Cisco and Solarflare Achieve Dramatic Latency Reduction for Interactive Web Applications with Couchbase, a NoSQL Database

What You Will Learn

The needs of modern interactive applications have changed dramatically in recent years. Today NoSQL databases are the database of choice for many high-throughput Web 2.0 applications.[1] This transition has occurred primarily because of the schema rigidity associated with traditional relational databases as well as their inability to scale out. In contrast, NoSQL databases can provide high operations-per-second throughput because of their distributed architecture. Distributed architecture and high throughput place different challenges on the network compared with those seen in traditional relational database management system (RDBMS) databases. The read and write latency of NoSQL databases is very low because the data is shared across nodes in a cluster and the application's working set is stored in memory. However, the latency induced by kernel overhead can increase the overall latency. Hence, to take full advantage of the power of NoSQL databases, network designers must consider how to optimize for lower, consistent latency for the total network stack by examining and optimizing the latency of each contributing element, especially the kernel, the TCP stack, and server I/O.

This document describes how a solution based on Couchbase Server, Cisco Nexus® Family switches, and the Solarflare 10 Gigabit Ethernet server adapter with OpenOnload can achieve the lower latencies required by today's interactive Web 2.0 applications.

Goals of the Test

NoSQL databases can achieve low-latency access by integrating a caching tier as part of the database and managing the working set of applications in memory. This caching tier reduces the volume of data being accessed from the storage tier of the database, so network latency between the application and the database becomes an important performance metric. By reducing this latency, an application can improve the overall user experience as well as achieve higher throughput. Since many Web 2.0 deployments today are still running mostly 1 Gigabit Ethernet links at the server access edge, the goals of this test were to:

- Determine the latency reductions first by moving from 1 Gigabit Ethernet to 10 Gigabit Ethernet
- Establish a 10 Gigabit Ethernet baseline was established and then determine further latency reductions possible by implementing Solarflare OpenOnload

Overview of NoSQL

Traditional databases, such as RDBMS databases, have high application-level latency due to schema complexity and a scale-up architecture. As the number of users grows and throughput requirements increase, bigger and more expensive servers (with more CPU, more memory, and variable I/O performance) are needed to keep up with demand. NoSQL databases, in contrast, have very low application latency and high throughput because of their distributed, in-memory architecture. The database tier scales horizontally along with the application server tier to address the ever-growing throughput needs of applications. These characteristics suit today's Web 2.0 applications particularly well because the number and activity level of users are difficult to predict.

In addition to scaling out, the growing amount of unstructured data, such as social media feeds and blogs, has created a need for databases that support schema variability. Schema changes in relational databases can be extremely difficult to implement, making it difficult to rapidly iterate and push out application changes to meet market demands. NoSQL databases, particularly document databases, provide flexibility for evolving schemas without forcing existing data to be restructured.

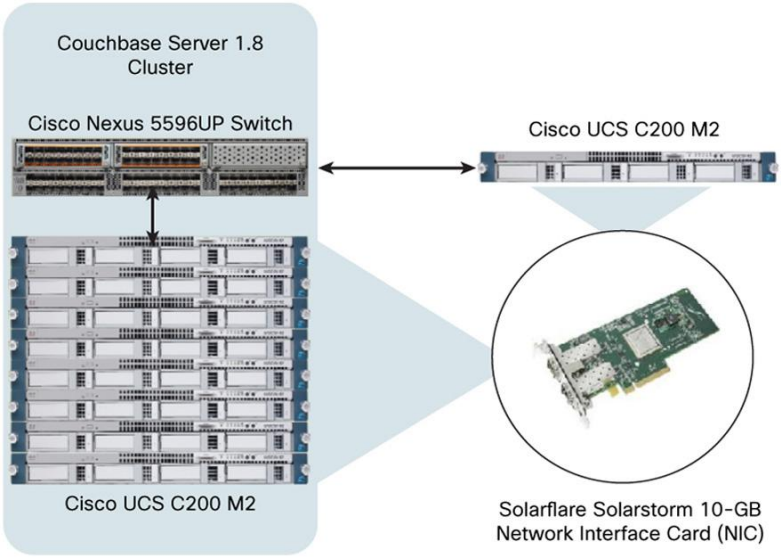
[1] http://blog.couchbase.com/how-couchbase-helped-omgpop-break-all-records-draw-something

