

White Paper

Throughput Improvements with the Intel®
Data Plane Development Kit

Lanner



Table of Contents

1	Abstract
1	About the Intel® DPDK
2	Experiment Design
4	Multiple-Process Example Setup
4	Test Result and Conclusion
5	Benefits in Network Security
5	Benefits in Cloud Computing
7	Appendix: Throughput Data

Abstract

The next-generation Ethernet promises network bandwidth from 10 to 40 Gbps and beyond. This trend of ever-increasing bandwidth is driven by the explosion of Internet users and growing use of bandwidth-hungry applications. However, elevating the processing throughput to match the bandwidth of the network has long been a challenge for the network appliance. Is it possible for an appliance to deliver traffic fast enough so that the overall system can attain the maximized efficiency? A solution is provided by the Intel® Data Plane Development Kit (DPDK).

The Research and Development of Lanner Electronics has conducted research on the effectiveness of data plane processing with the Intel DPDK. In the test environment, the standard OS networking stacks were replaced by DPDK programs. The test results show the Intel DPDK improves the throughput by 3 to 4 times on the Intel® multi-core processor architecture.

The Intel DPDK is specialized packet processing software optimized for multi-core architectures. It tackles the problem on how to architect software where packets from multiple streams of network traffic can achieve optimal performance. Lanner Electronics, known for our leadership in network appliance hardware, has been successfully developing and manufacturing Intel® architecture platforms that leverage the Intel DPDK to up-scale network appliances in packet processing performance. Furthermore, by basing our network appliances on Intel architecture, we

are able to provide a complete line of standards-based platforms which allows a low cost of development on high-performance servers with integrated hardware and software accelerators. ¹

About the Intel® DPDK

Data Plane packet processing requires moving data from an ingress LAN port to the system memory, classifying the data and then moving the data to a destination egress port as efficiently as possible. To speed up this process and thereby generate higher throughput, Intel® has introduced a framework for fast packet processing in data plane applications, formally called the Intel® Intel® Data Plane Development Kit (Intel DPDK). The Intel DPDK improves L2 through L3 packet processing performance on Intel architecture. It provides a set of libraries as the building blocks for creating customizable yet optimized data plane applications.

The Intel DPDK core components are a set of libraries that provide all the elements needed for high-performance packet processing applications. This set of libraries comprises the following categories:

- The Environment Abstraction Layer (EAL): The EAL is responsible for loading and launching the Intel DPDK. It acts as an interface/access to all the low-level resources such as hardware and memory space (e.g. PCI access, logging, memory allocation, etc).
- Memory Manager (librte_malloc): This library provides an API to allocate memory from the named memory zones created from the hugepages instead of the heap.
- Ring Manager (librte_ring): The Ring Manager provides an API for fixed-size lockless FIFO (First –in First-out) operations in a ring to store objects or to allow communications between cores.
- Memory Pool Manager (librte_mempool): The Memory Pool Manager is responsible for allocating pools of objects held in a ring with lockless lists available per logical core to support fast bulk enqueue/de-

queue of the objects (from librte_mbufs described below).

- Network Packet Buffer Management (librte_mbuf): The mbuf library provides the ability to create and destroy buffers that may be used by the Intel DPDK application to store message buffers. The message buffers are stored in the mempool (using the librte_mempool library)
- Time Manager (librte_timer): This library provides a timer service to the Intel DPDK execution units, providing the ability to execute a function asynchronously. It uses the HPET interface provided by the Environment Abstraction Layer (EAL).

The Intel DPDK library can be called from Linux user space. It is first compiled, linked and then loaded into the memory just like any other Linux programs. The engineers of Lanner Electronics are able to test its effectiveness by implementing it with Snort.

Snort is a sniffer, packet logger, and network intrusion detection and prevention system. The use of Snort is licensed under GNU General Public License (GPL).

