Direct Network Prototype Leveraging Light Peak Technology

Sreenivas Addagatla, Mark Shaw, Suyash Sinha
Microsoft Research
e-mail: <first-name>.<last-name>@microsoft.com

Prashant Chandra, Ameya S.Varde, Michael Grinkrug
Intel Labs
e-mail: <first-name>.<last-name>@intel.com

Abstract—**Light Peak is a new optical interconnect technology developed by Intel and targeted at connecting computing and consumer electronic devices. It provides bandwidth of 10Gbps and higher using optical fiber technology, but at extremely low cost. This paper investigates whether Light Peak technology can be leveraged to form the basis of a low-cost, high performance, scalable network. This paper presents our feasibility analysis, supported by a prototype network we constructed. The prototype consisted of a number of Light Peak PCI-Express cards, each with one host interface, an integrated switch, and transceiver pair with four optical ports. With very small forwarding delays, Light Peak supports direct networks with topologies that have interesting traffic characteristics suitable for small-scale clusters (containers with 1,000-10,000 servers).**

Keywords-Direct networks, Light Peak, PCI Express

I. INTRODUCTION

The importance of creating scalable, powerful, and efficient data centers continues to increase, and many designs are based on assembling the data center from modules (e.g. racks or containers comprised of many servers) [8]. However, these modular designs depend critically on having a low-cost, high-bandwidth network to connect the servers inside each module and connect the modules together. We present our design and prototype for a scalable, low-cost network for modular data centers that leverages commodity 10Gbps Light Peak technology.

A. Light Peak Technology

Light Peak [1] is a high-bandwidth, inexpensive optical interconnect technology developed by Intel to connect computing and consumer electronic devices. Some of the salient features of Light Peak are:

- The use of small packet sizes to achieve low latency.
- The ability to multiplex multiple I/O protocols over a common link with high bandwidth efficiency.
- The use of connection oriented approach and hierarchical addressing to scale to large number of connected nodes with small switching table sizes.
- The use of credit based flow control (inspired by [3]) to achieve small buffer sizes.
- The ability to flexibly allocate link bandwidth using priority and bandwidth reservation mechanisms.

B. Direct Networks

Direct Networks [4] are used within high-performance parallel processing clusters; typically limited to processor interconnects that operate at much higher speeds than regular host-to-host links. Connecting a “node” (processor) directly to a set of other nodes obviates the need for aggregating switches with very high cumulative capacity (as in “indirect networks”), and provides network capacity bundled with the servers themselves.

Direct networks thereby stand to benefit from gains in integration of components on a common system on the chip, and from the volume economics seen in the server market. In direct networking, small switches are integrated into the server nodes, with very small forwarding tables, buffers and power requirements. Also, common direct network topologies offer better resiliency than many indirect networks, due to the availability of many paths between nodes and distributed switching throughout the network.

C. Motivation

Although originally targeted at consumer scenarios of connecting devices and computers, Light Peak has several characteristics that make it appealing for use as a direct network interconnect, in a small local area network, as in a data center modular unit:

- Light Peak supports generic graph topologies for switch interconnections (unlike tree topologies for other interconnects such as USB or PCI Express).
- The optical medium is high-bandwidth, offering up to 10Gbps of throughput in each direction, with cables lengths of up to 100m, and planned upgrades to higher data rates in future.
- Packet forwarding at an intermediate switch can be performed entirely in hardware without interrupting the attached host: forwarding delays are expected to be very small.
- The ability to flexibly allocate link bandwidth can be used to implement performance isolation across multiple host interactions.

Current advances in Silicon Photonics [2] allow integration of optics on the server node at an extremely low price points. Yet, there are many other hardware costs in switches – high speed data structures, and packet buffers. The Light Peak architecture drives these costs down too – using the bare minimum in buffering and tiny tables to

978-0-7695-4208-9/10 $26.00 © 2010 IEEE
DOI 10.1109/HOTI.2010.13

control packet flow. This assures that the all-up costs are indeed low.

