GreenLight Technical Report

2010

Virtualization

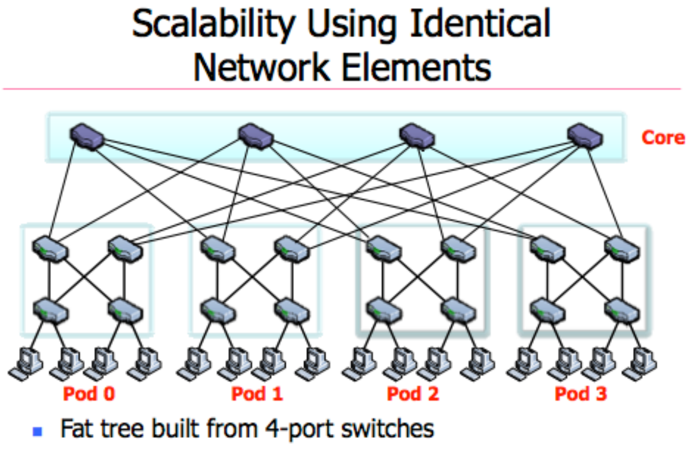
Amin Vahdat: Conserving Resources Through Virtualization

Research Activities

We have been active in investigating the network requirements for emerging data center architectures as well as virtualization techniques to increase the level of multiplexing and energy efficiency available from emerging container-based compute infrastructures. On the first front, we are considering GreenLight as an exemplar of emerging compute and storage containers supporting 1000-4000 individual processors and multiple petabytes of storage. This computation will be striped across multiple thousands of discrete machines each capable of delivering 10 Gb/s or more of network bandwidth across emerging 10 Gb/s switches. The ability to leverage all available computing and storage in these containers will hinge upon the ability to deliver low-latency and high bandwidth communication between individual elements. Our goal is to build a modular multi-stage network switch capable of delivering full bisection bandwidth (e.g., 20+ Tb/s in aggregate across 2,000 individual cores).

One of the hypotheses behind this work is that for emerging large-scale computations, network bandwidth is often the primary bottleneck in large-scale computations. Essentially, applications such as data mining, high-performance computing, and large-scale Internet services (search, map, email, social networking) run in parallel on thousands of servers with significant required communication bandwidth among the nodes. When there are significant network bottlenecks, CPU, memory, and disk is left idling waiting on data to transmit to or arrive from remote nodes.

To address these bandwidth bottlenecks, we designed a switch built around a fat-tree topology and emerging high-port count 10 Gb/s Ethernet switch silicon as the basis for GreenLight networking. Figure 1 below depicts the details of the topology, an instance of the Clos [Clos53] topology. The basic idea is to build a large-scale rearrangeably nonblocking switch architecture from constituent smaller-scale switches. In the example figure, we build a 16-port switch from constituent 4-port switches organized into 4 pods interconnecting 4 hosts each. While Figure 1 below depicts a small-scale instantiation, our topology has attractive scaling properties. In general, 5k2/4 k-port switches can deliver nonblocking bandwidth to k3/4 ports. Today, 64-port 10 Gb/s commodity switches on a chip are becoming available. Considering this topology for the k=64 case, 65,536 ports would be organized into 1,024 pods each, with the potential for nearly 1 Pb/sec of aggregation bisection bandwidth among all ports.


Scalability Using Identical Network Elements

- Fat tree built from 4-port switches

On the virtualization front, we completed our implementation and evaluation of Difference

Engine, to mitigate perhaps the largest barrier to higher degrees of multiplexing in datacenter deployment environments. In current deployments, CPU and network utilization on a per-virtual machine basis is

highly bursty. This allows data center engineers to leverage statistical multiplexing to more efficiently utilize available energy, compute, and network resources in the data center. Where previously machines that might have average utilization of 10% but bursty utilization to 80-100% had to have their own dedicated physical machines, virtualization allows for higher degrees of consolidation by assuming that periods of high activity will typically not be correlated and that when they are, VM migration can be leveraged to ease hot spots. However, the total amount of memory available on a physical machine limits the maximum degree of multiplexing. With current virtualization technology, each VM must be allotted its full amount of physical memory regardless of its dynamically changing behavior. Our work on Difference Engine leverages the observation that many physical memory pages are shared across VM's running similar operating systems and applications. Difference Engine dynamically and efficiently locates these pages and collapses them to a smaller physical memory footprint on individual physical machines. Difference Engine takes this process one step further by efficiently finding similar but not quite identical pages and collapsing them to base pages plus a compact representation of the difference. We perform all of this work independent of the operating system or the application, enabling us to transparently increase the available degree of multiplexing by a factor of 2-3 for a range of workloads.

References: A Study of Non-blocking Switching Networks, C. Clos, *Bell System Technical Journal*, 32(2), 1953.
PortLand: A Scalable Fault-Tolerant Layer 2 Data Center Network Fabric, Radhika Niranjan Mysore, Andreas Pamboris, Nathan Farrington, Nelson Huang, Pardis Miri, Sivasankar Radhakrishnan, Vikram Subramanya, and Amin Vahdat, *Proceedings of the ACM SIGCOMM Conference*, Barcelona, Spain, August 2009.
Data Center Switch Architecture in the Age of Merchant Silicon, Nathan Farrington, Erik Rubow, and Amin Vahdat, *Proceedings of the IEEE Symposium on Hot Interconnects*, August 2009.
Difference Engine: Harnessing Memory Redundancy in Virtual Machines, Diwaker Gupta, Sangmin Lee, Michael Vrable, Stefan Savage, Alex C. Snoeren, George Varghese, Geoffrey M. Voelker, and Amin Vahdat, *Proceedings of the 8th ACM/USENIX Symposium on Operating System Design and Implementation (OSDI)*, San Diego, CA, December 2008.
A Scalable, Commodity, Data Center Network Architecture, Mohammad Al-Fares, Alex Loukissas, and Amin Vahdat, *Proceedings of the ACM SIGCOMM Conference*, Seattle, WA, August 2008.

Research Findings

The tremendous scale out in compute and storage in data center environments has left a significant hole in the ability of scalable network fabrics to keep up with emerging all-to-all communication patterns. Thus, high-performance networking is moving beyond traditional parallel/supercomputing environments as a key requirement in data center environments. Our work on scaling network bandwidth with low latency, energy consumption, and built-in fault tolerance is finding significant interest from companies such Hewlett Packard, Broadcom, Microsoft, Yahoo!, Facebook, Fulcrum, Cisco, and Juniper. We are investigating models of collaborating with these companies to build next-generation switch architectures.

Our work on Difference Engine has already been picked up by Citrix, one of the leading vendors of virtualization technologies (selling the popular Xen virtual machine monitor). Based on our development and source code release for an earlier version of Xen, we expect to see our modifications available in the commercial version of Xen in a future release.

For more information please contact Amin Vahdat, mvahdat@ucsd.edu