehicleModel Audi TT
    vehicleLicencePlate 555 QQQ
    vehicleStatus active
  vehiclesEntry                <--- new entry
    vehicleIndex 4
    vehicleLicencePlate 777 SWE
    vehicleModel Koenigsegg One:1
    vehicleStatus active
>

```



## 7.2 rcvlan based example

Here is the example how to create row in vlanTable:

```

> get-config candidate xpath //vlan[Enter] <--- get vlan branch in candidate configuration
rpc-reply
data
  vlan
    vlanId 1
    vlanName vlan#1
    stgId 0
    vlanType 1
    vlanPortMembers 43:04:00:40
    vlanProtocolId 1
    ip
      ifIndex 1
      addr 192.168.0.11
      netMask 255.255.0.0
      bcastAddrFormat 0
      reasmMaxSize 1234
      macOffset 0
      vrfId 15
  vlan
    vlanId 3
    vlanName vlan no3
    stgId 0
    vlanType byPort
    vlanPortMembers 1/1/1-2/1/2
    vlanProtocolId 1
    ip
      ifIndex 1
      addr 1.2.3.4
      netMask 255.255.128.0
      bcastAddrFormat 0
      reasmMaxSize 1234
      macOffset 0
      vrfId 15
> lock candidate[Enter] <--- lock configuration
ok
> list edit-config createvlan[Enter] <--- display content of createvlan.xml
<test-option>set</test-option>
<config>
  <t:vlanModule xmlns:t="urn:ietf:params:xml:ns:yang:smiv2:rcvlan" >
    <t:vlan xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="create" >
      <t:vlanId>2</t:vlanId>
      <t:vlanName>vlan#2</t:vlanName>
      <t:stgId>2</t:stgId>
      <t:vlanType>1</t:vlanType>
      <t:vlanPortMembers>13:04:11:20</t:vlanPortMembers>
      <t:vlanProtocolId>1</t:vlanProtocolId>
      <t:ip>
        <t:ifIndex>2</t:ifIndex>
        <t:addr>192.168.0.22</t:addr>
        <t:netMask >255.255.0.0</t:netMask>
        <t:bcastAddrFormat>0</t:bcastAddrFormat>
        <t:reasmMaxSize>1234</t:reasmMaxSize>
        <t:macOffset>0</t:macOffset>
        <t:vrfId>15</t:vrfId>
      </t:ip>
    </t:vlan>
  </t:vlanModule>
</config>

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

> edit-config candidate createvlan [Enter]      <--- execute edit-config rpc
ok
> lock running [Enter]                        <--- lock configuration
ok
> commit [Enter]                              <--- commit changes to running configuraton
ok
> unlock running [Enter]                      <--- unlock configuration
ok
> unlock candidate [Enter]                    <--- unlock configuration
ok
> get xpath //vlan [Enter]                    <--- get running configuration
rpc-reply
data
  vlan
    vlanId 1
    vlanName vlan#1
    stgId 0
    vlanType 1
    vlanPortMembers 43:04:00:40
    vlanProtocolId 1
    ip
      ifIndex 1
      addr 192.168.0.11
      netMask 255.255.0.0
      bcastAddrFormat 0
      reasmMaxSize 1234
      macOffset 0
      vrfId 15
  vlan
    vlanId 3
    vlanName vlan no3
    stgId 0
    vlanType byPort
    vlanPortMembers 1/1/1-2/1/2
    vlanProtocolId 1
    ip
      ifIndex 1
      addr 1.2.3.4
      netMask 255.255.128.0
      bcastAddrFormat 0
      reasmMaxSize 1234
      macOffset 0
      vrfId 15
  vlan
    vlanId 2
    vlanName vlan#2
    stgId 22
    vlanType 1
    vlanPortMembers 13:04:11:20
    vlanProtocolId 1
    ip
      ifIndex 2
      addr 192.168.0.22
      netMask 255.255.0.0
      bcastAddrFormat 0
      reasmMaxSize 1234
      macOffset 0
      vrfId 15
>

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

### 7.3 Notifications in ncclient

To check the notification streams supported:

```
> get xpath //stream[Enter]
rpc-reply
data
  stream
    name NETCONF
    description NETCONF stream
    replaySupport true
    replayLogCreationTime 2017-04-06T21:10:58Z
  stream
    name SNMP
    description SNMP stream
    replaySupport false
```

To subscribe for notifications:

```
> create-subscription[Enter]
rpc-reply
ok
>
```

Later (libndGarageMib.so sends coLevelFallingAlarm and coLevelRisingAlarm periodically):

```
notification
  eventTime 2017-04-06T21:20:22Z
  coLevelFallingAlarm
    object-1
      garageCOLevel 9
```

The subscription can be terminated by using <close-session>

```
> close-session[Enter]
rpc-reply
ok
>
```

or <kill-session> (from a different client instance)

When subscribing for notifications with replay, a user must specify {startTime} or {startTime, stopTime}.

If only the startTime is specified, agent will replay all available notifications in the log since the startTime. After finishing the replay agent sends a "replayComplete" notification. The agent will continue to send notifications to the client after that.

```
> create-subscription startTime 2017-04-06T00:00:00Z[Enter]
>
rpc-reply
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
ok

notification
  eventTime 2017-04-06T23:16:04Z
  netconf-session-start
    username cli
    session-id 1

notification
  eventTime 2017-04-06T23:16:04Z
  netconf-session-start
    username snmp
    session-id 2

notification
  eventTime 2017-04-06T23:16:10Z
  netconf-session-start
    username borislav
    session-id 3

notification
  eventTime 2017-04-06T23:16:25Z
  coLevelRisingAlarm
    object-1
      garageCOLevel 51

(continued next page)
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
notification
  eventTime 2017-04-06T23:16:41Z
  replayComplete      <<<--- after this notification, session becomes regular notif subscriber
...
notification
  eventTime 2017-04-06T23:20:00Z
  coLevelFallingAlarm
  object-1
  garageCOLevel 9
```

There are two scenarios when stopTime is specified:

1) If the stopTime is before current time, agent will replay notifications up to the stopTime and cancel the subscription.

```
> create-subscription startTime 2017-04-15T00:00:00Z stopTime 2017-04-06T23:17:00Z[Enter]
>
rpc-reply
  ok

notification
  eventTime 2017-04-06T23:16:04Z
  netconf-session-start
  username cli
  session-id 1

notification
  eventTime 2017-04-06T23:16:04Z
  netconf-session-start
  username snmp
  session-id 2

notification
  eventTime 2017-04-06T23:16:10Z
  netconf-session-start
  username borislav
  session-id 3

notification
  eventTime 2017-04-06T23:16:25Z
  coLevelRisingAlarm
  object-1
  garageCOLevel 51

notification
  eventTime 2017-04-06T23:21:07Z
  replayComplete

notification
  eventTime 2017-04-06T23:21:07Z
  notificationComplete      <<<--- after this notification agent will stop sending new notifications
```

2) If the stopTime is in the future, the agent will replay all available notifications in the log since the startTime and send a "replayComplete" notification. The agent will continue to send any new notifications until the stopTime, at which point the agent will send a "notificationComplete" notification.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

> create-subscription startTime 2017-04-06T23:23:40Z stopTime 2017-04-06T23:30:00Z[Enter]
>
rpc-reply
  ok

notification
  eventTime 2017-04-06T23:23:42Z
  coLevelRisingAlarm
    object-1
      garageCOLevel 51

notification
  eventTime 2017-04-06T23:24:16Z
  replayComplete <<<--- after this notification agent will continue to send notifications ...

...

notification
  eventTime 2017-04-06T23:24:33Z
  coLevelFallingAlarm
    object-1
      garageCOLevel 9

...

notification
  eventTime 2017-04-06T23:30:00Z <<<--- ... until stopTime
  notificationComplete

```

An error may be reported, depending on the validity of the subscription request. E.g. if the startTime is in the future:

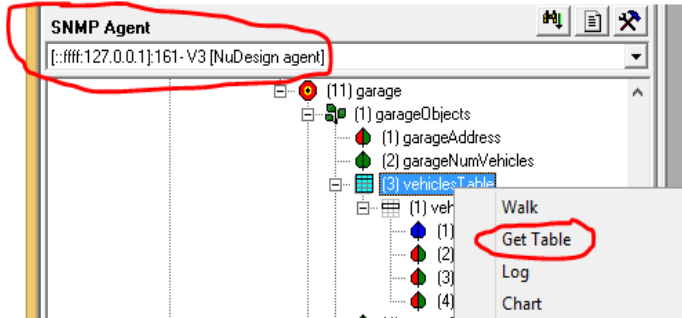
```

> create-subscription startTime 2019-04-24T10:00:00Z[Enter]
>
rpc-reply
  rpc-error
    error-type protocol
    error-tag bad-element
    error-severity error
    error-app-tag invalid-data
    error-path /rpc/create-subscription/startTime

```

## 8 Working with SNMP

To interact with the SNMP access agent you need to use an SNMP manager. For the purposes of this document, we are using NDT's MIBBrowser Pro. Start Visual MIBBrowser Pro, load ND-GARAGE-MIB (NDGarageV2.mib), select the **testagent**'s address as the target (127.0.0.1), right click "vehicleTable" and select "Get Table" from the context menu (initiated using a right mouse click):

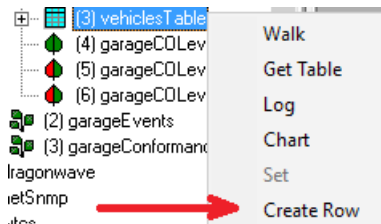


produces the following:

INDEX	vehicleLicencePlate	vehicleModel	vehicleStatus
.1	123 ABC	Maserati Quattroporte	active (1)
.2	345 XYZ	Jaguar F-Type	active (1)
.3	555 QQQ	Audi TT	active (1)
.4	777 SWE	Koenigsegg One:1	active (1)

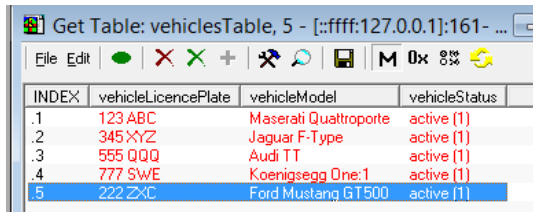
Instance	Syntax	Value	Set Value
vehicleStatus.5	RowStatus	noSuchInstance	createAndGo (4)
vehicleLicencePlate.5	DisplayString	noSuchInstance	222 ZXC
vehicleModel.5	DisplayString	noSuchInstance	Ford Mustang GT500

Now create a new row with MIBBrowser Pro, select "Create Rpw":



Fill in the details and then send the "set" request by clicking "Execute" button (circled in red). If there were not creation problems, then you can refresh the getTable window if it doesn't appear automatically:

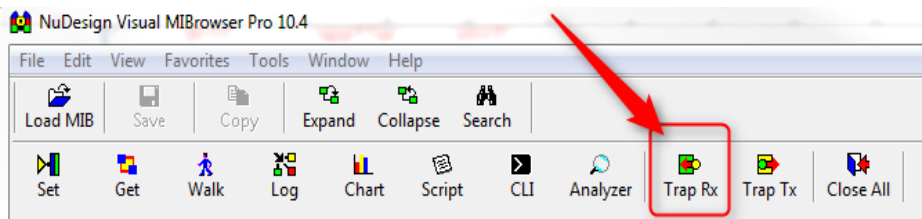
This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



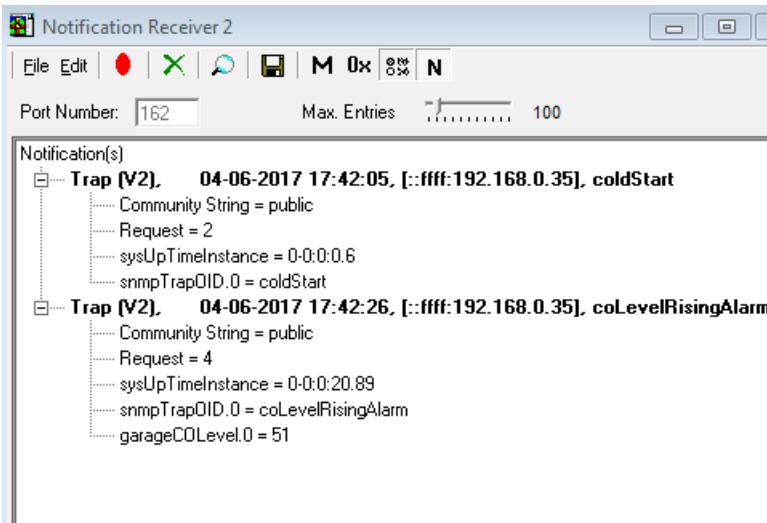
INDEX	vehicleLicencePlate	vehicleModel	vehicleStatus
1	123 ABC	Maserati Quattroporte	active (1)
2	345 XYZ	Jaguar F-Type	active (1)
3	567 QQQ	Audi TT	active (1)
4	777 SWE	Koenigsegg One:1	active (1)
5	222 ZXC	Ford Mustang GT500	active (1)

### 8.1 Notifications in MIBBrowser

Open a trap receiver (Trap Rx):



then (re)start the **testagent**. You'll see something like the following:



The first entry is an SNMP v3 trap notification indicating libndtsnmpaa.so has loaded and been started. i.e. the **coldstart**.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



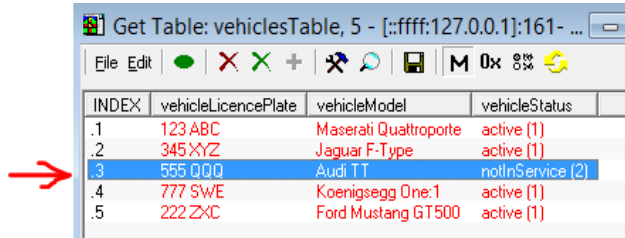
## 8.2 Putting it All together

Retrieving configurations from **ncclient**:

```
> get-config candidate xpath //vehiclesEntry[Enter] <--- get candidate configuration
data
garage
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
  vehiclesEntry
    vehicleIndex 2
    vehicleLicencePlate 345 XYZ
    vehicleModel Jaguar F-Type
  vehiclesEntry
    vehicleIndex 3
    vehicleLicencePlate 555 QQQ
    vehicleModel Audi TT
  vehiclesEntry
    vehicleIndex 4
    vehicleLicencePlate 777 SWE
    vehicleModel Koenigsegg One:1
  vehiclesEntry
    vehicleIndex 5
    vehicleLicencePlate 222 ZXC
    vehicleModel Ford Mustang GT500
> get-config running xpath //vehiclesEntry[Enter] <--- get running configuration
data
garage
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
  vehiclesEntry
    vehicleIndex 2
    vehicleLicencePlate 345 XYZ
    vehicleModel Jaguar F-Type
  vehiclesEntry
    vehicleIndex 3
    vehicleLicencePlate 555 QQQ
    vehicleModel Audi TT
  vehiclesEntry
    vehicleIndex 4
    vehicleLicencePlate 777 SWE
    vehicleModel Koenigsegg One:1
  vehiclesEntry
    vehicleIndex 5
    vehicleLicencePlate 222 ZXC
    vehicleModel Ford Mustang GT500
> █
```

The running and candidate configurations are the same. The reason is that SNMP AA synchronizes running and candidate after each successful set request.

Set one of the rows in the table to 'notInService' by right mouse clicking in the row and selecting it:



INDEX	vehicleLicencePlate	vehicleModel	vehicleStatus
.1	123 ABC	Maserati Quattroporte	active (1)
.2	345 XYZ	Jaguar F-Type	active (1)
.3	555 QQQ	Audi TT	notInService (2)
.4	777 SWE	Koenigsegg One:1	active (1)
.5	222 ZXC	Ford Mustang GT500	active (1)

Only 'active' rows are part of configuration. You can verify this in **ncclient**:

```
> get-config running xpath //vehiclesEntry[Enter] <--- get running configuration
data
garage
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
  vehiclesEntry
    vehicleIndex 2
    vehicleLicencePlate 345 XYZ
    vehicleModel Jaguar F-Type <--- #3 is deleted
  vehiclesEntry
    vehicleIndex 4
    vehicleLicencePlate 777 SWE
    vehicleModel Koenigsegg One:1
  vehiclesEntry
    vehicleIndex 5
    vehicleLicencePlate 222 ZXC
    vehicleModel Ford Mustang GT500
>
```

Delete a row using **ncclient**:

```
> list edit-config d1[Enter] <--- content of d1.xml
<default-operation>merge</default-operation>
<test-option>set</test-option>
<config>
  <t:garage xmlns:t="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB" >
    <t:vehiclesEntry xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="delete" >
      <t:vehicleIndex>4</t:vehicleIndex>
    </t:vehiclesEntry>
  </t:garage>
</config>

> lock candidate[Enter] <--- lock configuration
ok
> edit-config candidate d1[Enter] <--- execute edit-config rpc (arg = d1.xml)
ok
> lock running[Enter] <--- lock configuration
ok
> commit[Enter] <--- commit changes to running configuraton
ok
> unlock running[Enter] <--- unlock configuration
ok
> unlock candidate[Enter] <--- unlock configuration
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
ok
> get-config running xpath //vehiclesEntry[Enter] <--- get running configuration
data
  garage
    vehiclesEntry
      vehicleIndex 1
      vehicleLicencePlate 123 ABC
      vehicleModel Maserati Quattroporte
    vehiclesEntry
      vehicleIndex 2
      vehicleLicencePlate 345 XYZ
      vehicleModel Jaguar F-Type <--- #4 is deleted
    vehiclesEntry
      vehicleIndex 5
      vehicleLicencePlate 222 ZXC
      vehicleModel Ford Mustang GT500
>
```

Check with MIBrowser, by refreshing the window:

INDEX	vehicleLicencePlate	vehicleModel	vehicleStatus
.1	123 ABC	Maserati Quattroporte	active (1)
.2	345 XYZ	Jaguar F-Type	active (1)
.5	222 ZXC	Ford Mustang GT500	active (1)

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 9 CLI

This section describes commands implemented by the CLI Access Agent (libndtacli.so).

Type ? to see a list of the available commands:

```
> ?
Possible completions:
about
cls
cmpcfg
configure
datadll
exit
help
quit
script
session
show
sleep
> █
```

### 9.1 Command completion

To complete a partial command enter <tab> immediately after a command prefix, e.g.

show

```
> s[Tab]
Ambiguous command: s
Possible completions:
session
show
> s █
```

In the example below, there are 2 commands that start with prefix "s". List of possible completions is printed along with the error message "Ambiguous command". New command line is extended to the first 'non-ambiguous' letter (in this case new cmd line is not extended as "s" is already the longest common prefix for these 2 commands) and cursor positioned immediately after it.

Type next letter and [Tab] :

```
> conf[Tab]
```

and command will be completed:

```
> configure █
```

If there is no command that starts with prefix:

```
> cond[Tab]
Couldn't complete, no match found
> cond
```

## 9.2 Commands and parameters

### 9.2.1 ? or <prefix>?

List all commands or just commands that starts with <prefix>. E.g.

```
> s?
Possible completions:
  session
  show
>
```

### 9.2.2 about

*about*

Show **testagent** information:

```
> about[Enter]
*****
* NuDesign YANG Datastore Test Agent
* Version x.x.x.x
* Copyright 2014-2019 NDT
*****
>
```

### 9.2.3 cls

*cls*

Clear the screen.

### 9.2.4 cmpcfg

*cmpcfg <candidate|running|startup|url> <candidate|running|startup|url>*

compare configurations.

```
> cmpcfg startup running[Enter]

right: running
diff:
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
    config
  xmlns="urn:ietf:params:xml:ns:yang:ietf-snmp"
    snmp
...
>
```

### 9.2.5 configure

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## configure

Enter configuration mode (indicated by `[edit]` line above prompt, prompt changes to `#`)

```
> configure[Enter]
[edit ]
# █
```

See "Configuration Mode" section for details.

### 9.2.6 `datadll [load|unload] <name>`

`datadll [load | unload] <name>`

Load or unload data dll. Argument `<name>` is the name attribute for `datadll` element in "ndconf.xml":

```
<datadlls>
  <datadll name="garage" dll="Garage.dll" ></datadll>
```

Argument `<name>` is also displayed in the output of "show datadll":

```
> show datadll[Enter]
data
  dll
    name if
    path <SDK installdir>/ndconf/lib64/libietfInterfaces.so.0
    loaded true
>
> datadll unload if[Enter]
ok
> datadll load if[Enter]
ok
> █
```

NOTE: Loading or unloading causes a "netconf-capability-change" notification.

### 9.2.7 `exit`

`exit`

In operational mode, exit the `testagent`.

```
> exit
disconnected
[a-user@localhost testagent]$ █
```

In configuration mode, "exit" command moves the current position one level up. If at the root of configuration tree, exit configuration mode.

```
[edit ]                                <--- /
# edit garage vehiclesEntry 1[Enter]
[edit garage vehiclesEntry "1" ]       <--- /garage/vehiclesEntry[vehiclesIndex=1]
# exit[Enter]
[edit garage ]                          <--- /garage
# exit[Enter]
[edit ]                                <--- /
# exit[Enter]
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
> | <--- op mode
```

### 9.2.8 help

*help* <cmdname>

Print help for command specified in argument.

```
> help configure[Enter]
Enter configuration mode
> |
```

### 9.2.9 quit

*exit the testagent.*

### 9.2.10 script

*script* <file>

Execute a script file.

### 9.2.11 session

*session*

Display CLI session info

```
> session[Enter]
SESSION: user:cli, id:1, status:connected
> |
```

### 9.2.12 show

*show* [agents | datadll | history | running]

***show agents***

Display Access agents status

```
> show agents[Enter]
cli
  path: libndtaacli.so.0
  name: cli
  version: 1.1.0.0
  status: OK
nc
  path: libndtaanetconf.so.0
  name: netconf
  version: 1.0.0.0
  status: OK
snmp
  path: libndtaasnmp.so.0
  name: snmp
  version: 1.2.0.0
  status: OK
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
> █
```

### **show datadll**

Display list of configured data dlls with 'loaded' status

```
> show datadll[Enter]
data
dll
  name if
  path <SDK installdir>/ndconf/lib64/libietfInterfaces.so.0
  loaded true
dll
  name garage
  path /home/a-user/ndconf/ndGarageMib/Linux/libndGarageMib.so
  loaded true
> █
```

### **show history**

Display command line history.

```
> show history[Enter]
0: about
1: show running
> █
```

### **show running**

Display running configuration

```
> show running[Enter]
// config-running //
snmp
  engine
    enabled true
    listen
      name lis-1
      udp
        ip 127.0.0.1
        port 4161
    version
      v1
      v2c
      v3
  engine-id 80:00:12:99:04:6e:64:74
...
> █
Configuration mode
```

### **9.2.13 sleep**

*sleep* <seconds>



### 9.3 Configuration Mode

Configuration mode commands operate on configuration data. To enter configuration mode execute the `configure` command. The commands available in configuration mode at any position are:

```
[edit ]
# ?
Possible completions:
commit
create
delete
diff
edit
exit
help
load
quit
run
save
set
show
top
up
[edit ]
# █
```

Commands operate on the config data available at the current position, see `edit`, `up`, `top`, `exit`. On entering configuration mode, the current position is set to root. Excerpts from **testagent** configuration to be used in examples:

```
<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <garage xmlns="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB">
    ...
    <snmp xmlns="urn:ietf:params:xml:ns:yang:ietf-snmp">
    ...
```

Possible arguments for the "set" command at root level are:

```
[edit ]
# set ?
Possible completions:
<[Enter]>          Execute this command
config
garage
interfaces
mib-2.snmp
nacm
snmp
system
[edit ]
# set █
```

Change position to "garage"

```
[edit ]
# edit garage
[edit garage ]
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
# █
```

Possible args for "set" command at this level are:

```
[edit garage ]
# set ?
Possible completions:
<[Enter]>      Execute this command
garageObjects
vehiclesEntry
[edit garage ]
# set █
```

### 9.3.1 commit

*commit*

Applies changes in candidate configuration to running.

```
> commit[Enter]
ok
> █
```

### 9.3.2 create

*create arg1 [arg2] [...]*

Create configuration data (corresponds to edit-config, operation="create").

```
[edit garage]
# create vehiclesEntry 5[Enter]
ok
[edit garage ]
# show[Enter]
// config-candidate //
...
    vehiclesEntry
        vehicleIndex 1
...
    vehiclesEntry
        vehicleIndex 2
...
    vehiclesEntry
        vehicleIndex 5
[edit garage ]
# █
```

### 9.3.3 delete

*delete arg1 [arg2] [...]*

Delete configuration data.

```
[edit garage ]
# delete vehiclesEntry 2[Enter]
ok
[edit garage ]
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
# show[Enter]
// config-candidate //
...
  vehiclesEntry
    vehicleIndex 1
...
  vehiclesEntry
    vehicleIndex 5
...
[edit garage ]
# █
```

### 9.3.4 diff

```
diff [candidate running | candidate <file> | running candidate | running <file> ]
```

Perform a difference operation on the specifici

### 9.3.5 edit

```
edit arg1 [arg2] [...]
```

[edit] line shows current position in configuration hierarchy. To navigate configuration hierarchy use **edit**, **up** and **top**.

```
[edit ] <--- /
# edit garage vehiclesEntry 1[Enter]
[edit garage vehiclesEntry "1" ] <--- /garage/vehiclesEntry[vehiclesIndex=1]
# show[Enter]
// config-candidate //
  vehicleIndex 1
  vehicleLicencePlate 123 ABC
  vehicleModel Maserati Quattroporte
[edit garage vehiclesEntry "1" ] <--- /garage/vehiclesEntry[vehiclesIndex=1]
# up[Enter]
[edit garage ] <--- /garage
# show[Enter]
// config-candidate //
  garageObjects
    garageAddress 10 Street
...
[edit garage ] <--- /garage
# up[Enter]
[edit ] <--- /
# edit garage vehiclesEntry 1[Enter]
[edit garage vehiclesEntry "1" ] <--- /garage/vehiclesEntry[vehiclesIndex=1]
# top[Enter]
[edit ] <--- /
# █
```

### 9.3.6 exit

```
exit [discard]
```

Exit configuration mode if the current position is configuration root. Otherwise move one level up (the same as

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

'up' command)

```
[edit ]
# exit[Enter]
> █
```

If there are uncommitted changes made to candidate configuration

```
[edit ]
# exit[Enter]
There are uncommitted changes.
Use 'commit' to commit the changes, or 'exit discard' to discard them.
[edit ]
# exit discard[Enter]
> █
```

exit discard executed at any level is the same as quit.

### 9.3.7 load

*load <file> candidate|running*

Load configuration from <file> into candidate or running configuration.

```
[edit ]
# load newcfg.xml candidate[Enter]
ok
[edit ]
# █
```

### 9.3.8 quit

*quit*

Exit configuration mode discarding changes (if any), the same as 'exit discard'.

```
[edit ]
# exit[Enter]
There are uncommitted changes.
Use 'commit' to commit the changes, or 'exit discard' to discard them.
[edit ]
# quit[Enter]
> █
```

### 9.3.9 run

*run arg1 [arg2] [...]*

Execute system command.

```
[edit ]
# run ping 192.168.0.1[Enter]
Executing 'ping 192.168.0.1'
Press Ctrl-C to interrupt

Pinging 192.168.0.1 with 32 bytes of data:
Reply from 192.168.0.1: bytes=32 time<1ms TTL=64
```

```
...
Approximate round trip times in milliseconds:
  Minimum = 0ms, Maximum = 0ms, Average = 0ms
[edit ]
# █
```

### 9.3.10 save

```
save candidate|running <file>
```

Save running or candidate configuration to file.

```
[edit ]
# save candidate test[Enter]
  ok
[edit ]
# █
```

### 9.3.11 set

```
set arg1 [arg2] [...]
```

Modify configuration data (corresponds to edit-config, operation="replace").