However, the fundamental question is then can the resulting network built leveraging Light Peak technology achieve high performance? In this paper, we take a first step towards answering this question by constructing a prototype direct network and evaluating its performance using TCP/IP traffic. Of many possible direct network topologies, Cayley graphs [6] with fixed degree (e.g., wrap-around butterfly topologies [7]) seem to be most applicable for a network interface prototype with an integrated 4-port switch, offering high bisection bandwidth.

D. *Network Interface Prototype*

We built a prototype network interface card using Light Peak technology (shown in Figure 1) The prototype card is a Gen2 x4 PCI-Express add-in card and contains one host interface, an integrated crossbar switch and transceiver pair with four 10 Gbps optical ports with modified USB cable connectors. The integrated non-blocking crossbar switch is capable of delivering an aggregate bandwidth of 80 Gbps (40 Gbps receive and 40 Gbps transmit) through the optical ports and 10 Gbps to/from the host system. Traffic from one optical port to another optical port can be transmitted directly without any interaction with the host CPU. Each transceiver module supports two interfaces and provides electrical to optical conversion.

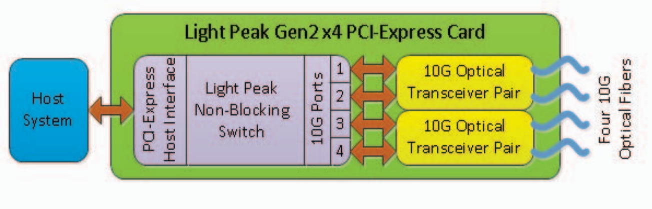
Light Peak Gen2 x4 PCI-Express Card
Host System
PCI-Express Host Interface
Light Peak Non-Blocking Switch
10G Ports
1
2
3
4
10G Optical Transceiver Pair
10G Optical Transceiver Pair
Four 10G Optical Fibers

Figure 1. Schematic of Light Peak hardware prototype

II. Software Architecture

Light Peak employs a connection-oriented link-layer transport protocol, where in paths must be configured prior to data transmission. Correspondingly, the notion of connection management is a primary functionality of the control plane, to support the data plane (sending and receiving traffic) of a Light Peak network. In addition, basic performance and fault management functions are needed as well.

The software to provision and run a network prototype based on Light Peak has the following logical components: (see Figure 2):

- Connection Manager
- Device Driver
- Link/Switch Status Monitor
- Failover Manager

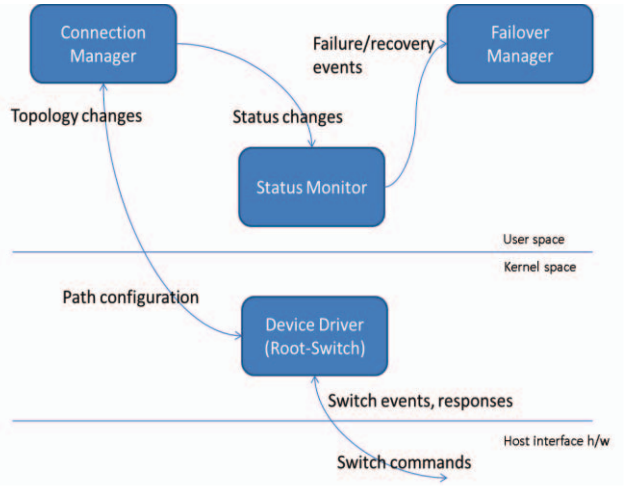

Figure 2. Key software component interactions of the prototype

A. *Connection Manager*

A network of interconnected Light Peak switches are grouped into a set of "domains," each of which is managed by a software component called "Connection Manager." The Connection Manager may be administratively associated with one of the domain member interfaces (called a "Root-Switch") and is responsible for device enumeration, path configuration, QoS and buffer allocations at the switches in its domain. The Light Peak protocol allows the Connection Manager of the domain to be changed without impacting previous configuration and data transfer.