Using Snort can be an effective and convenient way to test packets processing capability. Snort can process the raw packets without interpretation by the OS since it has its own packet decoder to decode up the TCP/IP stack. For more information on Snort, visit: <http://www.Snort.org/>

Experiment Design

In order to obtain a quantitative comparison on how much the Intel DPDK can improve data plane processing performance, we designed the following experiment. We ran Snort on an Intel DPDK-enabled platform by using one of the sample implementations from Intel – the Multi-process example. We then compared the throughput on the platform using the Intel DPDK to the throughput on the platform without the Intel DPDK. The experiment setting is as follows:

Parameters	Snort with the Intel DPDK	Snort without the Intel DPDK
<i>Application for Testing</i>	<i>Snort (version: 2.9.1.2)</i>	<i>Snort (version: 2.9.1.2)</i>
	<i>Snort mode: In-line mode(1)</i> <i>Executing Snort with the following command and options: ./Snort --daq afpacket --daq-var dpdk -i eth3:eth5 -Q -c /etc/Snort/Snort.conf</i>	<i>Snort mode: In-line mode(1)</i> <i>Executing the Snort with the following command and options: ./Snort --daq afpacket -i eth0:eth2 -Q -c /etc/Snort/Snort.conf</i>
<i>Linux Kernel Version</i>	<i>2.6.35.6 (x86_64)</i>	<i>2.6.35.6 (x86_64)</i>
<i>Lanner Product for Device Under Test</i>	<i>FW-8875 (refer to figure 3)</i>	<i>FW-8875 (refer to figure 3)</i>
<i>Core Configuration</i>	<i>1 core running the Intel DPDK and 7 cores running Linux kernel and user space</i>	<i>all 8 cores running the Linux kernel and user space</i>
<i>Intel® Network Interface card</i>	<i>Intel 82576 with Poll Mode Driver (2)</i>	<i>Intel 82576 (NIC's driver version: igb.ko 2.4.13)</i>
<i>Intel® DPDK Version</i>	<i>Intel DPDK version: 1.0</i>	<i>N.A.</i>
<i>Throughput Measurement Equipment</i>	<i>Lanner MR-950 (with 8 GbE and Fibre combo ports) and a software traffic generator running on it</i>	<i>Lanner MR-950 (with 8 GbE and Fibre combo ports) and a software traffic generator running on it</i>

(1)The In-line mode of Snort requires real-time traffic analysis of malicious packets, and proactively blocking traffic from offending IP addresses as in Intrusion Prevention Systems (IPS).

(2)The Intel® DPDK includes Poll Mode Drivers for 1 GbE and 10 GbE Ethernet controllers which are designed to work without asynchronous, interrupt-based signaling mechanisms, which greatly speeds up the packet pipeline.

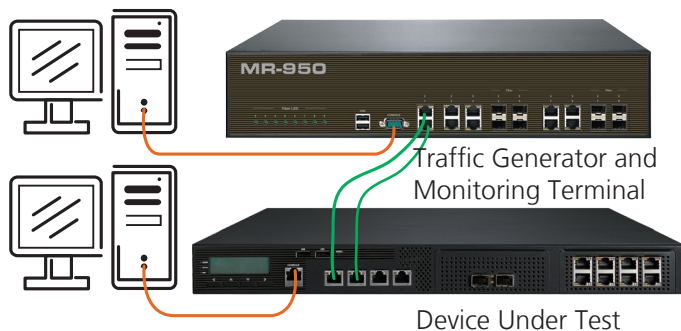


Figure 1: Test Environment

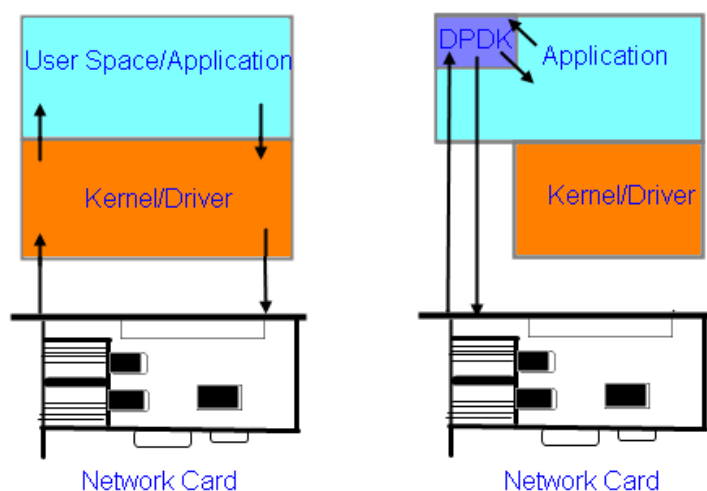


Figure 2: Linux kernel Software vs. Intel DPDK-based Data Acquisition in Snort

Figure 2 shows the software stacks in the experiment that are involved in receiving the packets from the Rx port, sending them to Snort for inspection and classification and routing them to the designated Tx port. The left side of the figure shows the unmodified version of the packet acquisition flow, while the right side of the figure shows the modified data acquisition for our experiment with Snort. The following highlights the key differences between these two configurations:

1. In the test with the Intel DPDK, the PMD (Poll Mode Driver) access the Rx and Tx descriptors directly without hardware interrupts to quickly receive, process and deliver packets. In contrast, the interrupt method was used in the original Linux kernel software test without the Intel DPDK.

