

4/11/2013



GILAT

## NGNMS – AUTOMATED TESTING

Usage examples for: `wptest.py` | Vitalie Ghelbert - Moldova

## Contents

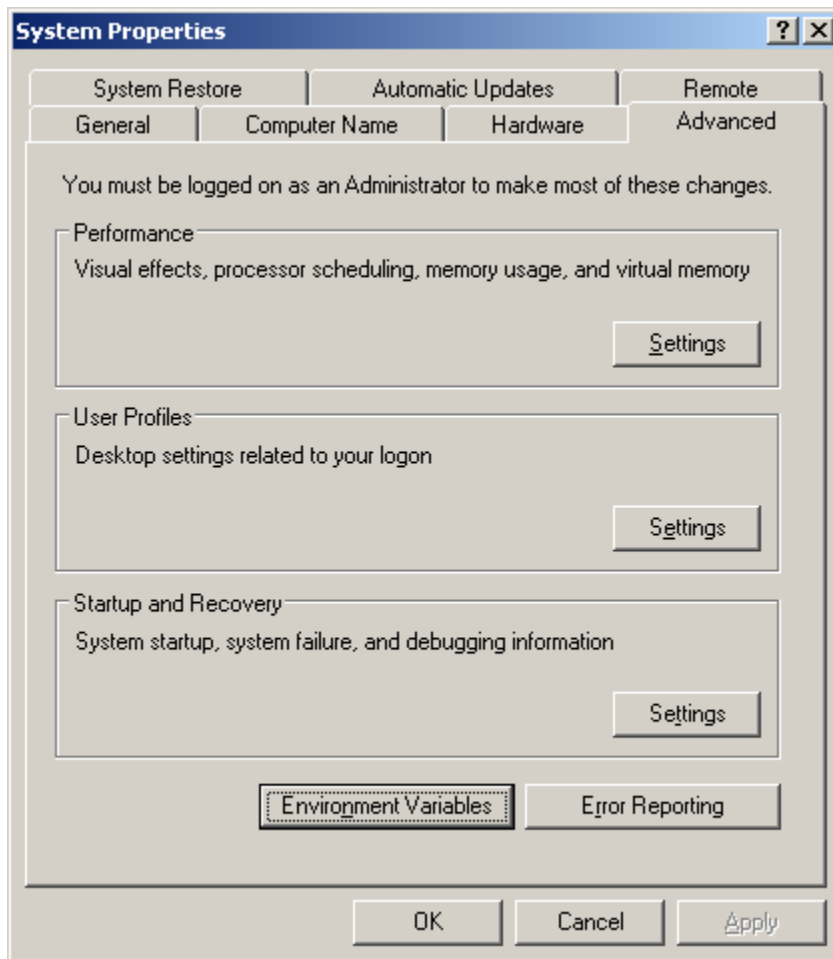
Instalation .....	2
Install xlrd, xlwt, xlutils.....	4
Getting ngnms testing tool. ....	5
Usage examples .....	6
Show help.....	6
Show active vsat's.....	7
Show one particular vsat .....	9
Show hub configuration.....	10
Checking vsat .....	11
Checking hub.....	13
Run one particular test .....	14
Run all active test cases .....	14
Configuring TESTCASES .....	0
Data from output.xls file after running test.....	0
Configuring HUB.....	1
Configuring VSAT. ....	1

