Demo Usage

Overview

This Demo shows how we deal with topology change in a datacenter. The demo is built by ONetSwitch and RYU controller. All boards are run as OpenFlow 1.3 switches.

In this demo, 3 computers are needed. The first one is the Mac Mini, which is run as a OpenFlow controller. The second computer is run as a video server to provide large video stream, and the third one is run as a video client to receive the streams.

The demo provide a whole view of the physical link of switches as well as client and server connected to switches. After it finished collect topology, it will start route streams to destination. It will also keep on watching the status of all links, if one link on the path of the video stream is down, it will recalculate path immediately and download flow-entries to ensure video stream not down.

Before start

Make sure the rack and the controller is upgrade successfully, and the other two computers have installed VLC on it.

Start the demo

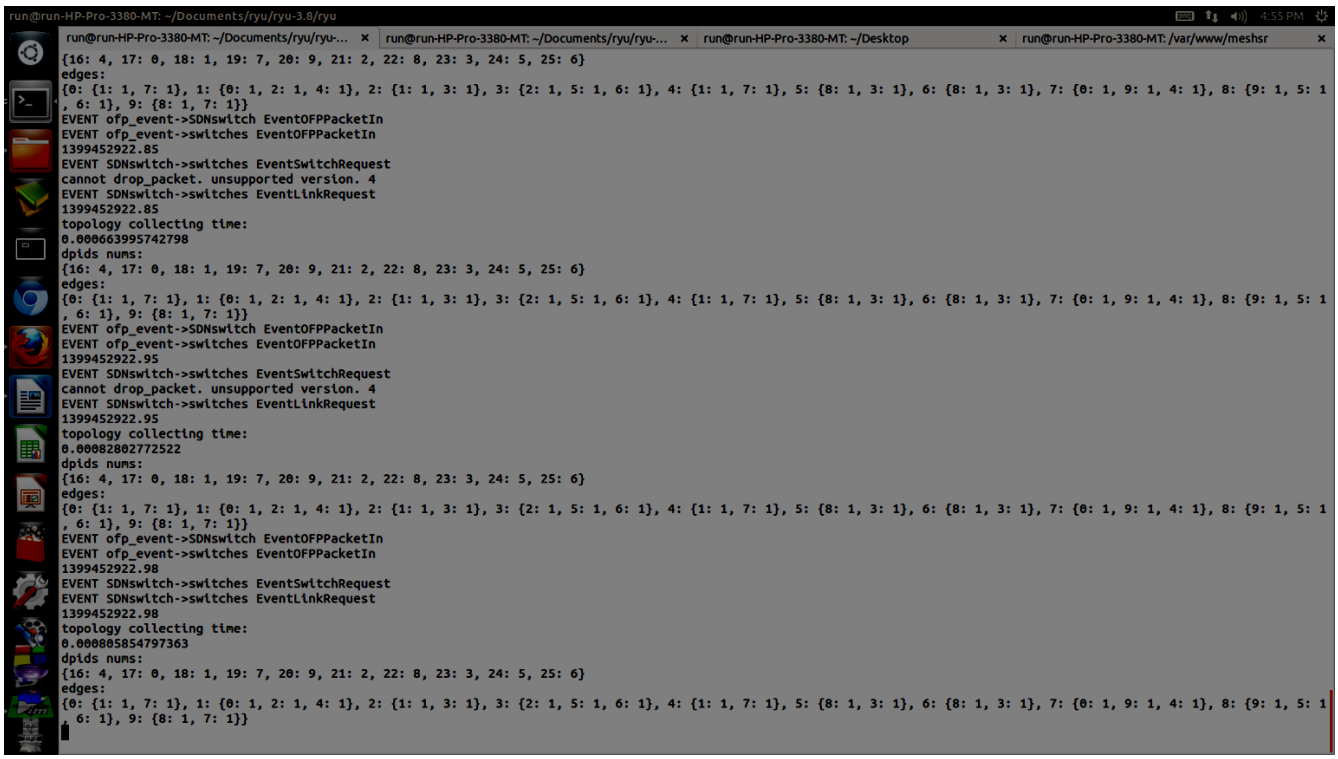
Each ONetSwitch provide 4 port at the front of its panel. The rest two computer working as video server and client should connect to any port ONetSwitches provided. They also have to setup IP first. Take 192.168.1.100 and 192.168.1.101 as an example. They have to ping each other.
At the computer 192.168.1.100, open a console, and run the following command:
ping –t 192.168.1.101
At the computer 192.168.1.101, open a console, and run the following command:
ping –t 192.168.1.100