Lanner Electronics has an acclaimed history of designing and manufacturing server-grade network platforms utilizing the high-performing Intel® processors and the highly integrated Intel® network Interface cards.

The Lanner FW-8875 is equipped with the quad-core Intel® Xeon® 5500/5600 series with Intel® I/O Controller Hub 10 (ICH10). The system supports up to a maximum of 24GB tri-channel DDR3 in 6 DIMMs. It also features 20 network ports via 3 optional modules. For more information, visit Lanner website at: http://www.lannerinc.com/x86_Network_Appliances/FW-8875



Figure 3: Lanner FW-8875

By using the poll mode driver, drawbacks caused by the interrupt method can be minimized. These drawbacks include the suspension of the executing task at the time the interrupt was received, causing a context switch to service the interrupt, and wasting system resources on scheduling the softIRQ handler that performs the actual work for which the interrupt was raised. The Intel DPDK provides a framework that works directly with the hardware for processing and detecting the arrival of the new frame by polling. This method of direct access avoids the overhead of requesting buffers from the Linux kernel; instead, it allocates buffers from the Intel DPDK's own memory pool with the DPDK API at the initialization time. No data copying is required from kernel buffers to user sockets and vice versa. From the beginning when a network card receives a frame to further pass it to the application network stack, the Intel DPDK framework helps eliminate kernel overhead such as context switching, scheduling, data copying and so on along the process.

2. In the case of the Intel DPDK implementation, the Snort application acquires the packets from the Intel DPDK. On the other hand, in the original data acquisition method, the received packets are passed up through the network stack in the Linux kernel.

After packets have been captured by the poll mode driver, they need to be classified. This task determines what needs to be done with incoming frames. Afterwards, the packets are decoded and interpreted by Snort's decoder. The packets will then go through many pre-processors to detect suspicious data.

After running through the preprocessors, the packets are sent to the detection engine. The engine decides whether it is the type of packet to be tested; if so, it is passed down the corresponding rule chains defined by the user (we defined it to read and alert every TCP packet). And if further forwarding is allowed, it then forwards the packet to the Intel DPDK again for transmission. Otherwise, the packet might be blocked or logged or both according to the rule set. Upon transmission, the network card transfers the packets directly from the application to its own internal lockless ring and transmits them over the link. Interrupt handling and the associated drawbacks can be spared again.

As we can see from the Intel DPDK test environment, the Intel DPDK resides in the Linux user space and processes the incoming packets outside the Linux kernel stack, hence bypassing most of the overhead of the OS networking stack. The fundamental Intel DPDK implementation concept is further discussed below.

Multiple-Process Example Setup

We utilized the multi-process example provided by the Intel DPDK to lay the foundation for the Intel DPDK implementation. The multi-process example application is an application consisting of two separate executables (server and client processes) which together perform L2 forwarding of packets. With this example, we made it possible to use the Intel DPDK in a multi-process environment where a single server process performed the packet I/O and one client process performed the packet processing workload.

In our experiment with Snort, the client process was implemented in Snort itself and the

initialization of the client process was done by the multi-process client library component of the multi-process example. The client process generally performs very little processing on the packets it receives. It reads packets in batches, and for each packet it allocates a new mbuf (a library of DPDK for supporting packet buffers for network traffic and data) and copies the packet into it, before transmitting the copied packet back to the server application.

The server process was implemented on the FW-8875 too. It was responsible for all packet I/O through the NICs used as well as the initialization of the network ports. It was also responsible for creating the shared memory pools (mbuf) for all processes to use.

The key to all communication between server and clients is to ensure that memory resources are properly shared among the processes making up the multi-process application. The following diagram illustrates the packet flow between the server and client processes.