NoSQL databases, at the most abstract level, provide a key-value store with no enforced structure on the value that is stored. Couchbase Server is an open source NoSQL database that automatically distributes data across commodity servers. Built-in object-level caching enables applications to read and write data with latency of less than a millisecond. Further, with no schema to manage, Couchbase Server effortlessly accommodates changing data management requirements without any application downtime. To help ensure high availability of the application, particularly to protect against node failures, the Couchbase Server data replication capability maintains multiple copies of the data in a cluster.

Architecture Overview

In these tests, eight servers were designated as Couchbase Servers, and one node was used as the load generator. The workload was a mixed load with 70 percent read operations and 30 percent write operations, with the working set in memory. More details about the workload and the workload generator are provided later in this document. All the servers and the load generators were connected to a Cisco Nexus 5596UP Switch using a Solarflare 10 Gigabit Ethernet server adapter with OpenOnload (Figure 1).

Figure 1: Test Architecture



Solarflare OpenOnload is an open source, high-performance network stack for Linux created by Solarflare. Solarflare OpenOnload performs network processing at the user level and is binary compatible with existing applications that use TCP and User Datagram Protocol (UDP) with Berkeley Software Distribution (BSD) sockets. It consists of a user-level shared library that implements the protocol stack, and a supporting kernel module.

Solarflare OpenOnload achieves dramatic performance improvements in part by performing network processing at the user level, bypassing the OS kernel entirely on the data path. By improving the host CPU efficiency, Solarflare OpenOnload enables applications to uses more server resources, resulting in dramatically accelerated application performance without any need to rewrite applications or change the existing Ethernet and TCP/IP infrastructure.

The Solarflare OpenOnload library can be configured to enable the application threads to run without contention with each other, and without requiring any interrupt processing or other interaction with the kernel active on the dedicated core. These techniques implemented in Solarflare OpenOnload enable a system running an application to be entirely vertically separated on CPU cores, thus avoiding any application-level crosstalk. Networking performance is improved without sacrificing the security and multiplexing functions that the OS kernel normally provides. Moreover, applications can choose at runtime to use Solarflare OpenOnload or the standard Linux kernel stack.

For the NoSQL test, Solarflare OpenOnload was used to improve the latency performance of the entire stack.