```
[edit ]
# show garage garageObjects garageAddress[Enter]
// config-candidate //
  garageAddress 10 Street <-- current address
[edit ]
# set garage garageObjects garageAddress "123 avenue rd."[Enter] <-- change address
  ok
[edit ]
# show garage garageObjects garageAddress[Enter]
// config-candidate //
  garageAddress 123 avenue rd. <-- new address
[edit ]
# █
```

### 9.3.12 show

```
show [running] [arg1] [arg2] [...]
```

Display configuration. By default displays candidate configuration data.

```
[edit ]
# show[Enter]
// config-candidate //
  snmp
    engine
      enabled true
      listen
      udp
...

```

To display running configuraton:

```
[edit ]
# show running[Enter]
// config-running //
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
snmp
  engine
    enabled true
    listen
      udp
...

```

Argument can be any node that is direct child of the current position.

```
# show ?
Possible completions:
<[Enter]>      Execute this command
garage
mib-2.snmp
history
running
snmp
system
[edit ]
# show running ?
Possible completions:
<[Enter]>      Execute this command
garage
mib-2.snmp
snmp
system
[edit ]
# show running █

```

```
[edit ]
# show garage[Enter]
// config-candidate //
  garageObjects
  garageAddress Avenue road
[edit ]
# █

```

### 9.3.13 top

*top*

Change current position to configuration root

See *edit*

### 9.3.14 up

*up*

Change current position one level up

See *edit*

## 10 *ncclient* - NetConf client

**ncclient** is the command line NetConf client. To start, from `<SDK installdir>/ndconf/ncclient` type `./ncclient` in a console:

```
*****
NuDesign DEMO NetConf Client
Version 1.0.0.0
Copyright (c) 2014-2019 NuDesign Technologies Inc.
*****
> █
```

UI is similar to CLI . Type ? (question mark) to see the list of available commands:

```
> ?
Possible completions:
about
cls
connect
exec
exit
help
hierarchy
list
quit
script
show
> █
```

### 10.1 Commands and parameters

#### 10.1.1 ? or <prefix>?

List all commands or just commands that start with <prefix>. E.g.

```
> c?
Possible completions:
configure
connect
> c █
```

### 10.1.2 about

*about*

Show **ncclient** information:

```
> about[Enter]
*****
NuDesign DEMO NetConf Client
Version 1.0.0.0
Copyright (c) 2014-2019 NuDesign Technologies Inc.
*****
> █
```

### 10.1.3 cls

This command clears the current display.

### 10.1.4 connect <host> [username]

Connect **ncclient** to NetConf agent (**testagent**). The host address is a mandatory argument. User name can be specified as the second argument, or at the "User:" prompt. The password must always be provided at the "Password:" prompt. This is example of unsuccessful connection:

```
> connect 192.168.0.21[Enter]
User: admin
Password: *****
Failed to connect to 192.168.0.21!
> █
```

If the connect succeeds, **ncclient** sends "hello" message (shown in light blue below), and reads the "hello" from server:

```
> connect 127.0.0.1 admin[Enter]
Password: *****
Fingerprint: A0 E0 6A E9 E1 29 85 3E 77 37 AB 20 97 88 03 88 83 5B 78 ED
Authentication methods: publickey,password,keyboard-interactive
Sending NETCONF client <hello>
<?xml version="1.0" encoding="UTF-8"?>
<hello>
<capabilities>
<capability>
urn:ietf:params:netconf:base:1.0
</capability>
<capability>
urn:ietf:params:netconf:base:1.1
</capability>
</capabilities>
</hello>
]]>]]>
Reading NETCONF server <hello>
capabilities
  capability urn:ietf:params:netconf:base:1.0
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



```
...
> █
```

At this point 'netconf' commands are available:

```
> ?
Possible completions:
about
cancel-commit
close-session
cls
commit
connect
copy-config
create-subscription
delete-config
discard-changes
disconnect
edit-config
exec
exit
get
get-config
get-schema
help
hierarchy
kill-session
list
lock
partial-lock
partial-unlock
quit
rpc
script
show
unlock
validate
> █
```

### 10.1.5 cancel-commit

*Cancel-commit [persist-id]*

Sends request a [specific] commit

```
> cancel-commit[Enter]
<?xml version="1.0" encoding="UTF-8" ?><rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-i
d="4">
<cancel-commit xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"></cancel-commit>
</rpc>
ok
> █
```

### 10.1.6 close-session

#### Close-session

Sends request for graceful termination of a NETCONF session.

```
> close-session[Enter]
<?xml version="1.0" encoding="UTF-8" ?><rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <close-session xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" />
</rpc>
ok
> █
```

### 10.1.7 create-subscription [startTime YYYY-MM-DDThh:mm:ssZ]

This operation initiates an event notification subscription that will send asynchronous event notifications to the **ncclient** until the subscription terminates.

```
> create-subscription[Enter]
ok
> █
```

Later:

```
>
notification
  eventTime 2015-04-03T20:56:19Z
  netconf-capability-change
  changed-by
    username cli
    session-id 1
  added-capability urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB?module=ND-GARAGE-
MIB&revision=2004-07-09
```

Subscribe for notifications with replay:

```
> create-subscription startTime 2015-04-01T00:00:00Z[Enter]
notification
  eventTime 2015-04-06T21:10:59Z
  netconf-session-start
  username cli
  session-id 1

notification
  eventTime 2015-04-06T21:10:59Z
  netconf-session-start
  username snmp
  session-id 2

notification
  eventTime 2015-04-06T21:11:17Z
  netconf-session-start
  username borislav
  session-id 3

notification
  eventTime 2015-04-06T21:11:19Z
  coLevelRisingAlarm
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

object-1
garageCOLevel 51

rpc-reply
ok
> █

```

### 10.1.8 commit

*commit*

The commit operation instructs the device to implement the configuration data contained in the candidate configuration, i.e candidate configuration is copied to running configuration.

```

> commit[Enter]
ok
> █

```

### 10.1.9 copy-config

*copy-config <source> <target>*

*copy-config candidate|running|startup|<url> candidate|running|startup|<url>*

Supported value for url is file URI scheme. Extension "xml" is appended to url.

```

> copy-config candidate abc[Enter]
source: candidate
target: file:///abc.xml
ok
> █

```

### 10.1.10 delete-config

*delete-config startup|<url>*

Supported value for url is file URI scheme.

```

> delete-config startup[Enter]
target: startup
ok
> █

```

### 10.1.11 discard-changes

*discard-changes*

Revert the candidate configuration to the current running configuration

```

> discard-changes[Enter]
ok
> █

```

### 10.1.12 disconnect

*disconnect*

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

Disconnects **ncclient** from the **testagent**, 'netconf' commands are removed.

### 10.1.13 edit-config

`edit-config candidate|running <file-name>`

The <edit-config> operation loads all or part of a specified configuration to the specified target configuration datastore. This command reads file specified as the argument (extension xml is assumed) from folder NGdemo/config/edit-config, and creates **edit-config** rpc request to be executed by **testagent**. E.g. file might contain:

```
<default-operation>merge</default-operation>
<test-option>set</test-option>
<config>
  <t:garage xmlns:t="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB" >
    <t:vehiclesEntry xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="create" >
      <t:vehicleIndex>4</t:vehicleIndex>
      <t:vehicleLicencePlate>777 SWE</t:vehicleLicencePlate>
      <t:vehicleModel>Koenigsegg One:1</t:vehicleModel>
    </t:vehiclesEntry>
  </t:garage>
</config>
```

This command will add rpc wrapper around it:

```
<rpc message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <candidate />
    </target>
    CONTENT OF THE FILE IS INSERTED HERE
  </edit-config>
</rpc>
```

before passing it to **testagent** for processing.

```
> get-config candidate xpath //vehiclesEntry[Enter] <--- retrieve current config
data
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
...
> list edit-config 2[Enter] <--- check content of 2.xml
<default-operation>merge</default-operation>
<test-option>set</test-option>
<config>
  <t:garage xmlns:t="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB" >
    <t:vehiclesEntry xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:ope
ration="create" >
      <t:vehicleIndex>4</t:vehicleIndex>
      <t:vehicleLicencePlate>777 SWE</t:vehicleLicencePlate>
      <t:vehicleModel>Koenigsegg One:1</t:vehicleModel>
    </t:vehiclesEntry>
  </t:garage>
</config>
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

> edit-config candidate 2[Enter]          <--- send edit-config rpc
ok
> get-config candidate xpath //vehiclesEntry[Enter] <--- retrieve config
data
  vehiclesEntry
    vehicleIndex 1
    vehicleLicencePlate 123 ABC
    vehicleModel Maserati Quattroporte
...
  vehiclesEntry          <--- new entry created
    vehicleIndex 4
    vehicleLicencePlate 777 SWE
    vehicleModel Koenigsegg One:1
> █

```

### 10.1.14 exec

Exec <exec-cmd>

This command executes the specified system command 'exec-cmd'.

```

> exec ls -l[Enter]
drwxrwxr-x. 3 ndconf ndconf   4096 Aug 15 11:06 .
drwxrwxr-x. 4 ndconf ndconf   4096 Nov  8 2017 ..
-rw-rw-r--. 1 ndconf ndconf   9388 Jun  8 2017 backup.xml
> █

```

### 10.1.15 exit

exit

Shutdown the **ncclient**.

### 10.1.16 get

get [path|subtree|xpath <filter>]

Retrieve running configuration and state data from NetConf server. Executing get without parameters retrieves all data. **testagent** supports subtree and xpath filtering. <filter> argument depends on argument preceding it.

```

> get xpath //sessions[Enter]
data
  sessions
    session
      session-id 1
...

```

```

> get subtree "<netconf-state><sessions><session /></sessions></netconf-state>"[Enter]
data
  netconf-state
    sessions
      session
        session-id 1
...

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

Specifying subtree filter in command line is cumbersome. "path" is simpler way to specify subtree filter. E.g. "/netconf-state/sessions/session" is internally translated in "<netconf-state><sessions><session /></sessions></netconf-state>"

```
> get path /netconf-state/sessions/session[Enter]
data
  netconf-state
    sessions
      session
        session-id 1
  ...
```

### 10.1.17 get-config

*get-config candidate|running|startup [path|subtree|xpath <filter>]*

Retrieve all or part of a specified configuration datastore (specified as the 1st argument). If filter is omitted retrieves all configuration data.

Processing of filter is the same as in *get*.

```
> get-config candidate
data
  garage
    garageObjects
      garageAddress 10 Street
      garageCOLevelRisingThreshold 50
      garageCOLevelFallingThreshold 10
    vehiclesEntry
      vehicleIndex 1
  ...
```

### 10.1.18 get-schema

*get-schema identifier [<version> [<format>]]*

Retrieve "schema" from the NetConf server:

```
> get-schema ietf-netconf[Enter]
data module ietf-netconf {

  // the namespace for NETCONF XML definitions is unchanged
  // from RFC 4741 which this document replaces

  namespace "urn:ietf:params:xml:ns:netconf:base:1.0";

  prefix nc;

  import ietf-inet-types {
    prefix inet;
  }

  organization
    "IETF NETCONF (Network Configuration) Working Group";
  ...
```

<version> and <format> are optional.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

<version> is specified as YYYY-MM-DD, e.g. 2011-03-08

The only supported format is "ncm:yang".

### 10.1.19 help

*help <cmdname>*

Print help for command specified in argument.

```
> help disconnect[Enter]
Disconnect from netconf server/agent
> █
```

### 10.1.20 hierarchy

*hierarchy command | config-template*

### 10.1.21 kill-session

*kill-session <session-id>*

Force the termination of a NETCONF session. "snmp" session can not be terminated this way.

```
> kill-session 3[Enter]
ok
> █
```

### 10.1.22 list

*list edit-config | rpc [<file-name>]*

Print content of file from 'edit-config' or 'rpc' folder'. If file name is missing print list of files in corresponding folder.

```
> list edit-config[Enter]
1.xml
2.xml
3.xml
> █
```

If the file name is missing print list of files in corresponding folder.

```
> list edit-config 2[Enter]
<default-operation>merge</default-operation>
<test-option>set</test-option>
<config>
  <t:garage xmlns:t="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB" >
    <t:vehiclesEntry xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:ope
ration="create" >
      <t:vehicleIndex>4</t:vehicleIndex>
      <t:vehicleLicencePlate>777 SWE</t:vehicleLicencePlate>
      <t:vehicleModel>Koenigsegg One:1</t:vehicleModel>
    </t:vehiclesEntry>
  </t:garage>
</config>
> █
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

### 10.1.23 lock

*lock running|candidate|startup*

Lock the entire configuration datastore system of a device.

```
> lock candidate[Enter]
ok
> |
```

### 10.1.24 partial-lock

*partial-lock <xpath>*

Perform a partial lock on a portion of the datastore system of a device, given by **<xpath>**.

```
> partial-lock netconf-state[Enter]
lock-id 100
locked-node /netconf-state
> |
```

### 10.1.25 partial-unlock

*partial-unlock <lock-id>*

Perform an unlock on a portion of the datastore system of a device, given by **<lock-id>**.

```
> partial-unlock 100[Enter]
> |
rpc-reply
ok
> |
```

### 10.1.26 quit

*quit*

Shutdown the **ncclient**.

### 10.1.27 rpc

*rpc <file-name>*

This command reads the file **<file-name>** from the **<SDK installdir>/ndconf/ndconf/ncclient/config/rpc** folder. **<file-name>** can be specified without extension, in that case ".xml" is appended. The file should contain valid netconf rpc request. An example of the content:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<get>
  <filter type="subtree">
    <garage>
      </garage>
    </filter>
  </get>
</rpc>
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



**message-id** is replaced by CLI before passing request to **testagent** for processing.

Assuming content of f1.xml is shown above:

```
> rpc f1[Enter]
garage
  garageObjects
    garageAddress 10 Street
    garageNumVehicles 2
...
```

If file does not exist:

```
> rpc f2[Enter]
Failed to open file config\rpc\f2.xml
> █
```

To display content of the file:

```
> rpc show f0[Enter]
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<get>
<filter type="subtree">
</filter>
</get>
> █
```

### 10.1.28 script

*script <file>*

Execute the specified script file.