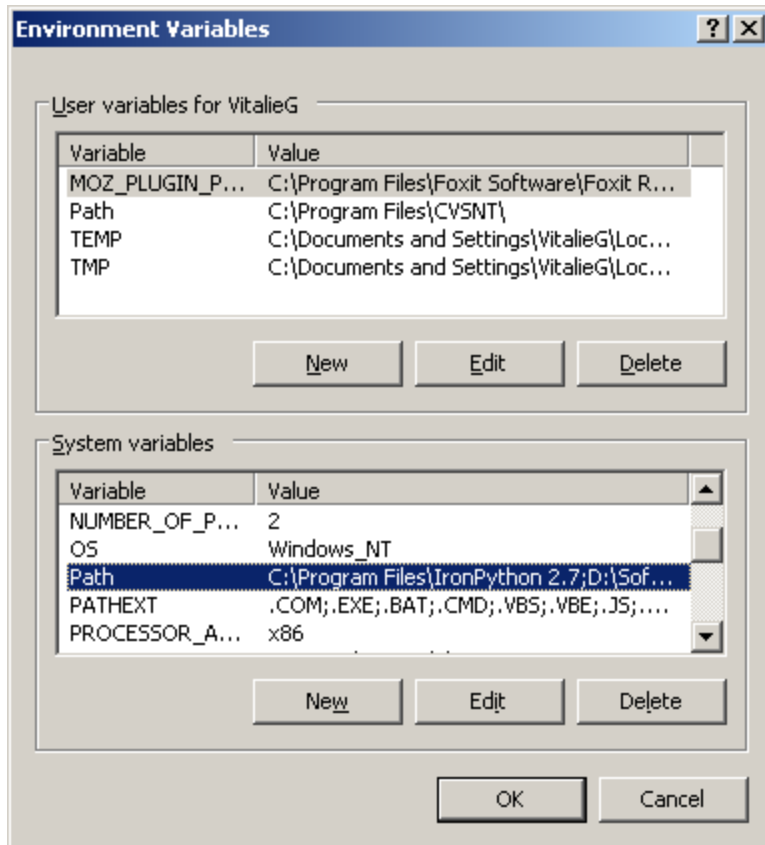
## Installation

- install python 2.7.3:
  - 32 bit: <http://python.org/ftp/python/2.7.3/python-2.7.3.msi>
  - 64 bit: <http://python.org/ftp/python/2.7.3/python-2.7.3.amd64.msi>
- install pycurl 32bit or 64 bit <http://www.lfd.uci.edu/~gohlke/pythonlibs/#pycurl>
- install setuptools: <https://pypi.python.org/packages/2.7/s/setuptools/setuptools-0.6c11.win32-py2.7.exe#md5=57e1e64f6b7c7f1d2eddfc9746bbaf20>

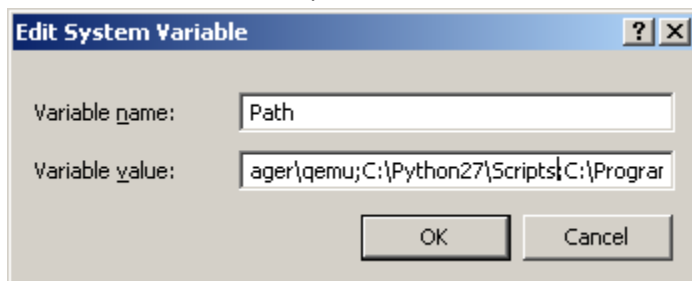
Add python path to PATH variable:

C:\Python27; C:\Python27\Scripts;





Select PATH variable and press Edit:



Copy and Paste C:\Python27; C:\Python27\Scripts; and press OK.

Note: don't miss (;) at the end!

## Install xlrd, xlwt, xlutils

```
C:\Documents and Settings\VitalieG\git\ngnms>easy_install xlrd
Searching for xlrd
Best match: xlrd 0.9.0
Processing xlrd-0.9.0-py2.7.egg
xlrd 0.9.0 is already the active version in easy-install.pth
Installing runxlrd.py script to C:\Python27\Scripts
```

```
Using c:\python27\lib\site-packages\xlrd-0.9.0-py2.7.egg
Processing dependencies for xlrd
```

```
C:\Documents and Settings\VitalieG\git\ngnms>easy_install xlwt
Searching for xlwt
Best match: xlwt 0.7.4
Processing xlwt-0.7.4-py2.7.egg
xlwt 0.7.4 is already the active version in easy-install.pth
```

```
Using c:\python27\lib\site-packages\xlwt-0.7.4-py2.7.egg
Processing dependencies for xlwt
Finished processing dependencies for xlwt
```