At start, the ping has no response, and return "No route to host" and other error message.
We should also start VLC, one computer stream a video, useing "UDP VIDEO_CLIENT_IP:1234" "H264+MP3(TS)", and the other open a UDP stream by open "udp://@:1234"

Then we can start controller and make all net active:
All the start commands are write in topo_start.sh, which is in ryu folder. One can easily start demo by run the script as follows:

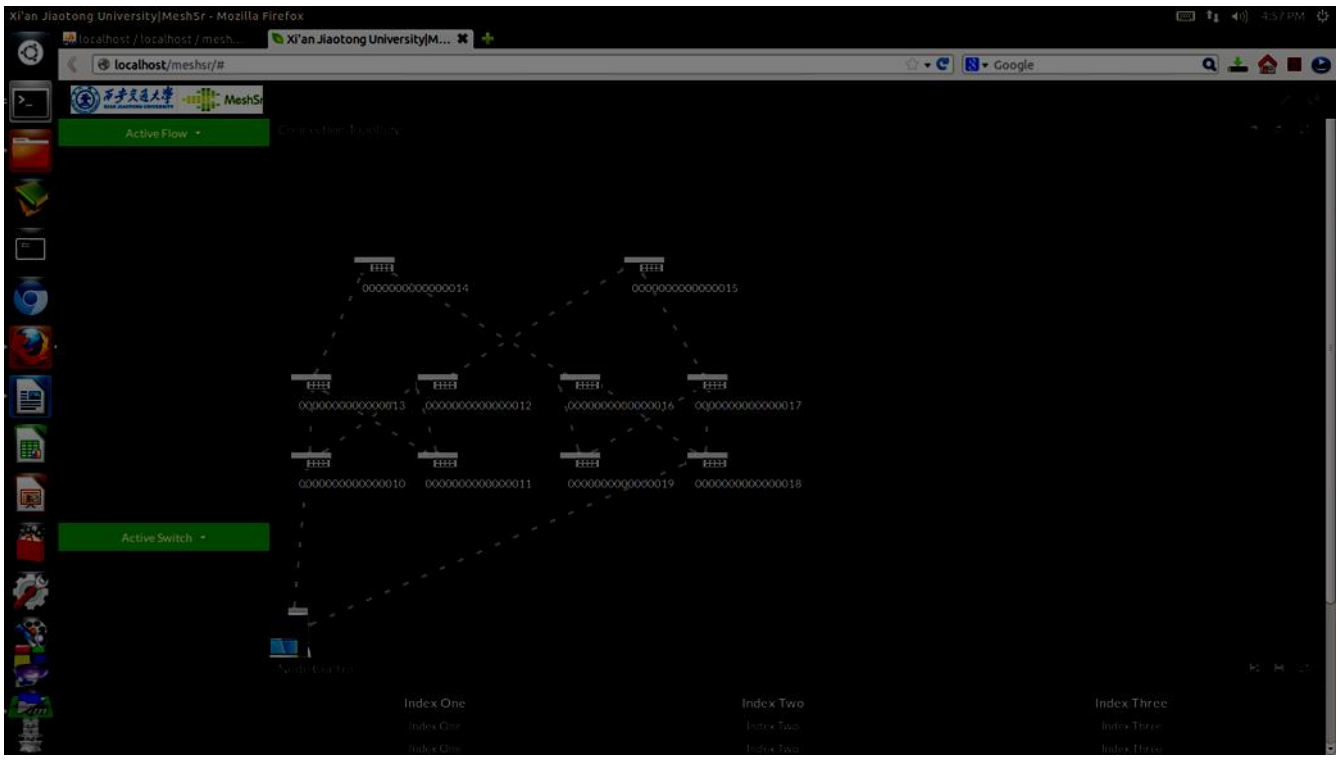
run@run-HP-Pro-3380-MT:~/Documents/ryu/ryu-3.8/ryu$ bash topo_start.sh

On the terminal screen, you may see the follow print continues:



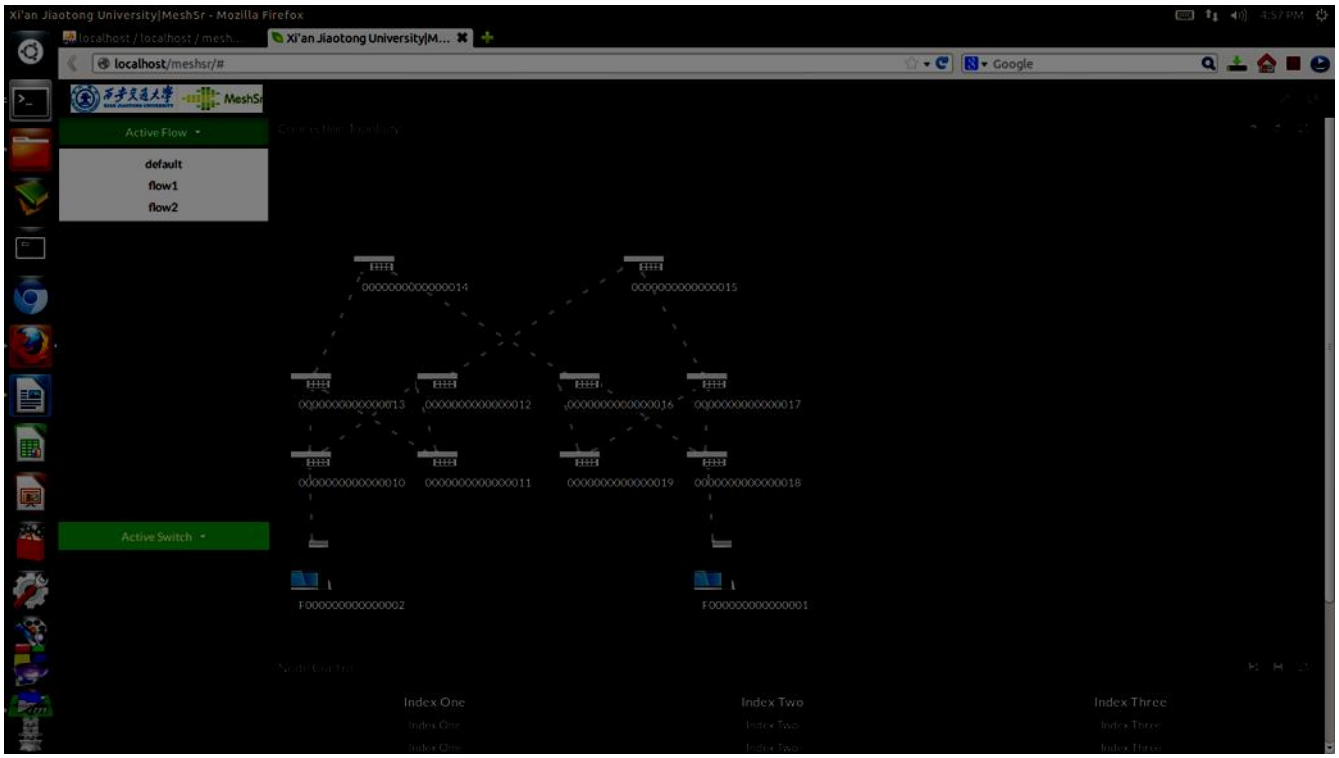
After 1-2 minutes, you can open a web browser, and access address "localhost:8080/meshsr"

Press F5 to refresh the screen, if the controller finish topology collect, you will get follows:

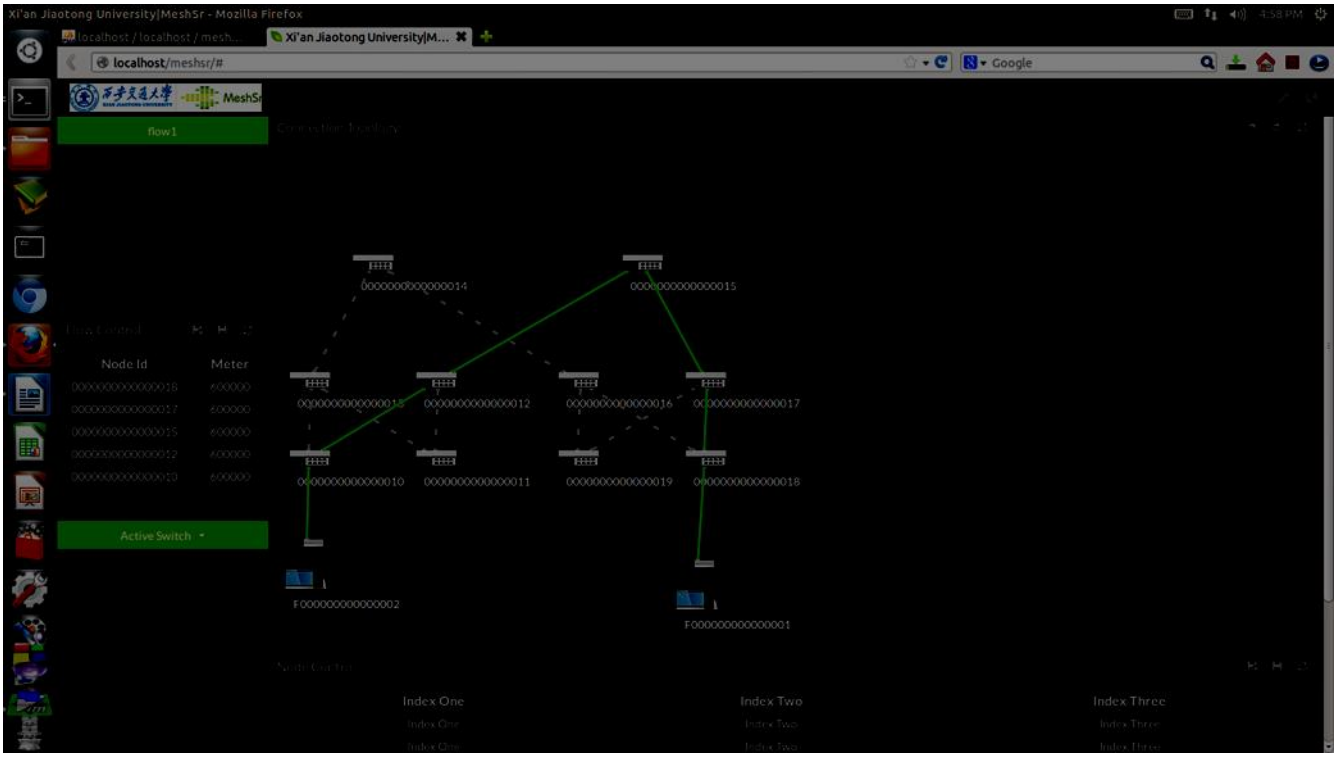


You can find all the topology is show on the default screen. One can also move switches and servers by drag it. All the lines show here means a real physical path between them.

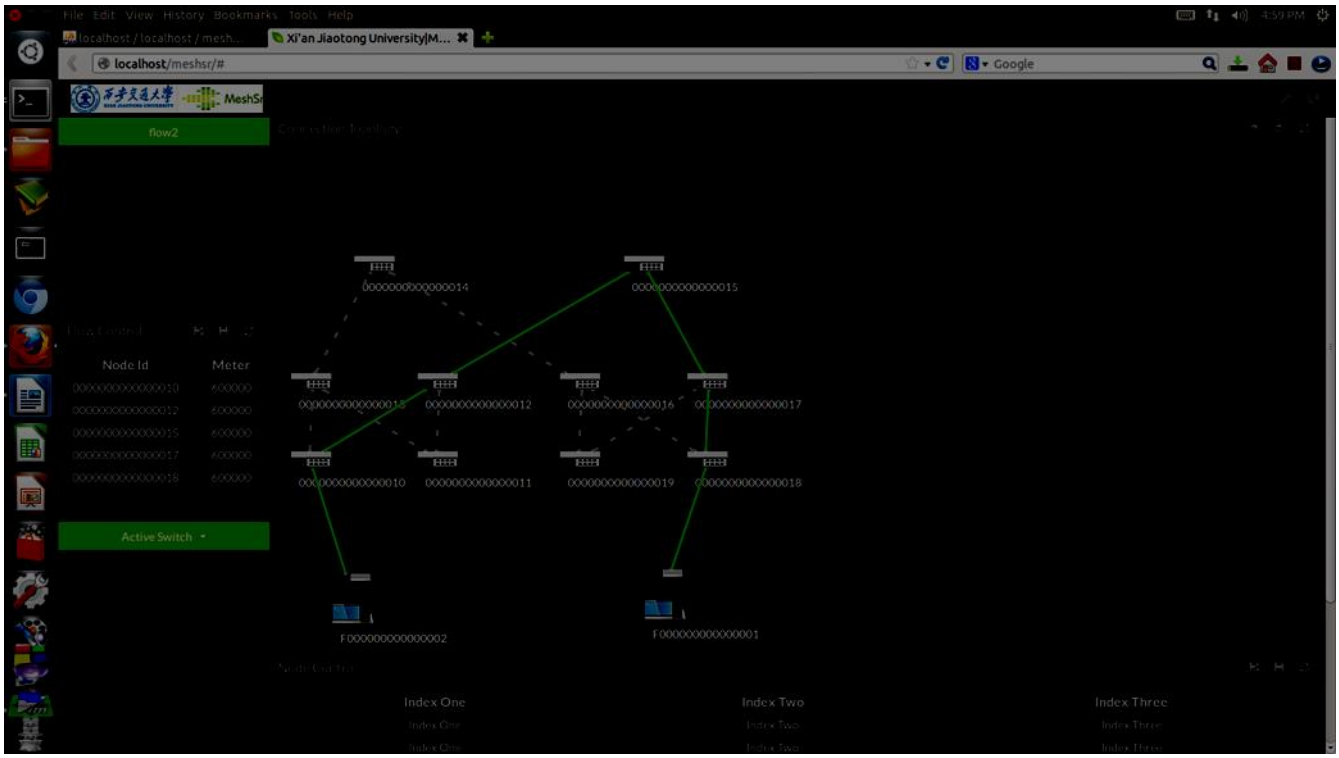
The green list title "Active Flow" can be clicked and a list shows all the real flow name.
In this demo, two flow are shown. One is from the server1 to server2 and the other is from server2 to server1.