Starting from the Root Switch, the Connection Manager enumerates each switch in the domain, building a topology graph. The Connection Manager also gets notified of topology changes caused by hot-plug and hot-unplug events. After initial enumeration, the Connection Manager must configure paths to enable data communication between nodes. Path configuration may be performed at initialization time or on-demand based on traffic patterns.

Multiple domains may be interconnected in arbitrary fashion. Light Peak configuration protocol provides primitives that enable communication between the CMs in adjacent domains, and the Connection Managers of the connected domains may exchange information with each other to perform inter-domain configuration of paths. Figure 3 illustrates an example of inter-domain connection.

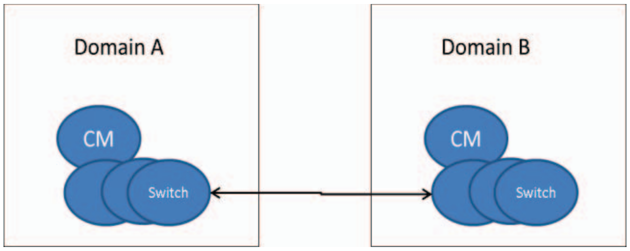

Figure 3. Light Peak Domains and Connection Manager

B. *Device Driver*

A Device Driver component for the Light Peak host interface is responsible for sending and receiving network traffic. In the prototype, the device driver is an operating system kernel component that interacts with the TCP/IP subsystem on the host on one side and communicates with the host interface. Also, it is responsible for the initialization, configuration updates and shutdown of the Light Peak host interface and associated crossbar switch.

The Light Peak host interface provides access to the device's status registers and can read/write to areas of host's memory using DMA. The host interface implements support for a pair of producer-consumer queues (one for transmit, one for receive) for each configured path. The host interface also presents a larger protocol data unit that can be used by software to send and receive data.

In addition to interfacing with the operating system TCP/IP stack, the device driver also exports a direct interface to send and receive data directly from user space (e.g., by the Connection Manager).

C. *Link/Switch Status Monitor*

The Status Monitor component gets updates from the Connection Manager with events related to the Light Peak interface and link failures within its domain. The Status Monitor instructs the Connection Manager to implement various recovery and rerouting strategies as appropriate.

In addition, the Status Monitor can also collect performance indicators from each Light Peak switch for network performance monitoring and troubleshooting purposes.

D. *Failover Manager*

In general, a failure at a domain's Root-Switch (i.e., Connection Manager itself) does not affect traffic already in transit, but subsequent link/switch failures require updates to path tables at every switch. The Failover Manager selects and assigns a new Connection Manager in the event of Root Switch failures. The selection can be administrative or based on a consensus algorithm.

When multiple domains are involved, a failure affecting inter-domain traffic requires messaging across corresponding Connection Managers.

III. TESTBED AND IMPLEMENTATION ON MICROSOFT WINDOWS®

Figure 4 shows an example configuration of our test bed, with 16 hosts and 20 Light Peak PCI Express card prototype switches. Each host has a 4-core Intel® Xeon® E5540 CPU, running Microsoft Windows® Server 2008 R2 operating system. To verify that the transit traffic does not interrupt a host's CPU, four of the hosts contain two Light Peak cards each, one of which is configured not to source or receive traffic into the host.