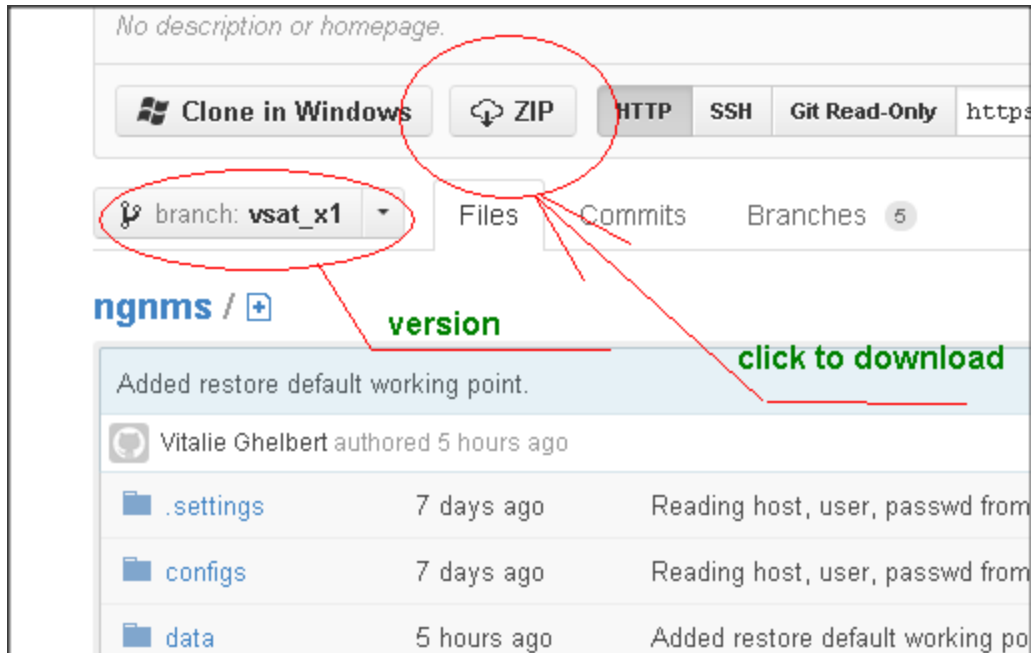
```
C:\Documents and Settings\VitalieG\git\ngnms>easy_install xlutils
Searching for xlutils
Best match: xlutils 1.5.2
Processing xlutils-1.5.2-py2.7.egg
xlutils 1.5.2 is already the active version in easy-install.pth
Installing margins-script.py script to C:\Python27\Scripts
Installing margins.exe script to C:\Python27\Scripts
Installing margins.exe.manifest script to C:\Python27\Scripts
```

```
Using c:\python27\lib\site-packages\xlutils-1.5.2-py2.7.egg
Processing dependencies for xlutils
Finished processing dependencies for xlutils
```

## Getting ngnms testing tool.

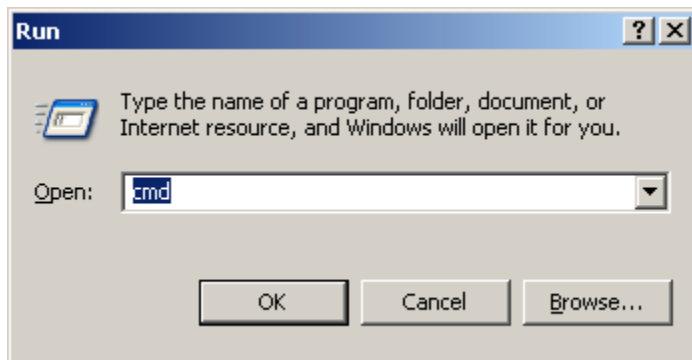
Download and unzip:

<https://github.com/gvitalie/ngnms>



## Usage examples

Open cmd to unzipped folder:



## Show help

```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --help
Usage: wptest.py [options]
```

Copyright 2013 Gilat

### Options:

- version        show program's version number and exit
- h, --help       show this help message and exit
- c DEVICE, --check=DEVICE  
                 check [hub, vsat]'s status.
- n NAME, --name=NAME   vsat name to check.
- s INFO, --show=INFO   show [all, hub, vsat, test]'s info.
- d, --disabled     show disabled rows only.
- i INFILE, --in-file=INFILE  
                 testcases input file [default: data/demo.xls]
- o OUTFILE, --out-file=OUTFILE  
                 save result to file [default: data/output.xls]
- r, --run          run one or [default:enabled] test cases

wptest.py - read and run test cases from excel file.

**Show active vsat's**

- `wptest.py --show vsat`
- `wptest.py --show vsat --disabled`

```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --show vsat
```

INFO: Excel file data/demo.xls!

