**Dependency**

Please download and install python 3.\* from the following site,

[*https://www.python.org/downloads/*](https://www.python.org/downloads/)

You also need following modules,

*Hyper-h2*

*cv2*

*pickle* (if not available with python package)

I think other modules such as *threading, time, matplotlib, socket* are available with the python package.

Module installation:

Ensure you have installed *pip* module first. For Linux, I think it will be installed with the python package. Otherwise please follow the link to install pip,

[*https://www.tecmint.com/install-pip-in-linux/*](https://www.tecmint.com/install-pip-in-linux/)

In windows, please follow the link to install *pip*,

<https://github.com/BurntSushi/nfldb/wiki/Python-&-pip-Windows-installation>

*Hyper-h2* installation (run it on a command prompt or a terminal)

pip install h2

*OpenCV* installation

pip install opencv-python

or

pip install opencv-contrib-python

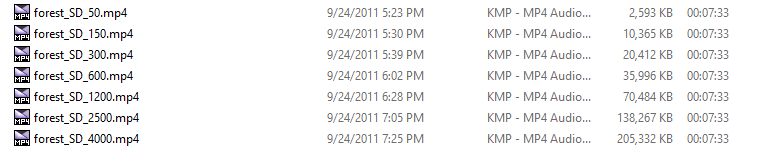
*pickle* installation

pip install pickle-mixin

**Video archive**

Video archive should contain the video with bitrates 50, 150, 300, 600, 1200, 2500, 4000.

We have to follow some specific naming conventions for the files. The name has 3 parts and the extension. For example,



Here, we can see the 3 parts of the for each video are separated with ‘\_’ and the extension is ‘mp4’. First part is the name of the video, second part is the resolution information and the third part is the bitrate.

You can download the video from the following dataset library, rename them according to the described format and put them in a folder.

<http://www-itec.uni-klu.ac.at/dash/?page_id=207>

(For simplicity, I have uploaded a video set in google drive. Those can be downloaded from the link: <https://goo.gl/vpnEAd> )

**Input guide/ Running procedure:**

For Server

Format:

**python h2server.py "PUSH" "PATH INCLUDING NAME OF THE FILE EXCEPT LAST PART" "VIDEO EXTENTION"**

Here, PUSH value is between 1 to 3

"PATH INCLUDING NAME OF THE FILE EXCEPT LAST PART" is a string which is path to the video archive. It also contains the filename except the last part and extension. Video file name has a specific format and consist of 3 parts and extension. It will be discussed separately.

Example:

**python h2server.py 3 "F:\Work\Educational info\Gottingen\Internet Technologies\video\_archive\forest\_SD" "mp4"**

For Client

Format:

**python h2client3.py "BUFFER" "MIN\_BUFFER" "QUALITY" "AUTO\_QUALITY" "SHOW\_OUTPUT" "SHOW\_VIDEO" "RTT"**

Here, BUFFER value is between 1 to 6

MIN\_BUFFER value can be between 0 to 5

QUALITY value ranges from 1 to 7

AUTO\_QUALITY value is either 1 or 0. For our case we should always put it 1

SHOW\_OUTPUT value is either 1 or 0. If you want to see the running statistics then put it 1.

SHOW\_VIDEO value is either 1 or 0. If 1 then you will be able to see the playback which is not important for the statistics. I prefer it to be 0 cause for some reason it slowing down the whole process.

RTT values are 0 100 200 300 400. If you want default RTT then put 0 which is not 0 in real-time.

\*Client can also be run without the parameters. In this case it will start with its default values.

Format:

**python h2client3.py**

Here the default values are BUFFER=6, MIN\_BUFFER=0, QUALITY=3, AUTO\_QUALITY=1, SHOW\_OUTPUT=1, SHOW\_VIDEO=0 and RTT=0

Example:

**python h2client3.py 6 0 3 1 1 0 0**

or

**python h2client3.py**

If you want to assign parameters, then assign all otherwise none.

Client will run for 214 seconds and at the end, it will write some files in the source directory. This files can be used to generate plots.