### 10.1.29 show

*show history*

Display command line history.

```
> show history[Enter]
0: connect
1: lock candidate
2: unlock candidate
```

### 10.1.30 unlock

*unlock running|candidate|startup*

Lock the entire configuration datastore system of a device.

```
> unlock candidate[Enter]
ok
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
> █
```

### 10.1.31 *validate*

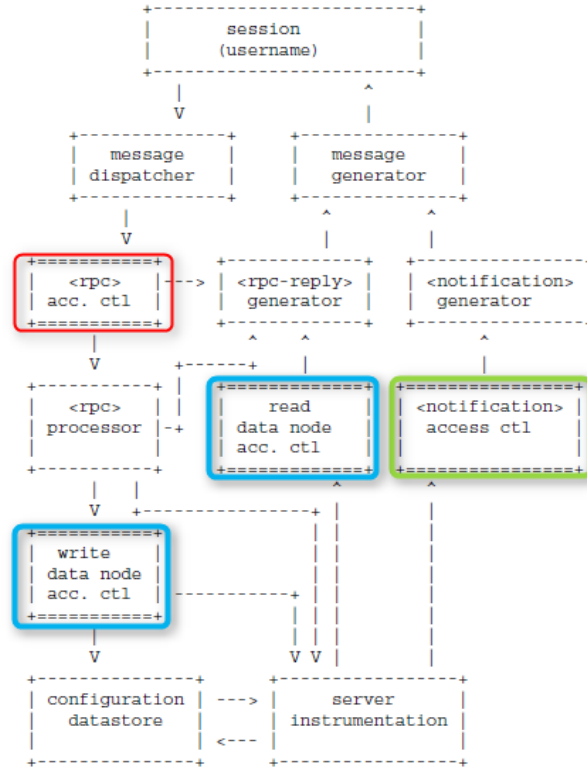
*validate* *running|candidate|startup|<url>|config <filename>*

Validate the contents of the specified configuration.

```
> validate startup[Enter]
source: startup ok
> delete-config startup[Enter]
target: startup
ok
> validate startup[Enter]
source: startup rpc-error
error-type application
error-tag internal-error
error-severity error
error-app-tag general-error
error-message Data store is deleted
> █
```

## 11 NACM - Access Control

Access Control is specified in RFC 6536: *Network Configuration Protocol (NETCONF) Access Control Model*. Here is the diagram from RFC with access control modules highlighted:



**NOTE:** `testagent` is shipped with the "nacm" disabled in the startup configuration. To enable access control, the user should modify the "nacm" section and restart the agent. Also, the user should add **their** user-name to each group. The examples in this document are for user "test".

**NOTE 2:** `testagent` creates a backup of `startup-cfg.xml` before overwriting it. See section "testagent" Command Line Options" for the backup location.

Access control configuration is located under the `nacm` element in `startup-cfg.xml`. Here is the excerpt:

```
<nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
  <enable-nacm>true</enable-nacm>          nacm is enabled
  <read-default>permit</read-default>      if no rule is found allow read and notify
  <write-default>deny</write-default>      if no rule is found deny write operations
  <exec-default>permit</exec-default>      if no rule is found allow execution of rpc
  <groups>
    <group>...</group>
    ... more groups ...
  </groups>
  <rule-list>
    <rule>...</rule>
    ... more rules ...
  </rule-list>
</nacm>
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 11.1 Incoming RPC Message Validation

The following nacm configuration allows execution of all rpcs except "validate" rpc for user "test"

```
<nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
  <enable-nacm>true</enable-nacm>
  <exec-default>permit</exec-default>           default is permit
  <groups>
    <group>
      <name>admin</name>
      <user-name>test</user-name>               replace "test" with your login
    </group>
  </groups>
  <rule-list>
    <name>rule1</name>
    <group>*</group>
    <rule>
      <name>rule1-2</name>
      <module-name>*</module-name>
      <rpc-name>validate</rpc-name>
      <access-operations>*</access-operations>
      <action>deny</action>
    </rule>
  </rule-list>
</nacm>
```

"get" is allowed, "validate" is denied:

```
> get [Enter]
rpc-reply
  data
    snmp
    engine
  ...
> validate candidate [Enter]
rpc-reply
  rpc-error
    error-type application
    error-tag access-denied
    error-severity error
    error-app-tag no-access
    error-path /nc:ietf-netconf/nc:validate
    error-message NC: access denied
> |
```

## 11.2 Outgoing <notification> Authorization

The ndGarageMib .so in **testagent** generates coLevelFallingAlarm and coLevelRisingAlarm notifications in random intervals.

The following nacm configuration allows all outgoing notifications.

```
<nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
  <enable-nacm>true</enable-nacm>
  <read-default>permit</read-default>         default is permit
```

In **ncclient**:

```
> create-subscription [Enter]
>
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

rpc-reply
  ok

notification
  eventTime 2015-12-11T20:06:37Z
  coLevelRisingAlarm          <<<--- "rising"
    object-1
      garageCOLevel 51
    id 4

notification
  eventTime 2015-12-11T20:07:28Z
  coLevelFallingAlarm        <<<--- "falling"
    object-1
      garageCOLevel 9
    id 5
...

```

The following addition to nacm configuration will deny "coLevelRisingAlarm" notification defined in "ND-GARAGE-MIB" YANG module to be sent to session with user "test".

```

<nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
...
  <groups>
    <group>
      <name>admin</name>
      <user-name>test</user-name>          replace "test" with your login
    </group>
  </groups>
  <rule-list>
    <name>rule1</name>
    <group>*</group>
    <rule>
      <name>rule1-2</name>
      <module-name>ND-GARAGE-MIB</module-name>
      <notification-name>coLevelRisingAlarm</notification-name>
      <access-operations>*</access-operations>
      <action>deny</action>
    </rule>
  </rule-list>
</nacm>

```

In **ncclient**, note that "rising" notifications are missing:

```

> create-subscription[Enter]
>
rpc-reply
  ok

notification
  eventTime 2015-12-11T20:12:51Z
  coLevelFallingAlarm        <<<--- "falling"
    object-1
      garageCOLevel 9
    id 5

notification
  eventTime 2015-12-11T20:14:13Z
  coLevelFallingAlarm        <<<--- "falling"
    object-1
      garageCOLevel 9

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
id 7
...
```

### 11.3 Data Node Access Validation

The following nacm configuration allows read access to all objects in datastore.

```
<nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
  <enable-nacm>true</enable-nacm>
  <read-default>permit</read-default> default is permit
```

In **ncclient**:

```
> get [Enter]
data
  snmp
    engine
      enabled true
...
```

Actually "permit" applies to all objects defined without nacm extension "default-deny-all". Definition of "nacm" container in **module ietf-netconf-acm** is:

```
container nacm {
  nacm:default-deny-all;
```

hence, "nacm" objects will be missing from the reply to "get" rpc. Add the following rule to be able to read "nacm" objects:

```
<rule>
  <name>rule1-4</name>
  <module-name>*</module-name>
  <path>//nacm</path>
  <access-operations>read</access-operations>
  <action>permit</action>
</rule>
```

When this rule is present (providing "user-name" is a member of the "group" specified in the "rule-list" to which this "rule" belongs):

```
> get [Enter]
data
  snmp
    engine
      enabled true
...
  nacm
    enable-nacm true
    read-default permit
...
```

Here is the example how to exclude object from the "read" view,

```
<rule>
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

<name>rule1-1</name>
<module-name>ND-GARAGE-MIB</module-name>
<path>//garageAddress</path>
<access-operations>read</access-operations>
<action>deny</action>
</rule>

```

Without the rule above:

```

> get xpath //garageObjects[Enter]
data
  garageObjects
    garageAddress 10 Street
    garageNumVehicles 2
    garageCOLevel 22
    garageCOLevelRisingThreshold 50
    garageCOLevelFallingThreshold 10
>

```

After adding the rule (note that garageAddress is missing):

```

> get xpath //garageObjects[Enter]
data
  garageObjects
    garageNumVehicles 2
    garageCOLevel 22
    garageCOLevelRisingThreshold 50
    garageCOLevelFallingThreshold 10

```

By default write operation is denied:

```

<nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
  <enable-nacm>true</enable-nacm>
  <write-default>deny</write-default>
  <default-is-deny/>
</nacm>

```

Using c1.xml as an example (see 6.1.1 edit-config):

```

<config>
  <t:garage xmlns:t="urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB" >
    <t:garageObjects>
      <t:garageAddress>123 eglinton</t:garageAddress>
    </t:garageObjects>
  </t:garage>
</config>

```

In ncclient:

```

> edit-config candidate c1[Enter]
rpc-error
  error-type application
  error-tag access-denied
  error-severity error
  error-app-tag no-access
  error-path /nc:config-candidate/nc:config/nd-garage:garage
  error-message NC: access denied

```

After adding the following rule:

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

<rule>
  <name>rule1-1</name>
  <module-name>ND-GARAGE-MIB</module-name>
  <path>//garageAddress</path>
  <access-operations>read create update delete</access-operations>
  <action>permit</action>
</rule>

```

edit-config should succeed.

```

> edit-config candidate c1[Enter]
ok

```

## 11.4 Debugging Access Control

Access control configuration can be quite complicated and sometimes hard to find the problem when unexpected result occurs. YangDEMOAgent outputs debug messages while processing access control to the log file. Such lines start with:

**nacmr** - rpc access control

**nacmn** - notification access control

**nacmd** - data access control

The 1st line of access control processing contains session id, user name and rpc/notification or object name. E.g. notification "netconf-session-start" for session id =1, user = "test",

```
nacmn [1:test] netconf-session-start
```

One or more log lines following the 1st line are related to processing that rpc/notification or object name. The last one contains either "permit" or "deny". E.g.

```
nacmn 11 permit
```

Here is the example of multi-line output while processing edit-config:

```

nacmd [2:test] /config-candidate/config/garage/garageObjects/garageAddress
nacmd 6.1, rule=rule1-1, module-match=ND-GARAGE-MIB
nacmd 6.2 A match=/config-candidate/config/garage/garageObjects/garageAddress
nacmd 6.3.2 create
nacmd 7 permit

```

Example of denied rpc:

```

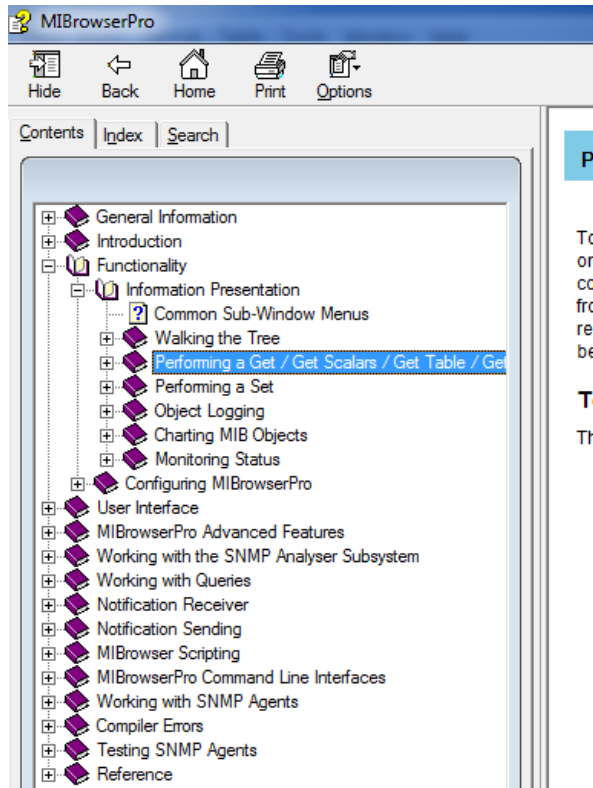
nacmr [2:test] /ietf-netconf/validate
nacmr 8 deny

```



## 12 SNMP

Use MIBBrowser to access **testagent**. Please check MIBBrowser help for more information.



This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 13 Code Generation

### 13.1 Code Generator

Using the NuDesign YANG code generator, the developer generates a C++ project that produces a data dll, from an input YANG document. The code provides a basis for providing instrumentation for the XML tree defined by the input document.

The resulting code is loadable from the NuDesign YANG Data Server. Access to this instrumentation is from the NuDesign Data (Demo) Server via an "Access Agent" (AA). The AA is Dynamic-Link Library also loaded by NuDesign Data (Demo) Server at start up. It is a protocol handler (any of NetConf, SNMP, CLI...).

The code generator itself is a pyang plugin called 'ndt.py'. During installation, the installer checks for the presence of pyang and if it doesn't exist, installs pyang 1.7.5. 'ndt.py' is dependent on version 1.7.5 of pyang.

### 13.2 ndt.py Plugin

#### 13.2.1 'ndt.py' Plugin Command Line Options

The following generation options are available:

-p DIRECTORY,	Where <b>DIRECTORY</b> is a path to a repository of YANG documents. (Note: this is a <b>pyang</b> option, not specific to the NDT YANG Code Generator).
-f FORMAT,	The <b>pyang</b> format selector. FORMAT must be 'ndt'.
--ndt-output=DIRECTORY,	Generate output in <b>DIRECTORY</b> .
--ndt-template=TEMPLATE,	Template directory for the project files. The path should point to the <b>ndt-template</b> directory.
--ndt-help,	Print help on usage of NDT YANG Code Generator and exit.
--ndt-version,	Print the version of NDT YANG Code Generator and exit.
--ndt-verbose,	Verbose mode: Print detailed debug messages.
--ndt-debug,	Print debug messages. Redundant if verbose mode is on.
--ndt-ignore-errors,	Ignore errors from validation.
--ndt-sdkdir=SDKDIR,	Directory of where <b>SDK</b> is installed.
--ndt-hidetimestamps,	Prevents time stamp information being added to the source code files, to make analysis of difference ('diff') on the project files, easier.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

### 13.3 Generating Code with `ndt . py`

To create a project, you may invoke the NDT code generator via `pyang` as follows:

```
pyang <-p path> yang-file -f ndt <ndt options>
```

See the previous sections for the specifics of the various options.

The following is a potential invocation given the default SDK installation directories.

E.g. This example builds a project from the `ND-GARAGE-MIB.yang` found in the installation's `testagent/schema` directory.

