Naval Research Laboratory

Washington, DC 20375-5320

NRL/MR/5590--12-9427

Testing Ethernet-Over-DWDM Circuits Using Open Source Tools

CHRISTOPHER L. ROBSON

Center for Computational Science
Information Technology Division

August 22, 2012

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
22-08-2012	Memorandum Report	01 March 2012 – 30 April 2012

4. TITLE AND SUBTITLE

Testing Ethernet-Over-DWDM Circuits Using Open Source Tools

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Christopher L. Robson

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
4555 Overlook Avenue, SW
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/5590--12-9427

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
4555 Overlook Avenue, SW
Washington, DC 20375-5320

10. SPONSOR / MONITOR'S ACRONYM(S)

NRL

11. SPONSOR / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The purpose of this test was to collect and report on performance characteristics of a One-gigabit Ethernet circuit provisioned over DWDM.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Christopher Robson
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	Unclassified Unlimited	7	**19b. TELEPHONE NUMBER** *(include area code)*
Unclassified	Unclassified	Unclassified			(202) 404-3138

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

Testing
Ethernet-Over-DWDM Circuits
Using Open Source Tools

1. Purpose.

   The purpose of this test was to collect and report on performance characteristics of a One-gigabit Ethernet circuit provisioned over DWDM.

2. Configurations.

   a. Test Network Configuration.

      Figure 2.a.1 illustrates the network to be tested between the two Sites.

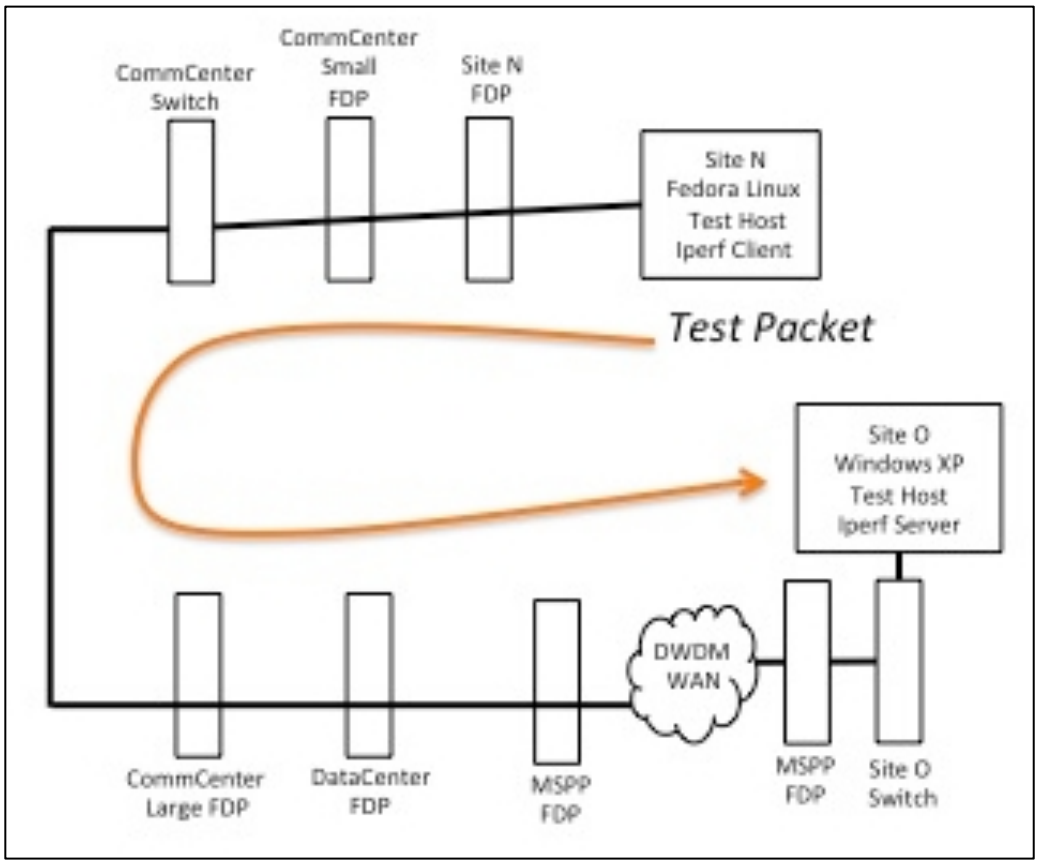

Figure 2.a.1.

   b. Test Host Configuration.