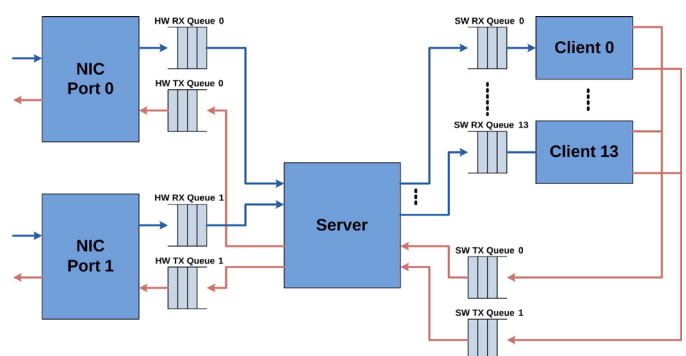


Figure 4: The packet flow between the server and client processes in a multi-process example

Test Result and Conclusion

We used a software traffic generator on Lanner MR-950 (Figure 1: Test Environment) for measuring throughput. We configured the system to generate full line-speed packets every second for 10 seconds for frame sizes 64, 128, 256, 512, 1024, 1280, and 1518 bytes.² We then calculated the percentage of throughput by dividing the number of packets received by the number of the packets transmitted. The result is shown in the

following graph. (Refer to the appendix for numerical representation)

percentage of throughput

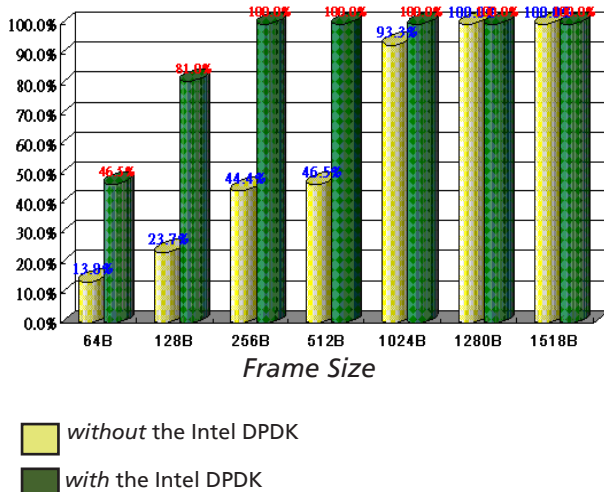


Figure 5: Percentage of throughput running on Snort with and without the Intel® DPDK

The result demonstrates that with the Intel DPDK, the packet processing capability can increase dramatically (around 3 to 4 times by throughput) on smaller packets. The throughput performance is lower for small packets due to their high arrival rate. As we can see from the Intel DPDK implementation, performance increase is achieved by separating the OS (and the packet analysis program) and the packet processing software—DPDK; each separate task is bound to a different core, too. Advanced tuning with this software architecture can be used to boost small packet performance to a desired level, i.e., a suitable number of cores can be dedicated to running the Intel® DPDK and the rest of the cores will run the OS, the OS networking stack the associated programs.

The test results demonstrate that Intel architecture platforms with multi-core processors can be an excellent vehicle for delivering high-performance and dynamically scalable data plane processing to meet today's packet processing demand.

Benefits in Network Security

The substantial increase in network performance is necessary with the expectation of providing the bandwidth required to support rapidly growing IP applications and more

complicated packet analyzing programs.

As network speeds continue to exceed the packet processing capability, some undesirable conditions such as packet loss might occur, meaning that the traffic is not 100% guarded or monitored. For applications with such stringent performance requirements, a tight coupling between the Intel DPDK and Intel architecture enables performance to scale up to 80 Mega packets per second (performance based on a future Intel® Xeon processor). Intel architecture processor combined with the Intel DPDK achieves this leap of performance by utilizing an efficient packet processing mechanism with major performance-enhancing techniques (e.g. poll mode drivers, lockless rings and zero-copy buffers) which greatly reduces CPU cycles used for data I/O and data delivery. Therefore, there will be more CPU cycles available for application processing.

Network functions that can benefit from this achievement are applications that need to perform complex packets processing at L2-L7 of the data stream on multi-line-rate Gigabit Ethernet and 10 Gigabit Ethernet. The following list shows some examples:

- Deep Packet Inspection and Flow Analysis
- Packet Classification and Policy Enforcement
- Load Balancing
- Traffic Management and QoS
- Bulk Cryptography (hardware accelerator available on high-end servers)
- Enterprise VPN
- Packet filtering
- XML encoding/decoding
- Regular Expressions (Regex) (hardware accelerator available on high-end servers)
- Compression/Decompression (hardware accelerator available on high-end servers)

Benefits in Cloud Computing

Cloud Computing is on the verge of massive technology and data rate changes for future computing and communication services. In order to realize this computing model, a

solid data communication infrastructure is inevitable. And solid data communication relies on a network appliance that is able to excel in deep packet inspection and multi-core processor technology. Deep packet inspection advances the Internet service with much more sophistication while the multi-core processor technology enriches the contents that the service delivers. Now with the Intel DPDK, it further assures ongoing significant performance improvements toward merging computing and communication models. Our server-grade platforms can host a variety of services with different deployment models encompassing public cloud, private cloud and hybrid cloud.