(Note: replace `<user>` below with a directory specific to your system.)

```
pyang -p /opt/ndConf-x.y-buildnnn5/ndconf/testagent/schema \
      /opt/ndConf-x.y-buildnnn/ndconf/schema/ND-GARAGE-MIB.yang \
      -f ndt \
      --ndt-output=/home/<user>/src
      --ndt-output=/home/<user>/src \
      --ndt-sdkdir=/opt/ndConf-x.y-buildnnn/ndconf \
      --ndt-template=/opt/ndConf-x.y-buildnnn/ndconf/CodeGenerator/template
```

## 13.4 ndtGen

'`ndtGen`' is a executable that simplifies invocation of '`ndt . py`'.

### 13.4.1 `ndtGen` Command Line Options

The following generation options are available:

- p <path>, An optional parameter. Where **path** is a directory path to a repository of YANG documents. (Note: this is a `pyang` option, not specific to the NDT YANG Code Generator).
- ndt-projschema <path>, An optional parameter. Where **path** is a directory path to a secondary repository, perhaps project specific, YANG documents.
- ndt-template <path>, Optional parameter. Where **path** is a directory path to the template

<sup>5</sup> The default SDK directory location is being denoted as `/opt/ndConf-x.y-buildnnn`. In a typical installation "`x.y`" would be the actual version number, be something like `"1.0"` and "`nnn`" would be the actual build number, something like `"002"`.

repository.

### 13.5 Generating Code with ndtGen

To create a project, you may invoke the NDT code generator via **ndtGen** as follows:

```
ndtGen /opt/ndConf-x.y-build $nnn$ /ndconf/schema/ND-GARAGE-MIB.yang
```

The above command line would generate the ndGarageMib project in the current directory, assuming the same as the previous section, /home/<user>/src. See the man page for ndtGen. For more information.

### 13.6 The Generated Project

From the above example project, the contents of the project would be as follows.

```
Linux/  
    makefile  
    StdAfx.h  
    ndGarageMib_data.cpp  
    ndGarageMib_data_if.h  
    ndGarageMib_notif.cpp  
    ndGarageMib_stx.h  
    m2y_ndGarageMib.xml  
    ndGarageMib_data.h  
    ndGarageMib_impl.cpp  
    ndGarageMib_notif.h  
    ndGarageMib_template.cpp  
    ndGarageMib.cpp  
    ndGarageMib_data_if.cpp  
    ndGarageMib_impl.h  
    ndGarageMib_stx.cpp  
    ndGarageMib_template.h
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 13.7 Building the Project

Change to the ' Linux ' directory of the project and to build it, from type:

```
$ make
```

Additionally, there is a make command line option, MODE. MODE can be used to specify **debug** or **release** builds.

E.g.

```
$ make MODE=release
```

## 13.8 Configuring testagent to use the Project's shared object output

To use the generated code's shared object, it needs to be made available to the NuDesign Data (Demo) Server. This is done via the **testagent's** configuration file,

```
/opt/ndConf-x.y-buildnnn/ndconf/testagent/config/ndconf.xml.
```

In the <datadlls> section,

```
<datadlls>
```

You can disable the provide configuration for **ndGarageMib** by commenting it out.

```
<!--<datadll name="garage" dll="libndGarageMib.so.0" ></datadll-->
```

Now add a new <datadll> section for the generated and built shared object.

```
<datadll name="garage" path="/home/<user>/src/ndGarageMib/Linux"
dll="libndGarageMib.so"></datadll>
```

Note the use of the "path" parameter, to override the default path for shared objects in the SDK.

## 14 Other Resources

### 14.1 Project Design Notes

The installation contains two PDF documents that provide design documentation for the two datadll projects included in the install. Both are installed in the `./doc` directory and are called

- NuDesign ndCONF Builder – CodeGen Tech Reference.pdf
- NuDesign ndCONF Builder – UsingExistingImpl Tech Reference.pdf

### 14.2 Class Reference

The installation contains references for the underlying C++ class libraries. Each class reference is contained in a PDF file. These are installed in the `./doc` directory. Also in the `./doc` directory is an html file called `Main.htm`, which serves as a front end to these PDF files.

### 14.3 man pages

Included with the installation are several Linux 'man' pages. These pages are available from a command line, for the following topics:

- `testagent`
- `teststart`
- `ndncsub`
- `ncclient`
- `mib2xml`
- `ndtGen`<sup>6</sup>

There is one for each of the executables included by the installation.

<sup>6</sup> Only available on the full release SDK.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 15 Products for ndCONF Development

The **ndCONF Agent Development Studio** contains the following products, downloadable from NuDesign's secure site.

### 15.1 ndCONF Builder

The ndCONF Builder generates C++ loadable data handlers (LDH) extensions projects from YANG documents. LDHs are used with the YANG Datastore Server to extend the data model. The server is included with the ndCONF SDK.

The YANG Datastore Server and the ndCONF Builder are included with the ndCONF SDK installation program.

The ndCONF Builder also has its own, standalone installation program that comes with the following documents:

- ndCONF User Guide - this User Guide, with the chapter 13 covering the ndCONF Builder operations
- NuDesign ndCONF Builder – CodeGen Tech Reference.pdf
- NuDesign ndCONF Builder – UsingExistingImpl Tech Reference.pdf

### 15.2 ndCONF SDK

The ndCONF SDK contains the extensible YANG Datastore Server, NETConf / YANG SDKs with basic CLI access and APIs to the protocol Access Agents and LDH extensions, the YANG Datastore's object providers.

Its installation comes with the following document, in addition to the C++ class references identified in section 14.2.

- ndCONF User Guide - this User Guide, it describes the installation and use of ndCONF SDK in a process of developing ndCONF Agent for user selected YANG modules.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

### 15.3 NETCONF, CLI, RESTCONF and SNMP Access Agents

Each protocol Access Agent when combined with YANG Datastore Server provides a full management and configuration Agent providing access to YANG objects via NETCONF, CLI, RESTCONF or SNMP client applications.

The Access Agent's installs come with the C++ class reference documentation identified in section 14.2.



## **16 About NuDesign**

NuDesign Technologies, Inc., headquartered in Toronto, ON, Canada, specializes in providing NetConf, SNMP, CLI and web agent & manager development applications, tools, libraries and consulting services to developers and networking OEMs worldwide. The benefits of deploying NuDesign's management software technologies are reliable, low risk, well integrated and quick-to-market solutions, supporting management operations over IPv4 / IPv6 networks in carrier, campus and enterprise settings.

Our focus is on industry standard management protocols such as NetConf, SNMP and protocols using HTTP transport. Our customers are Original Equipment Manufacturers, System Integrators, Service Providers and End Users worldwide.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 17 Appendix

### 17.1 'ND-GARAGE-MIB' Yang module

```

/*
This Yang module was created using NuDesign Technologies' Visual MIBuilder (Ver 5.2).
MIB File : NDGarageV2.mib
Module : ND-GARAGE-MIB
*/

module ND-GARAGE-MIB {

    namespace "urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB";
    prefix "nd-garage";

    import ietf-yang-types          { prefix "yang"; }
    import ietf-yang-smiv2         { prefix "smiv2"; }
    import SNMPv2-TC               { prefix "snmpv2-tc"; }

    organization
        "NuDesign Technologies, Inc.";
    contact
        "NuDesign Technologies, Inc.
        Web site: www.ndt-inc.com
        Email: contact@ndt-inc.com";
    description
        "The MIB module for managing car garage.";

    revision 2004-07-09 {
        description
            "New version released that is to be used across all NuDesign Products. sh";
    }
    revision 2002-10-11 {
        description
            "Added objects garageCOLevel, garageCOLevelRisingThreshold and
            garageCOLevelFallingThreshold and notifications
            coLevelRisingAlarm and coLevelFallingAlarm.
            Updated notif-groups and compliance. bl";
    }
    revision 2002-01-17 {
        description
            "Modified the access of vehicleModel to ReadCreate. sh";
    }
    revision 2001-03-21 {
        description
            "Addded notifications and conformance statements. bl";
    }
    revision 2000-05-01 {
        description
            "Initial version. bl";
    }

    container garage {
        config false;

        container garageObjects {

            smiv2:oid "1.3.6.1.4.1.4761.99.11.1";
            leaf garageAddress {
                type snmpv2-tc:DisplayString {
                    length "0..32";
                }
            }
        }
    }
}

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

        description
            "Garage address. For example: 10 St, SomePlace.";
        smiv2:max-access "read-write";
        smiv2:oid "1.3.6.1.4.1.4761.99.11.1.1";
    }
    leaf garageNumVehicles {
        type int32;
        description
            "Number of vehicles currently parked in the garage.";
        smiv2:max-access "read-only";
        smiv2:oid "1.3.6.1.4.1.4761.99.11.1.2";
    }
    leaf garageCOLevel {
        type int32;
        description
            "CO toxicity level in garage (airborne concentration level expressed in
            parts per million or PPM).";
        smiv2:max-access "read-only";
        smiv2:oid "1.3.6.1.4.1.4761.99.11.1.4";
    }
    leaf garageCOLevelRisingThreshold {
        type int32;
        description
            "Level of CO toxicity level in garage (in parts per million or PPM)
            considered dangerous for humans. 400 PPM causes serious headache
            after 1-2 hours of exposure. This level is life threatening after 3 hours.
            When the current sampled value is greater than or equal to
            this threshold, and the value at the last sampling interval
            was less than this threshold, a single event (coLevelRisingAlarm)
            will be generated.
            After a rising event is generated, another such event will not be
            generated until the sampled value falls below this threshold and
            reaches the garageCOLevelFallingThreshold.";
        smiv2:max-access "read-write";
        smiv2:oid "1.3.6.1.4.1.4761.99.11.1.5";
    }
    leaf garageCOLevelFallingThreshold {
        type int32;
        description
            "Level of CO toxicity level in garage (in parts per million or PPM)
            considered harmless for humans. 35 PPM is maximum exposure allowed by
            OSHA in the workplace over an eight hour period.
            When the current sampled value is less than or equal to this threshold,
            and the value at the last sampling interval was greater than this threshold,
            a single event (coLevelFallingAlarm) will be generated.
            After a falling event is generated, another such event will not be
            generated until the sampled value rises above this threshold and
            reaches the garageCOLevelRisingThreshold.";
        smiv2:max-access "read-write";
        smiv2:oid "1.3.6.1.4.1.4761.99.11.1.6";
    }
}
/*
container vehiclesTable {
    description
        "Table of vehicles parked in the garage.";
    smiv2:oid "1.3.6.1.4.1.4761.99.11.1.3";
*/
list vehiclesEntry {
    key "vehicleIndex";
    description
        "Row in the vehiclesTable.";

    smiv2:oid "1.3.6.1.4.1.4761.99.11.1.3.1";
}

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

leaf vehicleIndex {
    type int32;
    description
        "Index into vehiclesTable.";
    smiv2:max-access "not-accessible";
    smiv2:oid "1.3.6.1.4.1.4761.99.11.1.3.1.1";
}
leaf vehicleLicencePlate {
    type snmpv2-tc:DisplayString {
        length "6..7";
    }
    description
        "License plate of the vehicle, for example 123XYZ or ABCD987.";
    smiv2:max-access "read-create";
    smiv2:oid "1.3.6.1.4.1.4761.99.11.1.3.1.2";
}
leaf vehicleModel {
    type snmpv2-tc:DisplayString;
    description
        "Make and model of the vehicle; e.g 'NuDesign speedmaster'.
        NuDesign does not manufacture vehicles (yet). This is just
        for demonstration purpose.";
    smiv2:max-access "read-create";
    smiv2:oid "1.3.6.1.4.1.4761.99.11.1.3.1.3";
}
leaf vehicleStatus {
    type snmpv2-tc:RowStatus;
    description
        "Status of this row.";
    smiv2:max-access "read-create";
    smiv2:oid "1.3.6.1.4.1.4761.99.11.1.3.1.4";
}
}
}
/*
}
*/
}

notification notPaidAlarm {
    description
        "The SNMP trap that is generated when the vehicle leaves garage
        but did not pay the bill.";

    smiv2:oid "1.3.6.1.4.1.4761.99.11.2.1";

    container object-1 {
        leaf vehicleIndex {
            type leafref {
                path "/nd-garage-mib:ND-GARAGE-MIB/nd-garage-mib:vehiclesTable" +
                    "/nd-garage-mib:vehiclesEntry/nd-garage-mib:vehicleIndex";
            }
        }
        leaf vehicleLicencePlate {
            type leafref {
                path "/nd-garage-mib:ND-GARAGE-MIB/nd-garage-mib:vehiclesTable" +
                    "/nd-garage-mib:vehiclesEntry/nd-garage-mib:vehicleLicencePlate";
            }
        }
    }
}

notification coLevelRisingAlarm {
    description
        "The SNMP trap that is generated when CO toxicity level in garage

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
        crosses its rising threshold (garageCOLevelRisingThreshold).";
smiv2:oid "1.3.6.1.4.1.4761.99.11.2.2";
container object-1 {
    leaf garageCOLevel {
        type leafref {
            path "/nd-garage-mib:ND-GARAGE-MIB/nd-garage-mib:garageObjects/nd-garage-
mib:garageCOLevel";
        }
    }
}

notification coLevelFallingAlarm {
    description
        "The SNMP trap that is generated when CO toxicity level in garage
        crosses its falling threshold (garageCOLevelFallingThreshold).";

    smiv2:oid "1.3.6.1.4.1.4761.99.11.2.3";

    container object-1 {
        leaf garageCOLevel {
            type leafref {
                path "/nd-garage-mib:ND-GARAGE-MIB/nd-garage-mib:garageObjects/nd-garage-
mib:garageCOLevel";
            }
        }
    }
}

smiv2:alias "nuDesignTech" {
    smiv2:oid "1.3.6.1.4.1.4761";
}

smiv2:alias "ndtExperimental" {
    smiv2:oid "1.3.6.1.4.1.4761.99";
}

smiv2:alias "garage" {
    smiv2:oid "1.3.6.1.4.1.4761.99.11";
}

smiv2:alias "garageObjects" {
    smiv2:oid "1.3.6.1.4.1.4761.99.11.1";
}

smiv2:alias "garageEvents" {
    smiv2:oid "1.3.6.1.4.1.4761.99.11.2";
}

smiv2:alias "garageConformance" {
    smiv2:oid "1.3.6.1.4.1.4761.99.11.3";
}

smiv2:alias "garageCompliances" {
    smiv2:oid "1.3.6.1.4.1.4761.99.11.3.1";
}

smiv2:alias "garageGroups" {
    smiv2:oid "1.3.6.1.4.1.4761.99.11.3.2";
}
}
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

