

SNMP Programming Project  
David Huang  
CS 158B

# Functional Specifications

The goal of this project is to code an application that can monitor the traffic in a given device. The application should be able to discover the device's interfaces, IP neighbors, and give details on the traffic occurring on each interface. The output of this program must be user-friendly.

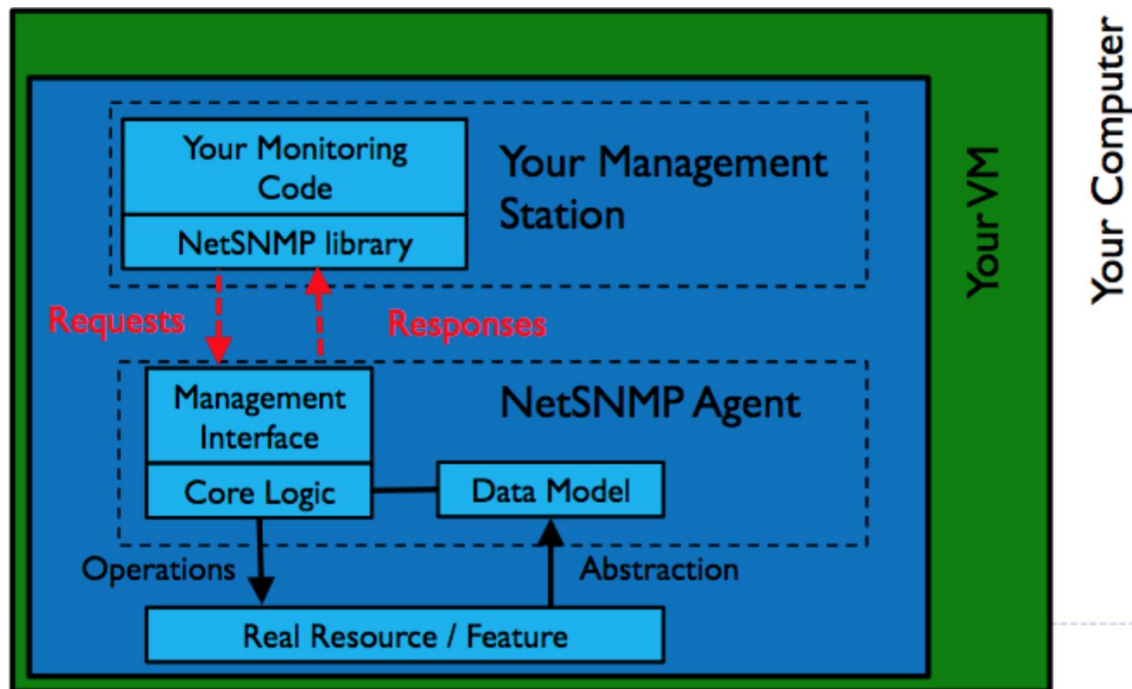
In order to run the application, the user's system should have the following requirements:

- [Ubuntu 14.04 LTS Trusty Tahr](#)
- SNMP Daemon
- [Net-SNMP](#)
- [Python 2.7](#)
- [Easy SNMP API](#)
- [MIB Database](#)
- Internet Connection

The application should only be using the API from Net-SNMP which is processed and turned Pythonic through EasySNMP's API. Their API is a direct fork from Net-SNMPs library. The data pulled from the SNMP queries should all be from the MIB-II database.

The user should easily be able to access the application and after entering in the IP they wish to monitor, SNMP Community, and polling details, receive the data requested in an timely and organized manner.

## Design Specifications



The overall design of the application will follow this model to reach functionality. Inside the user's VM, the application will make use of the EasySNMP API to create a session with the NetSNMP Agent to request the data needed to present to the user. In order to make sure the user can retrieve the data requested, the NetSNMP Agent needs to be configured such that there is a community that exists with a broader view than the default public view.