Manuscript approved June 7, 2012.

Each site was configured with a computer system for executing a performance test tool. At Site N, a Dell R310 Intel Xeon X3430 2.40Ghz 1U Fedora (v14) Linux (v2.6.35-14-95 SMP) system was configured to function as a testing client. Site O hosted a Dell D610 Laptop running Windows XP functioned as the testing server.

3. Test Tool.

   The performance tool used for this test was "iperf". It was used to measure the maximum TCP bandwidth performance of the Site N - Site O circuit. Iperf also reports delay jitter and datagram loss, but these characteristics where not the focus of this test. Another words, no concern was giving to options such as "disabling Nagle's Algorithm". However window size was manipulated for reasons explained in section 6.

4. Test Procedures.

   The test process ran a iperf client-server configuration. Site O's computer functioned as the TCP port 5001 receiver to the Site N computer TCP test traffic transmitter. Issued from a command line terminal window, the following commands where executed on each system to accomplish the desired iperf runtime configuration:

   i. Site N: iperf –c [Site O's IP address]
   ii. Site O: iperf -w1024 –s

5. Initial Test Results.

   From the onset, the test demonstrated issues between the Site O server and Site N client. Performance tests were reporting highly degraded bandwidth. Most notable was the bandwidth appeared to be "throttled", that is, fairly consistent bit rates at one-third the capacity of the circuit was observed. The following figure illustrates the low data rates reported during initial testing.

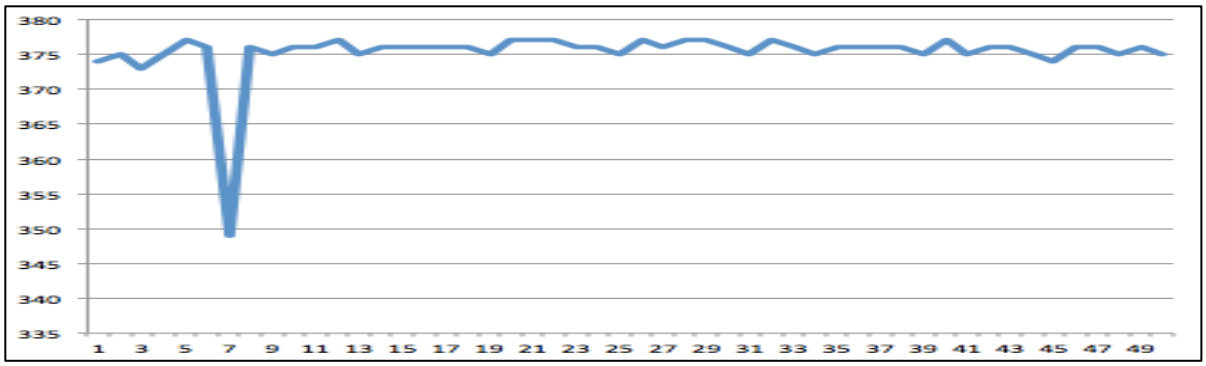

Figure 5.1.

6. Determining the Cause of the Performance Issues.

The following steps where taken in determining the cause of the performance issues experienced:

a. To help find a solution to the problem, the DWDM provider was contacted. At which time the provider determined that the MSPP[1] devices in the circuit where not configured according to best practices. First, it was determined the Site N MSPP was not configured in the same way as the Site O MSPP, that is, one had Autonegotiation[2] set while the other did not. However, despite changing this configuration performance did not improve. (Note: Autonegotiation was left enabled on both MSPPs).

b. The next observed anomaly with the MSPPs was the link-level configuration disparity. The Site N link, that circuit between the Site N MSPP and the Ethernet switch was configured in HDLC[3] frame format and the Site O MSPP link to Dell D610 was setup as GFP[4]. The DWDM provider suggested that this should be corrected to follow the most common configuration. Then tests where conducted to rule out the link-layer mismatched. Neither of these changes effected any improvement in performance. (Note: GFP was set as the final setting).

c. The next step taken was to completely reconfigure each MSPP. Again, this action did nothing to resolve the problem.

d. During the above debugging steps (a), (b) and (c), traffic statistics showed some questionable data which is listed below.

   i. SITE N input bytes: 10,648,267,560,912
   ii. SITE N output bytes: 18,485,580,310
   iii. SITE O input bytes: 18,489,281,768
   iv. SITE O output bytes: 10,648,571,690,608

   To further isolate the problem and because there was no ability to simultaneously observe in real time, statistics on the Site N and Site O host, the configuration in Figure 6.d.1 was implemented.