## 17.2 'ND-GARAGE-MIB' MIB module

```

-- *****
-- *
-- * Copyright (c) 2000-2019 NuDesign Technologies, Inc. All Rights Reserved.
-- *
-- * Module Name: ND-GARAGE-MIB
-- *
-- * Description: Defines objects for managing garage for motor vehicles
-- *
-- *****

ND-GARAGE-MIB DEFINITIONS ::= BEGIN

IMPORTS
    NOTIFICATION-TYPE, OBJECT-TYPE, MODULE-IDENTITY,
    enterprises, Integer32
        FROM SNMPv2-SMI
    NOTIFICATION-GROUP, OBJECT-GROUP, MODULE-COMPLIANCE
        FROM SNMPv2-CONF
    RowStatus, DisplayString
        FROM SNMPv2-TC;

garage MODULE-IDENTITY
    LAST-UPDATED "201303140000Z"
    ORGANIZATION
        "NuDesign Technologies, Inc."
    CONTACT-INFO
        "NuDesign Technologies, Inc.
        Web site: www.ndt-inc.com
        Email: contact@ndt-inc.com"
    DESCRIPTION
        "The MIB module for managing car garage."

    REVISION "201303140000Z"
    DESCRIPTION
        "Administrative changes. dp"

    REVISION "200407091226Z"
    DESCRIPTION
        "New version released that is to be used across all NuDesign Products. sh"

    REVISION "200210110900Z"
    DESCRIPTION
        "Added objects garageCOLevel, garageCOLevelRisingThreshold and
        garageCOLevelFallingThreshold and notifications
        coLevelRisingAlarm and coLevelFallingAlarm.
        Updated notif-groups and compliance. bl"

    REVISION "200201171512Z"
    DESCRIPTION
        "Modified the access of vehicleModel to ReadCreate. sh"

    REVISION "200103210000Z"
    DESCRIPTION
        "Adedd notifications and conformance statements. bl"

    REVISION "200005010000Z"
    DESCRIPTION
        "Initial version. bl"
 ::= { ndtExperimental 11 }

nuDesignTech      OBJECT IDENTIFIER ::= { enterprises 4761 }
ndtExperimental  OBJECT IDENTIFIER ::= { nuDesignTech 99 }
garageObjects     OBJECT IDENTIFIER ::= { garage 1 }

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

garageEvents      OBJECT IDENTIFIER ::= { garage 2 }
garageConformance OBJECT IDENTIFIER ::= { garage 3 }

-- Conformance Information

garageCompliances OBJECT IDENTIFIER ::= { garageConformance 1 }
garageGroups      OBJECT IDENTIFIER ::= { garageConformance 2 }

garageAddress OBJECT-TYPE
    SYNTAX      DisplayString (SIZE(0..32))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Garage address. For example: 10 St, SomePlace."
    ::= { garageObjects 1 }

garageNumVehicles OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Number of vehicles currently parked in the garage."
    ::= { garageObjects 2 }

vehiclesTable OBJECT-TYPE
    SYNTAX SEQUENCE OF VehiclesEntry
    MAX-ACCESS not-accessible
    STATUS      current
    DESCRIPTION
        "Table of vehicles parked in the garage."
    ::= { garageObjects 3 }

vehiclesEntry OBJECT-TYPE
    SYNTAX      VehiclesEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Row in the vehiclesTable."
    INDEX { vehicleIndex }
    ::= { vehiclesTable 1 }

VehiclesEntry ::= SEQUENCE {
    vehicleIndex
        Integer32,
    vehicleLicencePlate
        DisplayString,
    vehicleModel
        DisplayString,
    vehicleStatus
        RowStatus
}

vehicleIndex OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Index into vehiclesTable."
    ::= { vehiclesEntry 1 }

vehicleLicencePlate OBJECT-TYPE
    SYNTAX      DisplayString (SIZE(6..7))
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

    "Licence plate of the vehicle, for example 123XYZ or ABCD987."
    ::= { vehiclesEntry 2 }

vehicleModel OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Make and model of the vehicle; e.g 'NuDesign speedmaster'.
        NuDesign does not manufacture vehicles (yet). This is just
        for demonstration purpose."
    ::= { vehiclesEntry 3 }

vehicleStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Status of this row."
    ::= { vehiclesEntry 4 }

garageCOLevel OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "CO toxicity level in garage (airborne concentration level expressed in
        parts per million or PPM)."
    ::= { garageObjects 4 }

garageCOLevelRisingThreshold OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Level of CO toxicity level in garage (in parts per million or PPM)
        considered dangerous for humans. 400 PPM causes serious headache
        after 1-2 hours of exposure. This level is life threatening after 3 hours.
        When the current sampled value is greater than or equal to
        this threshold, and the value at the last sampling interval
        was less than this threshold, a single event (coLevelRisingAlarm)
        will be generated.
        After a rising event is generated, another such event will not be
        generated until the sampled value falls below this threshold and
        reaches the garageCOLevelFallingThreshold."
    ::= { garageObjects 5 }

garageCOLevelFallingThreshold OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Level of CO toxicity level in garage (in parts per million or PPM)
        considered harmless for humans. 35 PPM is maximum exposure allowed by
        OSHA in the workplace over an eight hour period.
        When the current sampled value is less than or equal to this threshold,
        and the value at the last sampling interval was greater than this threshold,
        a single event (coLevelFallingAlarm) will be generated.
        After a falling event is generated, another such event will not be
        generated until the sampled value rises above this threshold and
        reaches the garageCOLevelRisingThreshold."
    ::= { garageObjects 6 }

-- Events

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



```

notPaidAlarm NOTIFICATION-TYPE
  OBJECTS { vehicleLicencePlate }
  STATUS current
  DESCRIPTION
    "The SNMP trap that is generated when the vehicle leaves garage
    but did not pay the bill."
  ::= { garageEvents 1 }

coLevelRisingAlarm NOTIFICATION-TYPE
  OBJECTS { garageCOLevel }
  STATUS current
  DESCRIPTION
    "The SNMP trap that is generated when CO toxicity level in garage
    crosses its rising threshold (garageCOLevelRisingThreshold)."
  ::= { garageEvents 2 }

coLevelFallingAlarm NOTIFICATION-TYPE
  OBJECTS { garageCOLevel }
  STATUS current
  DESCRIPTION
    "The SNMP trap that is generated when CO toxicity level in garage
    crosses its falling threshold (garageCOLevelFallingThreshold)."
  ::= { garageEvents 3 }

-- Compliance statements

garageCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION
    "The compliance statement for agents which
    implement the ND-GARAGE-MIB. Note that garageNotifGroup2 is
    unconditionally optional for compliance to this MIB module."
  MODULE
    MANDATORY-GROUPS { garageBasicGroup,
                       garageNotifGroup1 }
  ::= { garageCompliances 1 }

-- Units of compliance

garageBasicGroup OBJECT-GROUP
  OBJECTS { garageAddress,
            garageNumVehicles,
            vehicleLicencePlate,
            vehicleModel,
            vehicleStatus,
            garageCOLevel,
            garageCOLevelRisingThreshold,
            garageCOLevelFallingThreshold }
  STATUS current
  DESCRIPTION
    "A collection of objects providing the info for the garage."
  ::= { garageGroups 1 }

garageNotifGroup1 NOTIFICATION-GROUP
  NOTIFICATIONS { coLevelRisingAlarm,
                  coLevelFallingAlarm }
  STATUS current
  DESCRIPTION
    "The threshold notifications."
  ::= { garageGroups 2 }

garageNotifGroup2 NOTIFICATION-GROUP
  NOTIFICATIONS { notPaidAlarm }
  STATUS current
  DESCRIPTION

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

    "The not-paid notification."
    ::= { garageGroups 3 }
END

```

```
-- This MIB was created using NuDesign Technology's Visual MIBuilder (Ver 4.4).
```

### 17.3 'RAPID-CITY-MIB' (rcvlan) Yang Module

```

module rcvlan {
    namespace "urn:ietf:params:xml:ns:yang:smiv2:rcvlan";
    prefix "rcv";

    import ietf-yang-smiv2      { prefix "smiv2"; }
    import ietf-inet-types     { prefix "inet"; }
    import SNMPV2-TC          { prefix "snmpv2-tc"; }

    organization
        "Avaya ...";
    contact
        "...";
    description
        "...";

    revision 2016-09-10 {
    }

        typedef InterfaceIndex {
            type int32; // {length "1..2147483647"; }
            description
                "Port IfIndex.";
        }

    typedef PortSet {
        type binary {
            length "0..255";
        }
        description
            "The string is 88 octets long, for a total of 704 bits. Each bit
            corresponds to a port, as represented by its ifIndex value . When a
            bit has the value one(1), the corresponding port is a member of the
            set. When a bit has the value zero(0), the corresponding port is not
            a member of the set. The encoding is such that the most significant

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

    bit of octet #1 corresponds to ifIndex 0, while the least significant
    bit of octet #88 corresponds to ifIndex 703. In order to accommodate
    future uses of this the string has a max size of 255 bytes.";
}

```

```

typedef RcVlanIdOrNone {
    type int32 {
        range "1..4094";
    }
    description
        "Range of VLAN IDs supported for application";
}

```

```

typedef VrfIdentifier {
    type uint32 {
        range "0..512";
    }
    description
        "Virtual Router Identifier.
        VRFID 0 is reserved for the Administrative VRF
        and cannot be used to create VRF's.
        ";
}

```

```

container vlanModule {
    config true;
    //container rcVlanTable {
    //    description
    //        "A list of Virtual LAN entries. The number of entries
    //        is given by rcVlanNumVlans.";
    //    smiv2:oid "1.3.6.1.4.1.2272.1.3.2";

    list vlan { // rcVlanEntry {
        key "vlanId";
        description
            "Entry containing configuration information for a
            particular Virtual LAN.

            The relationship between the various port sets in a

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

VLAN Entry are :

- o The set of ports defined by PortMembers must be a subset of the set of ports in the underlying STG of the VLAN.
  - o The bitwise AND of PortMembers and NotAllowToJoin must be the empty set.
  - o The bitwise OR of PortMembers and NotAllowToJoin must be the set of ports in the underlying STG of the VLAN.
  - o The set of ports defined by StaticMembers must be a subset of the set of ports defined by PortMembers.
  - o The bitwise XOR of PortMembers and StaticMembers defines the set of dynamic (potential) members of the VLAN.
  - o The set of ports defined by ActiveMembers must be a subset of the set of ports defined by PortMembers.
- ";

```
smiv2:oid "1.3.6.1.4.1.2272.1.3.2.1";
```

```
leaf vlanId { // rcVlanId {
  type rcv:RcVlanIdOrNone;
  description
    "A value that uniquely identifies the Virtual LAN
    associated with this entry. This value corresponds
    to the lower 12 bits in the IEEE 802.1Q VLAN Tag.";
  smiv2:max-access "read-only";
  smiv2:oid "1.3.6.1.4.1.2272.1.3.2.1.1";
}
leaf vlanName { // rcVlanName {
  type snmpv2-tc:DisplayString {
    length "0..64";
  }
  description
    "An administratively-assigned name for this VLAN.";
```

```

    smiv2:max-access "read-write";
    smiv2:oid "1.3.6.1.4.1.2272.1.3.2.1.2";
}
leaf stgId { // rcVlanStgId {
    type int32 {
        range "0..128";
    }
    description
        "Indicates the Spanning Tree Group (STG) used by
        this VLAN to determine the state of its ports.
        If this VLAN is not associated with any STG, this
        value should be set to zero.";
    smiv2:defval "1";
    smiv2:max-access "read-write";
    smiv2:oid "1.3.6.1.4.1.2272.1.3.2.1.9";
}
leaf vlanType { // rcVlanType {
    type enumeration {
        enum "byPort" { value "1"; }
        enum "byIpSubnet" { value "2"; }
        enum "byProtocolId" { value "3"; }
        enum "bySrcMac" { value "4"; }
        enum "byDstMcast" { value "5"; }
        enum "bySvlan" { value "6"; }
        enum "byIds" { value "7"; }
        enum "spbm-bvlan" { value "11"; }
        enum "private" { value "13"; }
    }
    description
        "The type of VLAN, distinguished according to the
        policy used to define its port membership.
        VSP9000 does not support bySvlan(6) and ByIds(7).";
    smiv2:max-access "read-write";
    smiv2:oid "1.3.6.1.4.1.2272.1.3.2.1.10";
}
leaf vlanPortMembers { // rcVlanPortMembers {
    type rcv:PortSet;
    description
        "The set of ports that are members (static or

```

```

        dynamic) of this VLAN.";
    smiv2:max-access "read-write";
    smiv2:oid "1.3.6.1.4.1.2272.1.3.2.1.11";
}
leaf vlanProtocolId { // rcVlanProtocolId {
    type enumeration {
        enum "none" { value "0"; }
        enum "ip" { value "1"; }
        enum "ipx802dot3" { value "2"; }
        enum "ipx802dot2" { value "3"; }
        enum "ipxSnap" { value "4"; }
        enum "ipxEthernet2" { value "5"; }
        enum "appleTalk" { value "6"; }
        enum "declat" { value "7"; }
        enum "decOther" { value "8"; }
        enum "sna802dot2" { value "9"; }
        enum "snaEthernet2" { value "10"; }
        enum "netBios" { value "11"; }
        enum "xns" { value "12"; }
        enum "vines" { value "13"; }
        enum "ipV6" { value "14"; }
        enum "usrDefined" { value "15"; }
        enum "rarp" { value "16"; }
        enum "pPpOE" { value "17"; }
    }
    description
        "The protocol identifier of this VLAN. This value
        is meaningful only if rcVlanType is equal to
        byProtocolId(3). For other VLAN types it should
        have the value none(0).";
    smiv2:max-access "read-write";
    smiv2:oid "1.3.6.1.4.1.2272.1.3.2.1.15";
}

list ip { // rcIpAddrTable {
    key "addr";
    description
        "The table of addressing information relevant to
        this entity's IP addresses.";
}