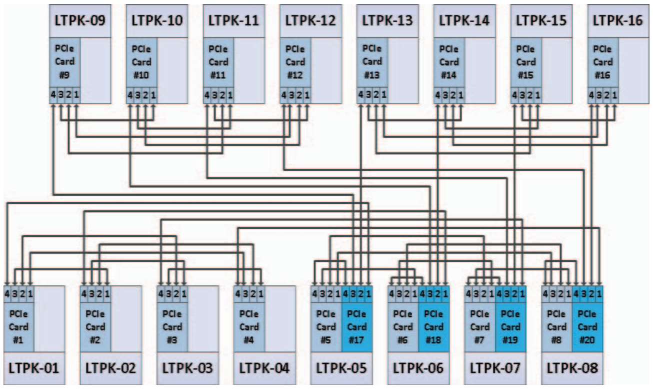

Figure 4. Example network layout with 16 hosts and 20 switches

The Device Driver implementation on Microsoft Windows® follows the NDIS 6.20 connectionless miniport driver model [5], with a network layer Maximum Transmission Unit (MTU) of 4096 bytes. The driver maps a set of DMA buffers as a circular queue pair (one for transmit side, one for receive side) for each of the configured paths.

For sending, the driver collects packets from the TCP/IP subsystem, and selects a transmit queue based on the destination IP address, and adds the packet to the queue. For receiving, a packet is removed from a receive queue and forwarded to the TCP/IP layer. The arrival of a packet in the receive queue, completion of a buffer transmission, as well as a receive queue being full are indicated as interrupt events to the driver. With our prototype implementation, we achieve 5.5 Gbps transmit and 7.8 Gbps receive throughputs from each host.

The Connection Manager, in addition to link layer path configuration, also implemented IP address assignment to hosts. Since Light Peak prototype interfaces lacked a globally unique identifier (such as an Ethernet MAC address), we used a globally unique identifier for the host (computer name) along with a locally unique identifier for the Light Peak network interface as a basis for IP address assignment.

IV. FUTURE WORK

This paper is an initial report on the feasibility of leveraging Light Peak technology to construct data center networks. Some of the important research questions that must be addressed in the future include: a) implementation of robust failover and fast rerouting mechanisms, b) interworking with conventional Ethernet infrastructure in the data center and c) packaging optimizations and topologies for data center modular units that can reduce cabling complexity and increase compute density.

V. SUMMARY

Compute density continues to increase in the data center driven by increasing numbers of lower cost servers. With increasing compute density, the importance of the data center network rises, as does the network's fraction of the total cost. We expect this will result in tighter integration of networking components onto server platforms. Light Peak technology appears well placed to benefit from these trends. Our prototype implementation built leveraging Light Peak technology shows that it is possible to construct direct networks with modern operating systems and networking stacks that can deliver high application-level performance with significantly lower complexity and power consumption than conventional counterparts. Light Peak appears to be a promising bet for the data center.

Acknowledgment

We thank Jim Larus, Burton Smith, Albert Greenberg, and Dave Maltz from Microsoft Research, and Kevin Kahn, Jim Held and Jason Ziller from Intel Labs for help with the project efforts and reviewing this paper before submission.

References

[1] Intel Corporation, "Light Peak Technology," http://www.intel.com/go/lightpeak/index.htm.

[2] Intel Corporation, "Silicon Photonics Technology", http://www.intel.com/go/sp/.

[3] T. Blackwell, K. Chang, H.T. Kung and D. Lin, "Credit-Based Flow Control for ATM Networks", IEEE Network, 1995.

[4] L. M. Ni and P.K. Mckinley, "A Survey of Wormhole Routing Techniques in Direct Networks," IEEE Computer, 1993.

[5] Microsoft Corporation, "NDIS Miniport Drivers," http://msdn.microsoft.com/en-us/library/ff565949%28VS.85%29.aspx.

[6] M. Heydemann, "Cayley Graphs and Interconnection Networks," in Graph Symmetry: Algebraic Methods and Applications, pp. 167-224, 1997.

[7] M. D. Wagh and O. Guzide, "Mapping Cycles and Trees on Wrap-around Butterfly Graphs," SIAM Journal on Computing, Volume 35, Issue 3, pp. 741-765, 2005.

[8] J. Hamilton, "An Architecture for Modular Datacenters," 3rd Biennial Conference on Innovative Data Systems Research (CIDR), 2007.