[1] Cisco ONS *15454* SONET Multiservice Provisioning Platform (*MSPP*) provides SDH solutions for interfaces such as DS3 and data interfaces such as 10/100/1000 Mbps Ethernet with STM1 through STM64 optical transport bit rates in both gray and DWDM wavelengths.
[2] Autonegotiation is a process for choosing link level transmission parameters such as link speed.
[3] High-Level Data Link Control (HDLC) is a bit-oriented synchronous data link layer protocol developed by the International Organization for Standardization (ISO).
[4] Generic Framing Procedure (GFP) or ITU-T G.7041 is used to mapping of variable length, higher-layer client signals over a circuit switched transport network.

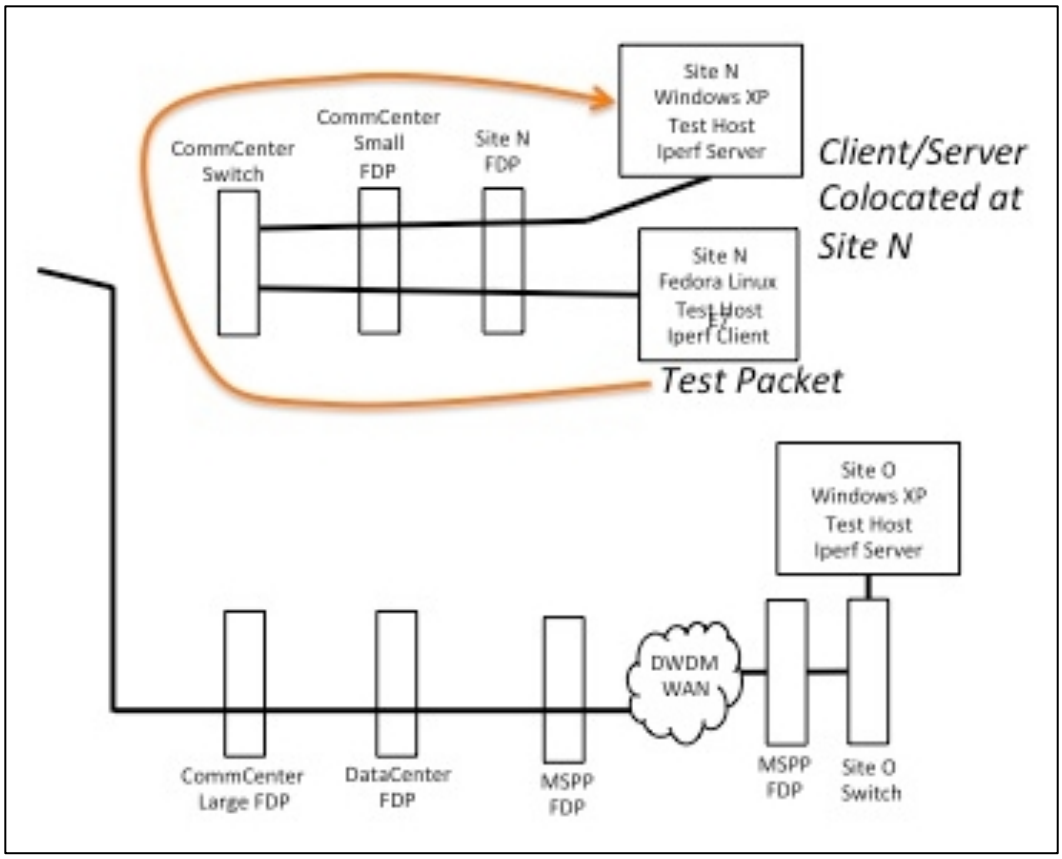

Figure 6.d.1.