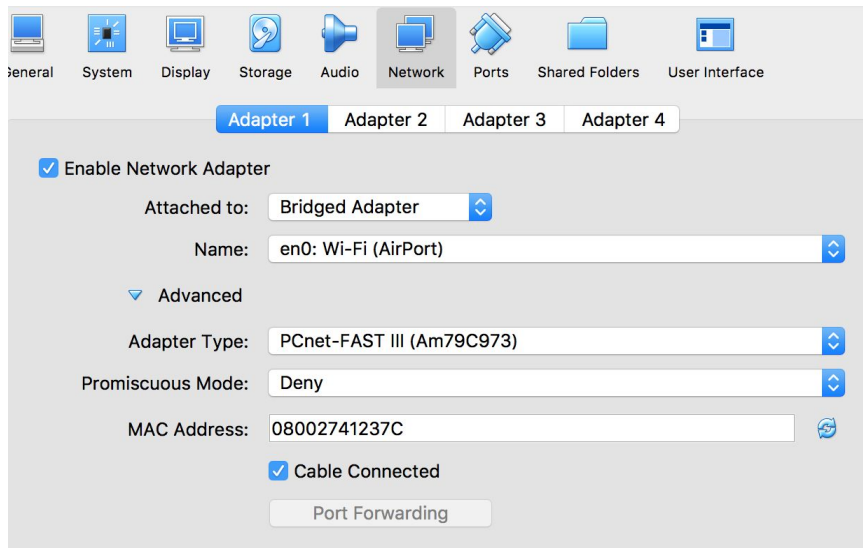
To discover the interfaces, we will be querying from the MIB-II table `ifTable` ([.1.3.6.1.2.1.2.2.1.2](#)).

In order to also find the neighboring devices, the VM needs to be connected using a bridged adapter with the host computer so it can join the host network and find the other devices. In order to find the neighboring devices, the application will be pulling from the ARP Table in MIB-II's `ipNetToMediaPhysAddress` ([.1.3.6.1.2.1.4.22.1.2](#)).

To calculate the bandwidth per interface, we will be querying the `ifInOctets` from the MIB-II's `ifTable` ([.1.3.6.1.2.2.1.2.1.10](#)). Taking one reading and calculating the difference after reading the value again after the user's time interval, we then convert that value into Mbps.

# Testing

To start, boot up a VM with Ubuntu 14.04, doesn't matter how much RAM or storage you give it. What does matter is that you bridge your internet connection to the VM so it can find the other devices in your network. I used these settings on VirtualBox.



After getting your VM to boot, go ahead and install NetSNMP following the instructions in this link. <https://www.maketecheasier.com/net-snmp-part-1/>

Then, install EasySNMP using these instructions.  
<http://easysnmp.readthedocs.io/en/latest/#installation>

After installing both and testing to make sure that SNMP is running correctly, find your snmpd.conf file (usually in /etc/snmp/) and make a new view that will walk at .1 instead of the default .1.3.6.1.2.1.25.1 . Edit the public community so that it reads from this view.

Finally, before running the application, to update your ARP tables do the following:

- 1) In terminal, run the command `ifconfig`
- 2) Find your IP Address and then ping the Broadcast Address (`ping x.x.x.x -b`)
  - a) Ex: if my IP address was 15.2.32.15, I would run `ping 15.2.32.255 -b`
- 3) Run command `arp -a` for reference

To run the application, just navigate to the file's location in terminal and run the command: `python snmp.py`

You will be prompted with questions about your input, remember when entering in the community that it should be shared with the SNMP agent or else there won't be any information. Another tip is to have something running in the background internet-wise such as a video or a download to get good results from the application.

Sample Output:

```
david@david-VirtualBox: ~/Desktop
david@david-VirtualBox:~/Desktop$ python snmp.py
Please enter in the following:
IP Address of Agent (or 'localhost' for local use):
localhost
Please enter in the community we will be using:
public
Please enter in the number of bandwidth samples:
5
Please enter in the time interval in seconds between each sample:
2
Hello David Huang <huang.z.david@gmail.com> from: 1 Washington Square San Jose,
CA 95112

Interfaces
-----
1 lo
2 eth0

Neighbors
-----
1 10.0.0.1
2 10.0.0.134
3 10.0.0.137

Network Bandwidth
-----
Interval 1 for 2 seconds:
Interface: lo Down: 0.00143890380859Mbps
Interface: eth0 Down: 0.000132751464844Mbps
Interval 2 for 2 seconds:
Interface: lo Down: 0.0Mbps
Interface: eth0 Down: 0.0Mbps
Interval 3 for 2 seconds:
Interface: lo Down: 0.00339660644531Mbps
```