Test Configuration and Methodology

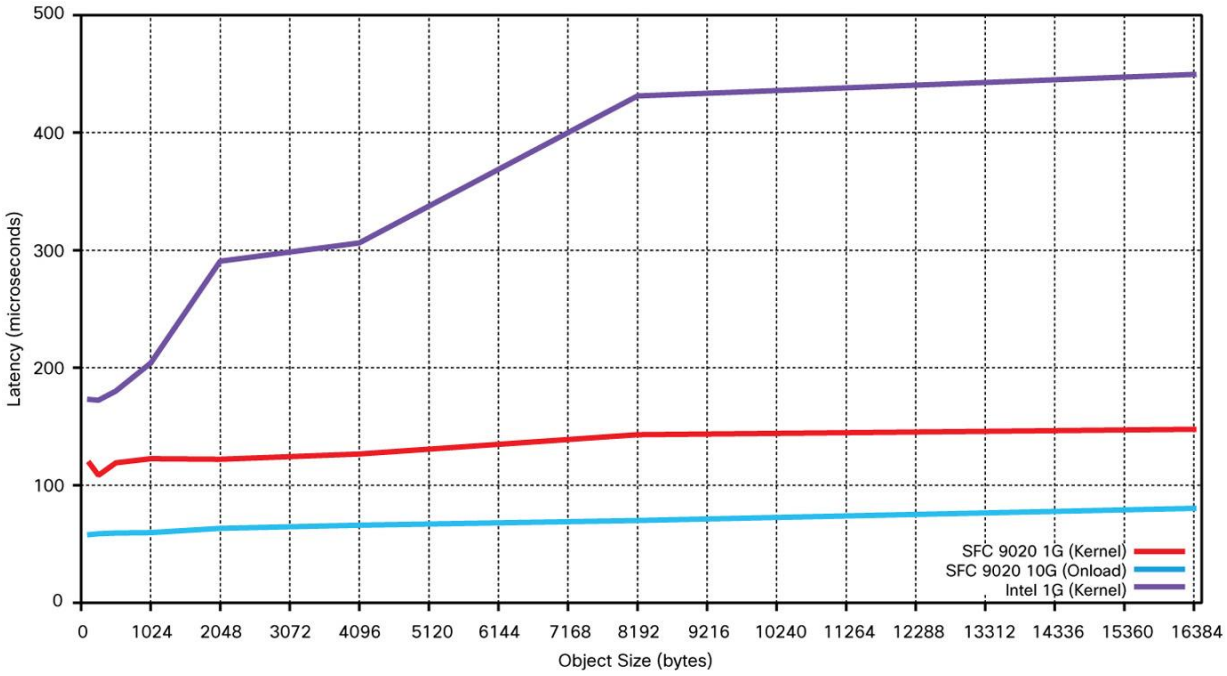
Eight nodes running Couchbase Server 1.8.0 were used as NoSQL database servers, and one node was used to run the tests. The benchmark used for the test was Couchbase's streaming load generator for key-value stores. Two sets of GET and SET operation-based tests were performed comparing latency using the kernel and Solarflare OpenOnload.

Test System and Parameters

- Couchbase Server 1.8.0
- Cisco Nexus 5548UP Switch
- Solarflare SFN5122F 10 Gigabit Ethernet Enhanced Small Form-Factor Pluggable (SFP+) server adapters
- Solarflare OpenOnload
- Servers: Nine Cisco UCS C200 M2 High-Density Rack Servers with Intel Xeon processor X5670 six-core 2.93-GHz CPU, running Red Hat Enterprise Linux (RHEL) 5.5 x86 64-bit, with 100-GB RAM and four 2-TB hard drives

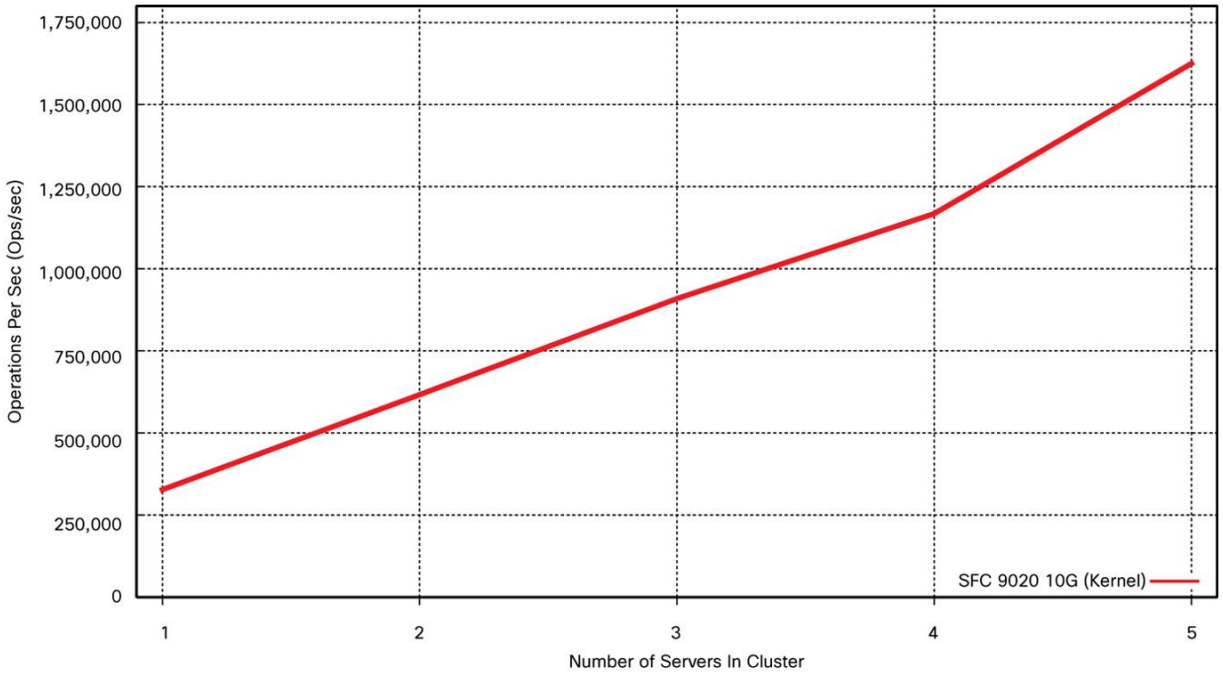
The data presented in Figure 2 shows that by moving from 1 Gigabit Ethernet to 10 Gigabit Ethernet, latency was reduced by 50 percent. For example, with object size 1024 bytes, 1 Gigabit Ethernet latency was 203 microseconds. With 10 Gigabit Ethernet, latency for 1024 bytes was 122 microseconds. By implementing Solarflare OpenOnload, latency was reduced another 50 percent, down to 59 microseconds. The latency represented on the chart is an average of GET and SET latencies across multiple independent runs measured on the client side. Furthermore, the 10 Gigabit Ethernet system is much more scalable, with latency remaining low even as object size was increased from 128 to 16384 bytes.