Lanner Electronics, a leading provider of network platforms and server-grade network modules, has developed Intel DPDK-based hardware as well as software components that allow both software vendors and application developers to accelerate network packet processing performance. We even provide driver support to some entry level chips such as 82583 and 82574L which are not supported by Intel's poll mode driver. Furthermore, because of the nature of the Intel DPDK implementation, software developers who previously relied on acquiring data from the Linux kernel may no longer need to do so. We provide application examples for writing the software to acquire data from the Intel DPDK instead of the Linux kernel.

With the Intel DPDK and Lanner drivers, we have paved a cost-effective way for network-centric applications which require superior packet processing performance to scale easily across many IA platforms without modification, while also gaining the benefits of throughput boost on all levels of these platforms.

About the Author

Nick Yeh

Software Division Director

Nick is currently responsible for leading software research and development on all of the embedded systems at Lanner. Since joining Lanner Electronics in 2004, he has held several different management positions within the company including BIOS and Firmware Development Manager and RISC-based Product Development Manager.

- 1) Hardware-assisted accelerators are only available on high-end server platforms.
- 2) For a system (like FW-8875) said to be performing at wire-speed, it will need to process 1488095.24 packets per second:
 $1000000000 / ((8 + 64 + 12) \times 8)$ packets per second at 64 byte frame.

Appendix: Throughput Data

The original throughput data is as follows:

With the Intel DPDK

Packet Size	Number of Packets Received	Number of Packets Transmitted by traffic generator	Percentage of Throughput (received/transmitted)
64 byte	13865155	29761904	46.5%
128 byte	13707502	16891890	81%
256 byte	9057970	9057970	100%
512 byte	4699248	4699248	100%
1024 byte	2394636	2394636	100%
1280 byte	1923076	1923076	100%
1518 byte	1625486	1625486	100%

Without the Intel DPDK

Packet Size	Number of Packets Received	Number of Packets Transmitted by traffic generator	Percentage of Throughput (received/transmitted)
64 byte	4123084	29761904	13.8%
128 byte	4006035	16891890	23.7%
256 byte	4028825	9057970	44.4%
512 byte	2185714	4699248	46.5%
1024 byte	2234486	2394636	93.3%
1280 byte	1923076	1923076	100%
1518 byte	1625486	1625486	100%

About Lanner Electronics Inc.

Founded in 1986 and publicly listed (TAIEX 6245) since 2003, Lanner Electronics, Inc. is an ISO 9001 certified designer and manufacturer of network application platforms, network video platforms and applied computing hardware for first-tier companies. Lanner's expertise also extends to include driver and firmware support, enabling customers to optimize hardware and software communication to achieve faster time to market. With headquarters in Taipei, Taiwan and branches in the U.S. and China, Lanner is uniquely positioned to deliver custom technical solutions with localized, value-added service.

Worldwide Offices

Taiwan - Corporate Headquarters

Lanner Electronics Inc.
7F, 173, Section 2, Datong road
Xizhi District, New Taipei City 221
Taiwan
T: +886-2-8692-6060
F: +886-2-8692-6101
E: sales@lannerinc.com

USA

Lanner Electronics (USA) Inc.
41920 Christy Street
Fremont, CA 94538
USA
T: +1-510-979-0688
F: +1-510-979-0689
E: sales_us@lannerinc.com

Canada

LEI Technology Canada Ltd
450 Matheson Blvd E, Unit 40
Mississauga, L4Z 1N8, On
Canada
Toll_free: +1 877-813-2132
T: +1 905-361-0624
E: sales_ca@lannerinc.com

China

First Floor, Xingtianhaiyuan Building,
West First Street Shucun
Agriculture University South Road
Haidian District, Beijing , 100193
P.R.China.
T: +86-10-82795600
F: +86-10-62963250
E: sales_bj@lannerinc.com