It was soon determined that there was a compatibility issue between Windows XP iperf and Linux iperf. By default, iperf running on Windows XP implements a TCP window size of 8 KBytes while Linux default is typically 85.3 Kbytes. The result of this imbalance is illustrated in Figure 5.1. which seems to indicate a circuit that has been constrained.

Further, the Fedora Linux was programmed with a 64 Kbytes window size, which further exasperated the problem. Figure 6.d.2. shows these TCP tuning parameters.

```
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_sack = 0
net.core.netdev_max_backlog = 250000
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.core.rmem_default = 16777216
net.core.wmem_default = 16777216
net.core.optmem_max = 16777216
net.ipv4.tcp_mem = 16777216 16777216 16777216
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 65536 16777216
```

Figure 6.d.2.

7. Final Results.

   The following four charts provide selected extractions of the performance data collected once the client and server TCP parameters where synchronized (as shown in section 4). As the charts demonstrate, iperf bandwidth had clearly improved to over ***80%*** utilization of the circuit. After ***8,755*** test runs, the average bandwidth speed realized was ***<u>918.09 Kbytes/sec</u>***.

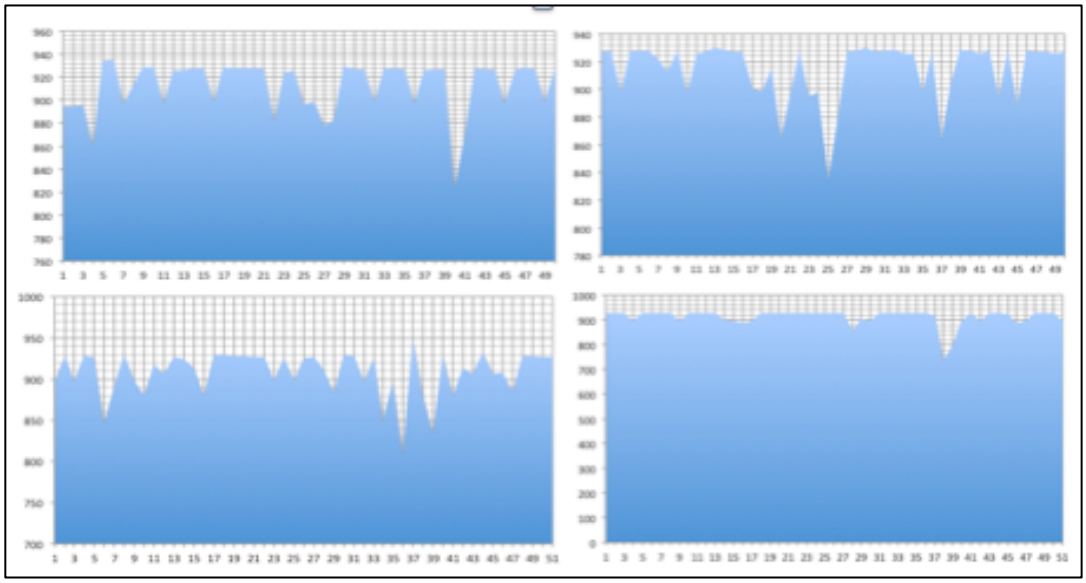

Figure 7.1.

8. Conclusions.

   By adjusting the server's TCP window size so that it was greater than the transmitting client, bandwidth use of the circuit was able to reach an overall average of about ***90%*** utilization.

9. Final Observation.

   While the issue of compatibility between window sizes was overcome, this issue could have been avoided by using the same system for the receiver as the transmitter. Further, because iperf's behavior was somewhat different on each of the systems, it might have been more prudent to exploit another bandwidth test tool. In fact, if both systems were Fedora Linux OS based systems[5] using another performance tool called "netperf", better performance results may have been realized. Other reasons for using Fedora Linux is flexibility and the availability of test suites, such as MSDPI[6].

---

[5] Because of logistics issues, deploying such a configuration was overruled.

[6] Multi-Service Domain Protecting Interface is a control plane based on Session Initiation Protocol under development at Site N. Its feature sets include, plain text domain database exchange, local/remote system management and full integration of the netperf test tool.