Figure 2: Latency Compared to Object Size: 1 Gigabit Ethernet, 10 Gigabit Ethernet Kernel, and Solarflare 10 Gigabit Ethernet Server Adapter with OpenOnload



Although the test scenario was not set up to optimize for maximum throughout, Figure 3 shows that by moving from 1 Gigabit Ethernet to 10 Gigabit Ethernet, the number of operations per second increases to nearly 2 million. A mixed workload of GET and SET operations with 70 percent GET operations and 30 percent SET operations was used on 1-KB documents.

Figure 3: Throughput



Summary of Findings

Industry experts report that more than 15 percent of Web 2.0 server farms are typically dedicated to memcached, a widely adopted caching technology and a precursor to NoSQL databases. As the adoption of NoSQL databases continues to grow, reducing the latency of database requests will become critical to companies deploying Web 2.0 applications.

The results from this test demonstrate that latency can be reduced by upgrading the network infrastructure from a 1 Gigabit Ethernet to a 10 Gigabit Ethernet network. Furthermore, by deploying a Cisco Nexus 10 Gigabit Ethernet switch and Solarflare 10 Gigabit Ethernet server adapter with Solarflare OpenOnload, database latency can be reduced an additional 50 percent even when the network is loaded with large messages.

Details of the Workload

The Python-based open source tool mcsoda was used as the workload generator. A mixed load of GET and SET operations with a ratio of 70:30 was run on the Couchbase Server cluster over varying document sizes. Examples of the commands used to measure latency and throughput of the system are shown here:

```
mcsoda.py http://<host>:8091 cur-items=100000 max-items=100000 min-value-
size=<document size> max-ops=1000000 threads=1 ratio-sets=0.3 batch=0 histo-
precision=2
```

```
mcsoda.py http://<host>:8091 cur-items=100000 max-items=100000 min-value-
size=1024 max-ops=1000000 threads=1 ratio-sets=0.3 batch=50 vbuckets=1024 histo-
precision=2
```

More information about the workload generator can be found at https://github.com/couchbase/testrunner/blob/master/pytests/performance/README.md.