Click a flow name will show the real path on which it is forwarded. Click flow1:

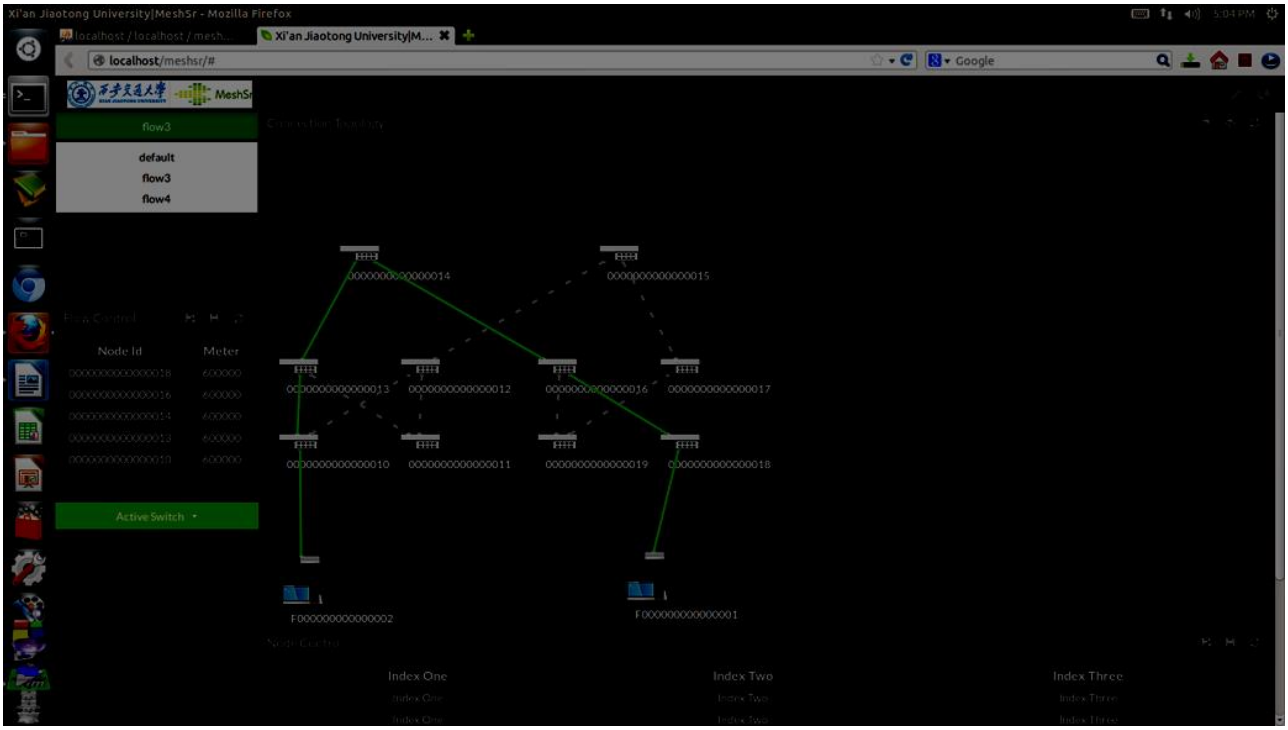


Then you can click flow2



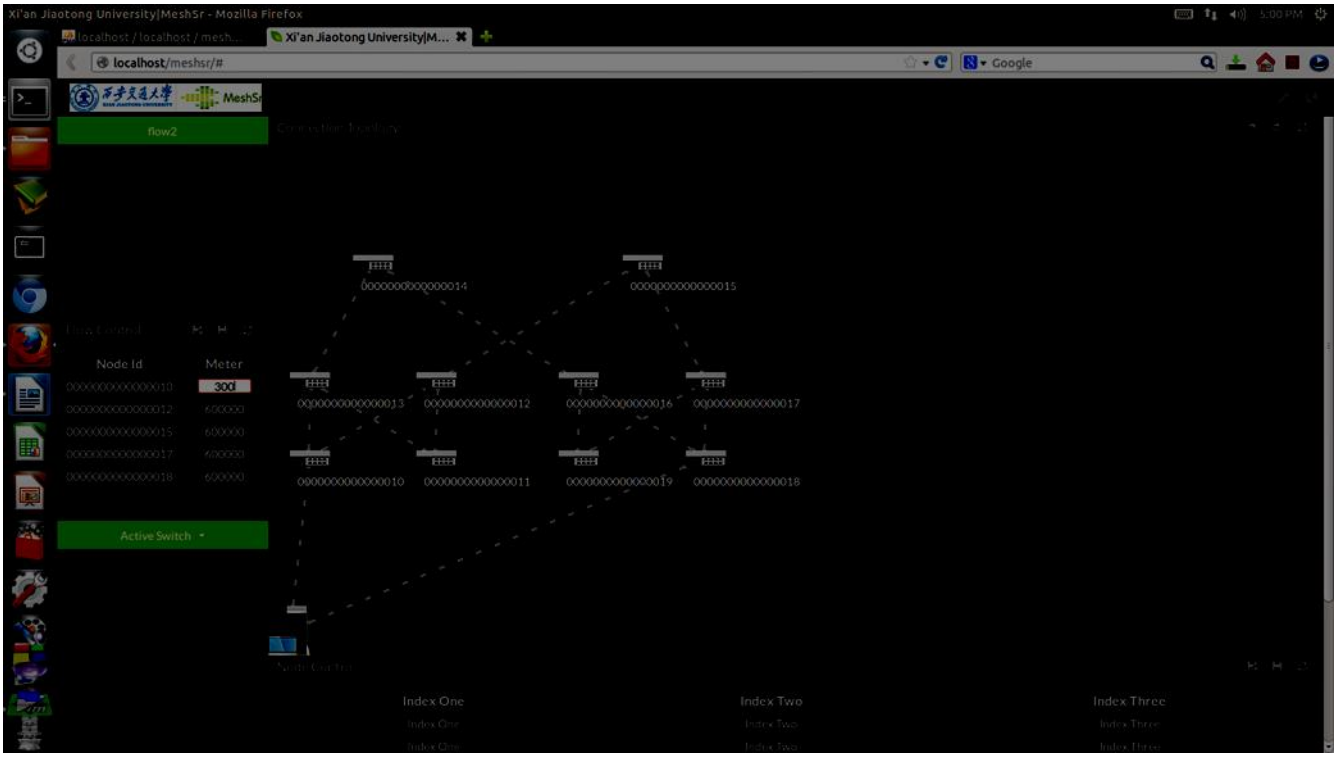
At this time, you may find difference in the console of the two computer. The ping start to return message, and this shows the network is really works.

You can change the topology. From the last picture, we can get that the flow is move between switch 17 and 18, so we unplugged the cable between the two switches. After a few seconds, press F5 to refresh the screen, we will get:



Now the physical link between 17 and 18 is disappear. The demo find a new way to forward the flow. It is really fast that you may find the video stream is only block or lost pixels for one second, and the ping package lost 1 or not lost any of them.

OpenFlow use meter to limit flow speed. Set meter value can let engineer to better realize QoS: select a flow by click it name on the left flow list. Then the meter value will show on the left bottom, as following. The default value is set to 600,000K. When double click the value, you can edit it. After set a new value, click somewhere else. Then click the save button  . If you set it small enough, the video will soon lost pixels or blocked.



If you set the value back, and save it. The video will play smooth again.