```

```
leaf ifIndex {
    type rcv:InterfaceIndex;
    description
        "The index value which uniquely identifies the
        interface to which this entry is applicable. The
        interface identified by a particular value of this
        index is the same interface as identified by the
        same value of ifIndex.";
    smiv2:max-access "read-only";
    smiv2:oid "1.3.6.1.4.1.2272.1.8.2.1.1";
}
leaf addr { // rcIpAdEntAddr {
    type inet:ipv4-address;
    description
        "The IP address to which this entry's addressing
        information pertains.";
    smiv2:max-access "read-only";
    smiv2:oid "1.3.6.1.4.1.2272.1.8.2.1.2";
}
leaf netMask { // rcIpAdEntNetMask {
    type inet:ipv4-address;
    description
        "The subnet mask associated with the IP address of
        this entry. The value of the mask is an IP
        address with all the network bits set to 1 and all
        the hosts bits set to 0.";
    smiv2:max-access "read-write";
    smiv2:oid "1.3.6.1.4.1.2272.1.8.2.1.3";
}
leaf bcastAddrFormat{ // rcIpAdEntBcastAddrFormat {
    type enumeration {
        enum "zeros" { value "0"; }
        enum "ones" { value "1"; }
    }
    description
        "The IP broadcast address format used on this
        interface.";
    smiv2:max-access "read-only";
```

```

        smiv2:oid "1.3.6.1.4.1.2272.1.8.2.1.4";
    }
    leaf reasmMaxSize { // rcIpAdEntReasmMaxSize {
        type int32 {
            range "0..65535";
        }
        description
            "The size of the largest IP datagram which this
            entity can re-assemble from incoming IP fragmented
            datagrams received on this interface.";
        smiv2:max-access "read-only";
        smiv2:oid "1.3.6.1.4.1.2272.1.8.2.1.5";
    }
    /*
    leaf rcIpAdEntVlanId {
        type int32 {
            range "0..4096";
        }
        description
            "A value that uniquely identifies the Virtual LAN
            associated with this entry. This value corresponds
            to the lower 12 bits in the IEEE 802.1Q VLAN Tag.";
        smiv2:max-access "read-write";
        smiv2:oid "1.3.6.1.4.1.2272.1.8.2.1.7";
    }
    */
    leaf macOffset { // rcIpAdEntMacOffset {
        type int32 {
            range "0..511";
        }
        description
            "Used to translate the ip address into mac address.
            The system has 512 mac addresses of which 0-507 are
            reserved for
            Boardwalk box. 508-511 is reserved for MG. One can either
            mention a mac offset while configuring an ip on the
            range.";
            vlan or it can be allotted by the system within the above

        smiv2:max-access "read-write";
        smiv2:oid "1.3.6.1.4.1.2272.1.8.2.1.9";
    }

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.





Postal: Bay Networks, Inc.  
4401 Great America Parkway  
Santa Clara, CA 95052-8185

Tel: 408-495-6159

Fax: 408-495-5215

E-mail:edwin\_tsang@baynetworks.com

"

DESCRIPTION

"Enterprise MIB for the Accelar product family."

::= { enterprises 2272 }

-- May 15, 2012

--

-- Local defines to avoid having to pull-in other RFC's.

--

BridgeId ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"An identifier that identifies the Bridge Id"

SYNTAX OCTET STRING (SIZE(8))

EnableValue ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Enable/Disable value."

SYNTAX INTEGER {

enable (1),

disable (2)

}

IdList ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"An identifier for a list of Ids."

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
SYNTAX OCTET STRING
```

```
--
```

```
-- This variable type is used through out the Rapid-City enterprise
```

```
-- MIB to denote the standard ifIndex in mib-2.
```

```
--
```

```
InterfaceIndex ::= TEXTUAL-CONVENTION
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "Port IfIndex."
```

```
    SYNTAX      Integer32
```

```
InterfaceIndexOrZero ::= TEXTUAL-CONVENTION
```

```
    DISPLAY-HINT "d"
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "This textual convention is an extension of the
```

```
        InterfaceIndex convention. The latter defines a greater
        than zero value used to identify an interface or interface
        sub-layer in the managed system. This extension permits the
        additional value of zero. the value zero is object-specific
        and must therefore be defined as part of the description of
        any object which uses this syntax. Examples of the usage of
        zero might include situations where interface was unknown,
        or when none or all interfaces need to be referenced."
```

```
    SYNTAX      Integer32 (0..2147483647)
```

```
IpIIsbPlsbNodeNickName ::= TEXTUAL-CONVENTION
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "Plsb Node Nick name."
```

```
    SYNTAX      OCTET STRING (SIZE(3))
```

```
Ipv6NextHdr ::= TEXTUAL-CONVENTION
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

STATUS    current
DESCRIPTION
    "The ipv6 next header to look for."

```

```

SYNTAX INTEGER {
    hop-by-hop (0),
    icmpv4 (1),
    igmpv4 (2),
    ipInIp (4),
    tcp (6),
    egp (8),
    udp (17),
    ipv6 (41),
    routing (43),
    frag (44),
    rsvp (46),
    ipsecESP (50),
    ipsecAh (51),
    icmpv6 (58),
    noNxtHdr (59),
    destOptions (60),
    undefined (255)
}

```

```

--
-- The string is 88 octets long, for a total of 704 bits. Each bit
-- corresponds to a port, as represented by its ifIndex value . When a
-- bit has the value one(1), the corresponding port is a member of the
-- set. When a bit has the value zero(0), the corresponding port is not
-- a member of the set. The encoding is such that the most significant
-- bit of octet #1 corresponds to ifIndex 0, while the least significant
-- bit of octet #88 corresponds to ifIndex 703."
--

```

```
PortSet ::= TEXTUAL-CONVENTION
```

```

STATUS    current
DESCRIPTION
    "The string is 88 octets long, for a total of 704 bits. Each bit
    corresponds to a port, as represented by its ifIndex value . When a

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

bit has the value one(1), the corresponding port is a member of the set. When a bit has the value zero(0), the corresponding port is not a member of the set. The encoding is such that the most significant bit of octet #1 corresponds to ifIndex 0, while the least significant bit of octet #88 corresponds to ifIndex 703. In order to accommodate future uses of this the string has a max size of 255 bytes."

SYNTAX OCTET STRING (SIZE(0..255))

RcLongDisplayString ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The semantics are identical to the standard DisplayString in RFC1213 and RFC2579, except for the longer length that is allowed with this TC. "

SYNTAX OCTET STRING (SIZE(0..65535))

RcVlanIdOrNone ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Range of VLAN IDs supported for application"

SYNTAX Integer32 (1..4094)

Timeout ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"An identifier that identifies the timeout value."

SYNTAX Integer32

VrfIdentifier ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Virtual Router Identifier.

VRFID 0 is reserved for the Administrative VRF and cannot be used to create VRF's.

"

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
SYNTAX Unsigned32 (0..512)

org                OBJECT IDENTIFIER ::= { iso 3 }
dod                OBJECT IDENTIFIER ::= { org 6 }

--
-- Additions to resolve some SNMP V2 dependencies so we don't
-- have to pull-in a lot of mib modules.
--

internet          OBJECT IDENTIFIER ::= { dod 1 }
snmpV2            OBJECT IDENTIFIER ::= { internet 6 }
snmpModules       OBJECT IDENTIFIER ::= { snmpV2 3 }
snmpMIB           OBJECT IDENTIFIER ::= { snmpModules 1 }
snmpMIBObjects    OBJECT IDENTIFIER ::= { snmpMIB 1 }
snmpTraps         OBJECT IDENTIFIER ::= { snmpMIBObjects 5 }

--
-- Enterprise specific MIB groups
--

rcMgmt            OBJECT IDENTIFIER ::= { rapidCity 1 }

-- APLS device end

rcSystem          OBJECT IDENTIFIER ::= { rcMgmt 1 }
rcTftp            OBJECT IDENTIFIER ::= { rcMgmt 2 }
rcVlan            OBJECT IDENTIFIER ::= { rcMgmt 3 }
rcIp              OBJECT IDENTIFIER ::= { rcMgmt 8 }

-- VLAN Table

rcVlanNumVlans   OBJECT-TYPE
    SYNTAX      Integer32 (1..128)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of VLANs currently defined in the switch."
    ::= { rcVlan 1 }
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

**rcVlanTable OBJECT-TYPE**

SYNTAX SEQUENCE OF RcVlanEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"A list of Virtual LAN entries. The number of entries  
is given by rcVlanNumVlans."

::= { rcVlan 2 }

**rcVlanEntry OBJECT-TYPE**

SYNTAX RcVlanEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"Entry containing configuration information for a  
particular Virtual LAN.

The relationship between the various port sets in a  
VLAN Entry are :

- o The set of ports defined by PortMembers must be a subset of the set of ports in the underlying STG of the VLAN.
- o The bitwise AND of PortMembers and NotAllowToJoin must be the empty set.
- o The bitwise OR of PortMembers and NotAllowToJoin must be the set of ports in the underlying STG of the VLAN.
- o The set of ports defined by StaticMembers must be a subset of the set of ports defined by PortMembers.
- o The bitwise XOR of PortMembers and StaticMembers defines the set of dynamic (potential) members of the VLAN.
- o The set of ports defined by ActiveMembers must be a subset of the set of ports defined by PortMembers.

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

"

```
INDEX { rcVlanId }  
 ::= { rcVlanTable 1 }
```

```
RcVlanEntry ::= SEQUENCE {  
    rcVlanId  
        RcVlanIdOrNone,  
    rcVlanName  
        DisplayString,  
    rcVlanColor  
        Integer32,  
    rcVlanHighPriority  
        TruthValue,  
    rcVlanRoutingEnable  
        TruthValue,  
    rcVlanIfIndex  
        InterfaceIndex,  
    rcVlanAction  
        INTEGER,  
    rcVlanResult  
        INTEGER,  
    rcVlanStgId  
        Integer32,  
    rcVlanType  
        INTEGER,  
    rcVlanPortMembers  
        PortSet,  
    rcVlanActiveMembers  
        PortSet,  
    rcVlanStaticMembers  
        PortSet,  
    rcVlanNotAllowToJoin  
        PortSet,  
    rcVlanProtocolId  
        INTEGER,  
    rcVlanSubnetAddr  
        IpAddress,  
    rcVlanSubnetMask  
        IpAddress,
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



```
rcVlanAgingTime
    Integer32,
rcVlanMacAddress
    MacAddress,
rcVlanRowStatus
    RowStatus
}
```

**rcVlanId OBJECT-TYPE**

SYNTAX RcvlanIdOrNone

MAX-ACCESS read-only

STATUS current

**DESCRIPTION**

"A value that uniquely identifies the Virtual LAN associated with this entry. This value corresponds to the lower 12 bits in the IEEE 802.1Q VLAN Tag."

::= { rcVlanEntry 1 }

**rcVlanName OBJECT-TYPE**

SYNTAX DisplayString (SIZE(0..64))

MAX-ACCESS read-write

STATUS current

**DESCRIPTION**

"An administratively-assigned name for this VLAN."

::= { rcVlanEntry 2 }

**rcVlanColor OBJECT-TYPE**

SYNTAX Integer32 (0..32)

MAX-ACCESS read-write

STATUS current

**DESCRIPTION**

"An administratively-assigned color code for this VLAN. The value of this object is used by the VLAN Manager GUI tool to select a color when it draws this VLAN on the screen."

::= { rcVlanEntry 3 }

**rcVlanHighPriority OBJECT-TYPE**

SYNTAX TruthValue

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
MAX-ACCESS read-write
STATUS      obsolete
DESCRIPTION
    "A flag to note whether frames in this VLAN should
        be assigned a high switching priority."
DEFVAL { false }
::= { rcVlanEntry 4 }
```

```
rcVlanRoutingEnable OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS read-write
STATUS      obsolete
DESCRIPTION
    "A flag to note whether IP routing is enabled in
        this VLAN."
DEFVAL { false }
::= { rcVlanEntry 5 }
```

```
rcVlanIfIndex OBJECT-TYPE
SYNTAX      InterfaceIndex
MAX-ACCESS read-only
STATUS      current
DESCRIPTION
    "When rcVlanRoutingEnable is set to true(1), this
        value indicates the 'logical' ifIndex assigned to
        this VLAN. Otherwise, this value is meaningless
        and should be set to zero."
::= { rcVlanEntry 6 }
```

```
rcVlanAction OBJECT-TYPE
SYNTAX      INTEGER {
    none (1),
    flushMacFdb (2),
    flushArp (3),
    flushIp (4),
    flushDynMemb (5),
    all (6),
    flushSnoopMemb (7),
    triggerRipUpdate (8),
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
        flushSnoopMRtr (9),
        flushIpRsmItEdgePeer (10)
    }
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "VLAN related actions."
    DEFVAL { none }
    ::= { rcVlanEntry 7 }
```

```
rcVlanResult OBJECT-TYPE
    SYNTAX INTEGER {
        none (1),
        inProgress (2),
        success (3),
        fail (4)
    }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The result from the last VLAN action."
    DEFVAL { none }
    ::= { rcVlanEntry 8 }
```

```
rcVlanStgId OBJECT-TYPE
    SYNTAX Integer32 (0..128)
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Indicates the Spanning Tree Group (STG) used by
        this VLAN to determine the state of its ports.
        If this VLAN is not associated with any STG, this
        value should be set to zero."
    DEFVAL { 1 }
    ::= { rcVlanEntry 9 }
```

```
rcVlanType OBJECT-TYPE
    SYNTAX INTEGER {
        byPort (1),
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

        byIpSubnet (2),
        byProtocolId (3),
        bySrcMac (4),
        byDstMcast (5),
        bySvlan (6),
        byIds (7),
        spbm-bvlan (11),
        private (13)
    }
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "The type of VLAN, distinguished according to the
    policy used to define its port membership.
    VSP9000 does not support bySvlan(6) and ByIds(7)."
```

::= { rcVlanEntry 10 }

```
rcVlanPortMembers OBJECT-TYPE
SYNTAX PortSet
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "The set of ports that are members (static or
    dynamic) of this VLAN."
```

::= { rcVlanEntry 11 }

```
rcVlanActiveMembers OBJECT-TYPE
SYNTAX PortSet
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "The set of ports that are currently active in
    this VLAN. Active ports include all static ports
    and any dynamic ports where the VLAN policy was
    met."
```

::= { rcVlanEntry 12 }

```
rcVlanStaticMembers OBJECT-TYPE
SYNTAX PortSet
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The set of ports that are static members of this  
VLAN. A static member of a VLAN is always active  
and is never aged out."