Cisco Nexus 5000 Series Switches

Cisco Nexus 5000 Series Switches simplify data center transformation with innovative, standards-based, high-performance Ethernet and a unified fabric server access layer. Current-generation data centers are increasingly dense and multicore. The Cisco Nexus 5000 Series meets business, service, application, and operation requirements of such data centers. Cisco Nexus 5000 Series Switches provide:

- High-performance, low-latency 10 Gigabit Ethernet, delivered by a cut-through switching architecture, for 10 Gigabit Ethernet server access in next-generation data centers
- Fibre Channel over Ethernet (FCoE)-capable switches that support emerging IEEE Data Center Bridging (DCB) standards to a deliver a lossless Ethernet service with no-drop flow control
- Unified ports that support Ethernet, Fibre Channel, and FCoE
- A variety of connectivity options: Gigabit, 10 Gigabit (fiber and copper), FCoE, and Fibre Channel
- Converged fabric with FCoE for network consolidation, reducing power and cabling requirements and simplifying data center networks, especially for SAN consolidation of Fibre Channel

Solarflare SFN5122F Dual Port SFP+ 10 GbE Server Adapter

The test used the Solarflare SFN5122F 10 GbE SFP+ Server Adapter and OpenOnload application acceleration middleware. The tests were performed using the standard Linux kernel TCP/IP stack as well as Solarflare OpenOnload. Solarflare OpenOnload is an open source high-performance network stack for Linux created by Solarflare. Solarflare OpenOnload performs network processing at the user level and is binary compatible with existing applications that use TCP and UDP with BSD sockets. It includes a user-level shared library that implements the protocol stack, and a supporting kernel module.

IT managers designing high-speed, low-latency networks need compatibility with the control planes and data paths of TCP/IP, Ethernet, and the Linux operating system. With Solarflare OpenOnload, designers of high-performance networks get access to all the necessary Layer 2 and Layer 3 features, such as virtual LAN trunking for the management, separation, and security of IP traffic flows; link aggregation for failover and redundancy; Address Resolution Protocol (ARP) caching for high-speed mapping of Ethernet to IP addresses; and Internet Control Message Protocol (ICMP) notifications for network diagnostics and resiliency.

Because Solarflare OpenOnload provides true transparency to POSIX sockets, network and application architects do not need to rewrite their applications to get the performance increase that the software provides. Solarflare's solution transparently supports the Linux networking functions required by scale-out computing applications, such as poll, select, epoll, fork, exec, signals, and comprehensive socket flags. The middleware also includes optimizations for concurrent socket access by multithreaded applications, as well as support for fine-grained latency measurements, diagnostics, and per-socket performance controls.

Solarflare server adapters provide the highest possible line-rate performance, excel in small-message processing, and have demonstrated performance leadership in the most demanding application environments. Solarflare 10 Gigabit Ethernet server adapters provide the following benefits:

- PCIe 2.0 and 40 Gbps of bidirectional bandwidth
- Offload of critical computation-intensive tasks to help ensure that the least possible burden is placed on the server CPU, freeing processor cycles for customer applications
- Full compatibility with Solarflare OpenOnload

About Couchbase Server

Couchbase Server is a NoSQL database optimized for the data management needs of interactive web applications. Couchbase scales horizontally and automatically distributes data and I/O across commodity servers or virtual machines. It supports live cluster topology changes while continuing to serve data to the application. Built-in object caching technology delivers consistent low-latency read and write operations while sustaining high throughput.

About Solarflare

Solarflare is the leading provider of application-intelligent networking I/O products that bridge the gap between applications and the network, delivering improved performance, increased scalability, and higher return on investment (ROI). The company's solutions are widely used in scale-out server environments such as high-frequency trading, high-performance computing, cloud computing, virtualization, and big data environments. Solarflare's products are available from leading distributors and value-added resellers, as well as from Dell, IBM, and HP. Solarflare is headquartered in Irvine, California, and operates an R&D facility in Cambridge, UK.

For More Information

- Solarflare OpenOnload: http://www.solarflare.com/Content/UserFiles/Documents/Solarflare_OpenOnload_IntroPaper.pdf
- Cisco Nexus 5000 Family switches: http://www.cisco.com/go/nexus
- Couchbase Server: http://www.couchbase.com/couchbase-server/overview

C11-708169-00 06/12