ENABLED

VSAT : ENABLED

VSAT : V1 : ENABLED

```
Active           = x
Name             = V1
Console IP       = 192.168.140.76
Console PORT     = 1010
Connection timeout = 10
Number of tries  = 3
Tries timeout    = 10
```



```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --show vsat --disabled  
=====
```

```
INFO: Excel file data/demo.xls!  
  
=====
```

```
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH  
DISABLED  
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
VSAT : DISABLED  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
-----  
VSAT : V2 : DISABLED  
-----  
Active                =  
Name                  = V2  
Console IP            = 192.168.140.76  
Console PORT          = 1016  
Connection timeout    = 10  
Number of tries       = 3  
Tries timeout         = 10  
-----  
-----  
VSAT : V3 : DISABLED  
-----  
Active                =  
Name                  = V3  
Console IP            = 192.168.140.76  
Console PORT          = 101  
Connection timeout    = 10  
Number of tries       = 3  
Tries timeout         = 10  
-----
```

### Show one particular vsat

- `wptest.py --show vsat --name V4`

[illegible]

### Show hub configuration

- `wptest.py --show hub`
- `wptest.py --show hub --disabled`

```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --show hub
```

INFO: Excel file data/demo.xls!

ENABLED

HUB : ENABLED

HUB : NS\_3 : ENABLED

```
Active           = x
Name            = NS_3
Type            = NS
URL             = https://172.20.255.1
User            = rnd
Password        = 6DTR2ZHGS6MQQ
```

```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --show hub --disabled
```

INFO: Excel file data/demo.xls!

DISABLED

HUB : DISABLED

HUB : NS 3 : DISABLED

```
Active      =
Name        = NS_3
Type        = NS
URL         = https://ngnms-server/
User        = admin
Password    = manager
```

## Checking vsat

- `wptest.py --check vsat`
- `wptest.py --check vsat --name V2`

```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --check vsat
```

INFO: Excel file data/demo.xls!

[illegible]

```
-- V1 : ENABLED --
```

```
Active      = x
Name        = V1
Console IP  = 192.168.140.76
Console PORT = 1010
Connection timeout = 10
Number of tries = 3
Tries timeout = 10
```

```
step:\> Checking connection ...
status: -> SUCCESS!
```

```
step:\> Checking link status!  
status: Total Backbone Links UP = 1  
status: ->Link UP!
```

```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --check vsat --name V1
```

INFO: Excel file data/demo.xls!

[illegible]

ENABLED

[illegible]

```
-- V1 : ENABLED --
```

Active = x

Name = V1

Console IP = 192.168.140.76

Console PORT = 1010

Connection timeout = 10

Number of tries = 3

Tries timeout = 10

```
step:\> Checking connection ...
```

```
status: -> SUCCESS!
```

```
step:\> Checking link status!
```

status: Total Backbone Links UP = 1

```
status: ->Link UP!
```

INFO: Excel file data/demo.xls!

[illegible]

DISABLED

[illegible]

VSAT : V2

VSAT : V3

VSAT : V4

VSAT : V5

VSAT : V9

VSAT : V10

## Checking hub

- wptest.py --check hub

```
C:\Documents and Settings\VitalieG\git\ngnms>wptest.py --check hub
=====
INFO: Excel file data/demo.xls!
=====
step:\> Connecting to: https://ngnms-server
info:\> [user: admin] [password: manager]
status: 200 https://ngnms-server/navigation/statustree/network
step:\> Scanning ngnms network tree ...
-----
Teleport: main
  Satellite: Satellite
    RF Cluster: rfCluster 1
      NS: controller
-----
info:\> Available network segments names on server:
controller          = 579
```

### Run one particular test

- `wptest.py --run --name 1`

### Run all active test cases

- `wptest.py --run`

Note: to run all disabled tests use `--disabled` option.

- `wptest.py --run --disabled`

## Configuring TESTCASES

The screenshot displays a software interface for configuring test cases, featuring a main table and three pop-up windows. Red arrows indicate the flow of configuration data between the table and the windows.

**Main Table:**

Active	Test N°	OT symbol rate	OT mode code	RTN Channels Frequency Plan	ID symbol rate	ID mode code	Number of Channels	Dynamic/Static	Symbol Rate					
T	T	T	T	T	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4					
1	45000000	QPSK 1/3	DYNAMIC	768	128	QPSK 1/2	QPSK 3/4	111	112	1124	256	128	768	
2	45000000	QPSK 2/3	DYNAMIC	1024	768	QPSK 6/7	QPSK 4/5	111	112	1024	256	128	768	
3	45000000	QPSK 3/5	STATIC	512	768	BPSK 3/4	QPSK 4/5	111	112	1024	256	128	768	
X	4	45000000	16APSK 9/10	STATIC	768	768	QPSK 3/4	QPSK 3/4	111	112	1024	1024	128	768
5	45000000	BPSK 8/9	DYNAMIC	768	768	BPSK 3/4	QPSK 4/5	111	112	1024	256	128	768	
6	45000000	QPSK 3/4	DYNAMIC	768	768	BPSK 4/5	BPSK 2/3	111	112	1024	256	128	768	
7	45000000	BPSK 5/6	STATIC	768	768	BPSK 4/5	BPSK 3/4	111	112	1024	256	128	768	
x	8	45000000	BPSK 9/10	DYNAMIC	768	768	BPSK 3/4	QPSK 1/2	111	112	1024	256	128	768
9	45000000	QPSK 3/4	DYNAMIC	768	768	BPSK 2/3	QPSK 2/3	111	112	1024	256	128	768	
10	45000000	QPSK 2/3	STATIC	768	768	BPSK 6/7	BPSK 3/4	111	112	1024	256	128	768	
11	45000000	QPSK 1/4	DYNAMIC	768	768	QPSK 2/3	BPSK 6/7	111	112	1024	256	128	768	