::= { rcVlanEntry 13 }

rcVlanNotAllowToJoin OBJECT-TYPE

SYNTAX PortSet

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The set of ports that are not allowed to become  
members of this VLAN."

::= { rcVlanEntry 14 }

rcVlanProtocolId OBJECT-TYPE

SYNTAX INTEGER {

none (0),  
ip (1),  
ipx802dot3 (2),  
ipx802dot2 (3),  
ipxSnap (4),  
ipxEthernet2 (5),  
appleTalk (6),  
decLat (7),  
decOther (8),  
sna802dot2 (9),  
snaEthernet2 (10),  
netBios (11),  
xns (12),  
vines (13),  
ipV6 (14),  
usrDefined (15),  
rarp (16),  
pPPoE (17)

}

MAX-ACCESS read-write

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
STATUS    current
DESCRIPTION
    "The protocol identifier of this VLAN. This value
        is meaningful only if rcVlanType is equal to
        byProtocolId(3). For other VLAN types it should
        have the value none(0)."
```

```
::= { rcVlanEntry 15 }
```

```
rcVlanSubnetAddr OBJECT-TYPE
SYNTAX     IPAddress
MAX-ACCESS read-write
STATUS     current
DESCRIPTION
    "The IP subnet address of this VLAN. This value
        is meaningful only if rcVlanType is equal to
        byIpSubnet(2). For other VLAN types it should
        have the value 0.0.0.0."
```

```
::= { rcVlanEntry 16 }
```

```
rcVlanSubnetMask OBJECT-TYPE
SYNTAX     IPAddress
MAX-ACCESS read-write
STATUS     current
DESCRIPTION
    "The IP subnet mask of this VLAN. This value
        is meaningful only if rcVlanType is equal to
        byIpSubnet(2). For other VLAN types it should
        have the value 0.0.0.0."
```

```
::= { rcVlanEntry 17 }
```

```
rcVlanAgingTime OBJECT-TYPE
SYNTAX     Integer32 (0|10..1000000)
MAX-ACCESS read-write
STATUS     current
DESCRIPTION
    "The timeout period (in seconds) used for aging
        out dynamic members of this VLAN. This field is
        only relevant for policy-based VLANs."
```

```
DEFVAL { 600 }
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
::= { rcVlanEntry 18 }
```

rcVlanMacAddress OBJECT-TYPE

SYNTAX MacAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The MAC address assigned to the virtual router interface of this VLAN. This field is meaningful only if rcVlanRoutingEnable is equal to true(1)."

```
::= { rcVlanEntry 19 }
```

rcVlanRowStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Used to create/delete entries in the rcVlanTable."

```
::= { rcVlanEntry 20 }
```

-- IP Address Table

rcIpAddrTable OBJECT-TYPE

SYNTAX SEQUENCE OF RcIpAddrEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table of addressing information relevant to this entity's IP addresses.

This table is identical to the ipAddrTable in MIB2 except the columns rcIpAdEntIfIndex and rcIpAdEntAddr are reversed. Also, the table is indexed by both these variables."

```
::= { rcIp 2 }
```

rcIpAddrEntry OBJECT-TYPE

SYNTAX RcIpAddrEntry

MAX-ACCESS not-accessible

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

STATUS    current
DESCRIPTION
    "The addressing information for one of this
                                     entity's IP addresses."

```

```

INDEX { rcIpAdEntIfIndex,
        rcIpAdEntAddr }
::= { rcIpAddrTable 1 }

```

```

RcIpAddrEntry ::= SEQUENCE {
    rcIpAdEntIfIndex
        InterfaceIndex,
    rcIpAdEntAddr
        IpAddress,
    rcIpAdEntNetMask
        IpAddress,
    rcIpAdEntBcastAddrFormat
        INTEGER,
    rcIpAdEntReasmMaxSize
        Integer32,
    rcIpAdEntRowStatus
        RowStatus,
    rcIpAdEntVlanId
        Integer32,
    rcIpAdEntRouterPort
        TruthValue,
    rcIpAdEntMacOffset
        Integer32,
    rcIpAdEntIfType
        INTEGER,
    rcIpAdEntVrfId
        VrfIdentifier
}

```

```
rcIpAdEntIfIndex OBJECT-TYPE
```

```
SYNTAX    InterfaceIndex
```

```
MAX-ACCESS read-only
```

```
STATUS    current
```

```
DESCRIPTION
```

```
    "The index value which uniquely identifies the
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.



interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex."

```
::= { rcIpAddrEntry 1 }
```

rcIpAdEntAddr OBJECT-TYPE

SYNTAX IPAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The IP address to which this entry's addressing information pertains."

```
::= { rcIpAddrEntry 2 }
```

rcIpAdEntNetMask OBJECT-TYPE

SYNTAX IPAddress

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The subnet mask associated with the IP address of this entry. The value of the mask is an IP address with all the network bits set to 1 and all the hosts bits set to 0."

```
::= { rcIpAddrEntry 3 }
```

rcIpAdEntBcastAddrFormat OBJECT-TYPE

SYNTAX INTEGER {

zeros (0),

ones (1)

}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The IP broadcast address format used on this interface."

```
::= { rcIpAddrEntry 4 }
```

rcIpAdEntReasmMaxSize OBJECT-TYPE

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
SYNTAX      Integer32 (0..65535)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The size of the largest IP datagram which this
    entity can re-assemble from incoming IP fragmented
    datagrams received on this interface."
 ::= { rcIpAddrEntry 5 }
```

```
rcIpAdEntRowStatus  OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "Used to create/delete entries"
 ::= { rcIpAddrEntry 6 }
```

```
rcIpAdEntVlanId  OBJECT-TYPE
SYNTAX      Integer32 (0..4096)
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "A value that uniquely identifies the Virtual LAN
    associated with this entry. This value corresponds
    to the lower 12 bits in the IEEE 802.1Q VLAN Tag."
 ::= { rcIpAddrEntry 7 }
```

```
rcIpAdEntBrouterPort  OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Used to indicate whether this entry correponds
    to a brouter port (as oppose to a routable VLAN).
    This value cannot be changed after the row is
    created."
 ::= { rcIpAddrEntry 8 }
```

```
rcIpAdEntMacOffset  OBJECT-TYPE
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

SYNTAX Integer32 (0..511)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Used to translate the ip address into mac address.

The system has 512 mac addresses of which 0-507 are reserved for Boardwalk box. 508-511 is reserved for MG. One can either mention a mac offset while configuring an ip on the vlan or it can be allotted by the system within the above range."

::= { rcIpAddrEntry 9 }

rcIpAdEntIfType OBJECT-TYPE

SYNTAX INTEGER {  
circuitLessIP (1),  
brouterPort (2),  
vlan (3)

}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The address entry IfType indicates the type of the interface. In the case of circuit-less-IP there is

no association with any ports. In other words, it can be thought of as a virtual interface. The value 'brouter port' indicates this interface is associated with a specific physical port. The value 'vlan' indicates a logical port that contains one or physical ports/MLT ports."

::= { rcIpAddrEntry 10 }

rcIpAdEntVrfId OBJECT-TYPE

SYNTAX VrfIdentifier

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The VrfId used in this table is to associate VLANs or Brouter ports to a VRF after the creation of VLANs or Brouter ports.

VRFID 0 is reserved for the Administrative VRF."

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```
 ::= { rcIpAddrEntry 12 }
```

```
-- FlowControl Group
```

```
rcIpFlowTable OBJECT-TYPE
```

```
SYNTAX SEQUENCE OF RcIpFlowEntry
```

```
MAX-ACCESS not-accessible
```

```
STATUS current
```

```
DESCRIPTION
```

```
    "IpFlow Table. This table contain entries that
      correspond to RSVP records in the switching
      fabric Address Resolution table.
```

```

      A connection that has a matching entry in this
      table will be processed with a higher priority
      than connections that do not have an entry."
```

```
 ::= { rcIp 3 }
```

```
rcIpFlowEntry OBJECT-TYPE
```

```
SYNTAX RcIpFlowEntry
```

```
MAX-ACCESS not-accessible
```

```
STATUS current
```

```
DESCRIPTION
```

```
    "Proprietary flow control variables."
```

```
INDEX { rcIpFlowSrcIpAddress,
        rcIpFlowSrcIpPort,
        rcIpFlowDstIpAddress,
        rcIpFlowDstIpPort,
        rcIpFlowProtocol }
```

```
 ::= { rcIpFlowTable 1 }
```

```
RcIpFlowEntry ::= SEQUENCE {
```

```
    rcIpFlowSrcIpAddress
```

```
        IpAddress,
```

```
    rcIpFlowSrcIpPort
```

```
        Integer32,
```

```
    rcIpFlowDstIpAddress
```

```
        IpAddress,
```

```
    rcIpFlowDstIpPort
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

```

        Integer32,
rcIpFlowProtocol
        INTEGER,
rcIpFlowRowStatus
        RowStatus
}

```

```

rcIpFlowSrcIpAddress OBJECT-TYPE
    SYNTAX      IPAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The source IP address of an IP packet."
    ::= { rcIpFlowEntry 1 }

```

```

rcIpFlowSrcIpPort OBJECT-TYPE
    SYNTAX      Integer32 (0..65535)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The source port of an IP packet. A zero value
        in this field is used as a wildcard value."
    DEFVAL { 0 }
    ::= { rcIpFlowEntry 2 }

```

```

rcIpFlowDstIpAddress OBJECT-TYPE
    SYNTAX      IPAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The destination IP address of an IP packet."
    ::= { rcIpFlowEntry 3 }

```

```

rcIpFlowDstIpPort OBJECT-TYPE
    SYNTAX      Integer32 (0..65535)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The destination port of an IP packet. A zero

```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

value in this field is used as a wildcard value."

```
DEFVAL { 0 }
::= { rcIpFlowEntry 4 }
```

rcIpFlowProtocol OBJECT-TYPE

```
SYNTAX INTEGER {
    ip (4),
    tcp (6),
    udp (17)
}
```

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The protocol type of an IP packet. A zero value  
in this field is used as a wildcard value."

```
DEFVAL { ip }
::= { rcIpFlowEntry 5 }
```

rcIpFlowRowStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Status of entry."

```
::= { rcIpFlowEntry 6 }
```

END

## 17.5 Capabilities

The following capabilities are available in the ndCONF server:

```
capability urn:ietf:params:netconf:base:1.0
capability urn:ietf:params:netconf:base:1.1
capability urn:ietf:params:netconf:capability:writable-running:1.0
capability urn:ietf:params:netconf:capability:candidate:1.0
capability urn:ietf:params:netconf:capability:confirmed-commit:1.0
capability urn:ietf:params:netconf:capability:confirmed-commit:1.1
capability urn:ietf:params:netconf:capability:rollback-on-error:1.0
capability urn:ietf:params:netconf:capability:validate:1.0
capability urn:ietf:params:netconf:capability:validate:1.1
capability urn:ietf:params:netconf:capability:startup:1.0
capability urn:ietf:params:netconf:capability:url:1.0
capability urn:ietf:params:netconf:capability:xpath:1.0
```

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.

capability urn:ietf:params:netconf:capability:notification:1.0  
capability urn:ietf:params:netconf:capability:interleave:1.0  
capability urn:ietf:params:netconf:capability:partial-lock:1.0  
capability urn:ietf:params:xml:ns:yang:ietf-inet-types?module=ietf-inet-types&revision=2013-07-15  
capability urn:ietf:params:xml:ns:yang:ietf-yang-types?module=ietf-yang-types&revision=2013-07-15  
capability urn:ietf:params:xml:ns:yang:ietf-yang-smiv2?module=ietf-yang-smiv2&revision=2012-06-22  
capability urn:ietf:params:xml:ns:yang:smiv2:SNMPv2-TC?module=SNMPv2-TC&revision=1999-04-01  
capability urn:ietf:params:xml:ns:yang:ietf-netconf-notifications?module=ietf-netconf-notifications&revision=2011-08-07  
capability urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring?module=ietf-netconf-monitoring&revision=2010-10-04  
capability urn:ietf:params:xml:ns:yang:ietf-netconf-acm?module=ietf-netconf-acm&revision=2011-10-04  
capability http://ndt-inc.com/ns/ndt-mgmt?module=ndt-mgmt&revision=2014-04-01  
capability urn:ietf:params:xml:ns:netconf:notification:1.0?module=notifications&revision=2008-07-14  
capability urn:ietf:params:xml:ns:netmod:notification?module=nc-notifications&revision=2008-07-14  
capability urn:ietf:params:xml:ns:yang:ietf-snmp?module=ietf-snmp&revision=2014-12-10  
capability urn:ietf:params:xml:ns:yang:smiv2:SNMPv2-MIB?module=SNMPv2-MIB&revision=2002-10-16  
capability urn:ietf:params:xml:ns:yang:ietf-interfaces?module=ietf-interfaces&revision=2014-05-08  
capability urn:ietf:params:xml:ns:yang:iana-if-type?module=iana-if-type&revision=2014-05-08  
capability urn:ietf:params:xml:ns:yang:smiv2:ND-GARAGE-MIB?module=ND-GARAGE-MIB&revision=2004-07-09  
capability urn:ietf:params:xml:ns:yang:smiv2:rcvlan?module=rcvlan&revision=2016-09-10

This manual and associated NuDesign Software are provided on an "As Is" basis. By using them, you are consenting to the Terms of the "NuDesign Software License Agreement" associated with the Product or Product Evaluation, that you licensed from NuDesign. This manual may not be copied.