**FWD Link Window:**

- Uplink Center Frequency: 2810000 (27850000, 30000000 kHz)
- Symbol Rate: 4500000 (3000000, 60000000 sps)
- MODCOD Range - Base MODCOD: QPSK 3/4
- MODCOD Range - Most Efficient MODCOD: 8PSK 4/5
- TX Power: 0 dBm (dBm)
- RFT Uplink LO: 27000000 (10000000, 30000000 kHz)
- 10MHz Output to RFT: Enable

**RTN Link Window:**

- RTN Frequency Slice 1
- RTN Channels Frequency Plan: Static Frequency Plan
- RTN Channel Types

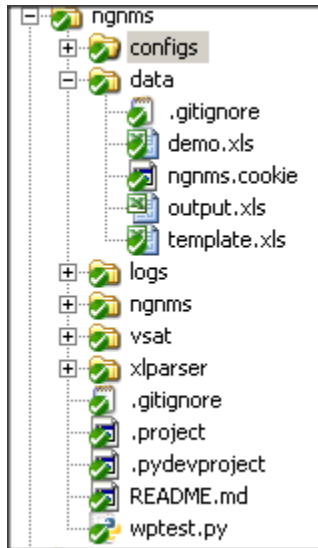
**Dynamic Channels - Channel Mix Window:**

* Index	Symbol Rate	MODCOD	* Number of Channels	Dynamic/Static
1	128	QPSK 3/4	1	Static
2	2048	QPSK 3/4	1	Static



By default, you could use demo.xls file from data directory.

Hint: Make a copy before changing demo.xls file.



You can specify witch input file to use with -i option and where to store data with -o option:

```
wptest.py -i data/demo.xls -o data/output.xls --run --name 4
```

that will read test cases from [data/demo.xls] file, and store to [data/output.xls] file.

NOTE: by default, if no -i and -o options provided, [data/demo.xls] and [data/output.xls] are used.

### Data from output.xls file after running test

	AG	AH	AI	AJ	AK	AL	AM
tion	Max OB bit rate	Max IB bit rate	VSAT CPU	Number of transmitted OB packets	Number of received IB packets	Number of OB retransmit packets	Number of IB retransmit packets
	S	S	S	S	S	S	S
	0	0	[8]/[9]	0	0	0	0

## Configuring HUB.

A	B	C	D	E	F
Active	Name	Type	URL	User	Password
x	NS_3	NS	<a href="https://172.20.255.1">https://172.20.255.1</a>	rnd	6DTR2ZHGS6MQQ
	NS_3	NS	<a href="https://ngnms-server/">https://ngnms-server/</a>	admin	manager
	HSP1	HSP			
	HSP2	HSP			

Active: just one line should be active.

Name: fill here network segment name

Type: optional

URL: ngnms link

User: ngnms user

Password: ngnms password

## Configuring VSAT.

A	B	C	D	E	F	G
Active	Name	Console IP	Console PORT	Connection timeout	Number of tries	Tries timeout
x	V1	192.168.140.76	1010	10	3	10
	V2	192.168.140.76	1016	10	3	10
	V3	192.168.140.76	101	10	3	10
	V4	10.111.35.6	1004	10	3	10
	V5	10.111.35.7	1005	10	3	10
	V9	10.111.35.8	1009	10	3	10
	V10	10.111.35.9	1010	10	3	10

Active: just one line should be active.

Name: fill correct Vsat name.

Console IP: telnet ip connection

Console Port: telnet port connection

Connection timeout: time until timeout

Number of tries: how many tries to check until link UP.

Tries timeout: time between each try if vsat has link DOWN.