Advanced Core Operating System (ACOS): Experience the Performance

Table of Contents

Disclaimer

This document does not create any express or implied warranty about A10 Networks or about its products or services, including but not limited to fitness for a particular use and noninfringement. A10 Networks has made reasonable efforts to verify that the information contained herein is accurate, but A10 Networks assumes no responsibility for its use. All information is provided "as-is." The product specifications and features described in this publication are based on the latest information available; however, specifications are subject to change without notice, and certain features may not be available upon initial product release. Contact A10 Networks for current information regarding its products or services. A10 Networks' products and services are subject to A10 Networks' standard terms and conditions.

Trends Affecting Application Networking

The modern data center is undergoing a transformation, primarily driven by the rapid adoption of cloud and mobile computing technologies. Enterprise and service provider customers are experiencing rapid growth of data center traffic, which demands high-performance, scalable and efficient data center infrastructure. To keep up with the data traffic growth, application networking technologies have evolved significantly and delivered capabilities to improve the performance and security of data center architectures.

However, the explosion of data traffic growth from mobile, big data and Web traffic presents enormous challenges to scaling the performance of application networking systems. Ever increasing data center traffic requires increasingly higher and more predictable application networking system performance to consistently meet the expected service levels of end users in a cost-effective manner.

Over the last five years, the general-purpose CPU architectures used by application networking system vendors have migrated from single core, 32-bit CPU systems toward multicore, 64-bit CPUs. However, most application networking software architectures have not been optimized to leverage these fundamentally new CPU and system designs. As a result, some vendors have not been able to cost-effectively scale system performance to meet the pressing demands of the modern data center. Fundamentally, an architectural-level rethink is needed to scale performance of application networking systems. That architectural system redesign is uniquely available in the A10 Networks® Advanced Core Operating System (ACOS®) platform.

Application networking systems (like Application Delivery Controllers, or ADCs) implement higher level; application-aware performance optimization and security functions to improve application performance and availability in ways not possible for traditional networking systems.

The Era of Multicore

Traditional networking systems (like switches and routers) provide very high-performance network processing by leveraging custom semiconductors (ASICs) to process only limited headers of the data packet. As a result, traditional networking systems are unable to implement significant value-added functions on data traffic as it traverses these devices. Meanwhile, application networking systems (like Application Delivery Controllers, or ADCs) implement higher level, application-aware performance optimization and security functions by looking deeply into and taking actions on the unstructured data payloads deep within the data packet. In order to implement these functions, application networking systems are dependent upon general-purpose processors and memory resources that can do any number of application inspection and processing tasks without the fixed function limitations created by ASICs. The CPU industry, and Moore's Law in particular, play a pivotal role in influencing the evolution of the application networking innovation cycle.

Traditionally, CPU processing performance was improved by increasing the clock speed, but increasing clock speed came with steep power consumption penalties. In 2005, in order to overcome these power constraints, the semiconductor industry introduced multicore processors with two or more central processing units (CPUs) on a single silicon chip to scale performance, reduce power and space, and reduce overall costs. But, multiple processors do not necessarily translate into a linear increase in computing performance. In fact, performance scaling in a multicore system is entirely dependent upon the software architecture.

At the system level, doubling the number of processors will not automatically double the performance without dramatic changes in system software design, and will therefore result in suboptimal utilization of expensive multicore CPUs. When designing a multicore system, the most important issue to consider is that there is a CPU speed mismatch with other slower non-CPU components such as memory and input/output (I/O) subsystems, so there is a performance penalty for memory and I/O operations. Additionally, multicore systems will not maximize multicore CPU utilization if software architecture doesn't evolve to support parallel computing principles. The good news is that these constraints can be addressed and multicore systems performance can be maximized with the right combination of scalable software architecture and optimal memory design.

Multicore System Design Challenges

Multicore processors pose new opportunities to scale up performance with parallel processing, but also new system design challenges for application networking system designs. As such, software and system architects will face the following design challenges:

1. **Memory:** The bus speed of memory access has always been lower than the CPU and this gap will impact performance if memory accesses are not optimized, since the CPU must sit idle while waiting for memory access. In a multicore system, increased memory access among multiple processors can create access bottlenecks and data duplications, thereby impacting performance.

2. **Parallelism**: Parallel processing allows a system to increase overall performance by executing multiple tasks simultaneously on multiple processors. But traditional software architectures that were developed for single CPU/single core CPUs were not designed to take advantage of multiple processors. Multicore CPU system software must be re-architected to optimally manage concurrency and reduce the overhead involved in managing shared resources. Without such a re-architecture of the software design, the benefits of parallel processing with multicore CPUs cannot be realized.
3. **Inter-Process Communication (IPC)**: Parallel processing requires communication between the processors to exchange system and state data so that all cores can simultaneously process all tasks without sub-optimally limiting tasks to certain cores. In a multicore system with parallel processing, the overhead associated with the replicating memory across cores increases exponentially with the number of cores and degrades overall performance.

To continue to benefit from Moore's Law, high-performance system software must address the fundamental challenges associated with multicore architectures.

Parallel processing system designs are traditionally classified as symmetric multi-Processing (SMP) or massively parallel processing (MPP) systems. In SMP systems, multiple processors share operating system, memory and other I/O resources. The single operating system load-balances tasks equally among processors to improve performance, but the SMP systems cannot scale up performance due to significant locking overhead involved in managing shared resources. In contrast, MPP systems do not share any resources among processors. Each processor in MPP will own dedicated memory and operating system resources. Each processor will be assigned to do different specialized tasks to increase performance gains. However, MPP systems cannot scale up performance due to inefficient load sharing among multiple processors and significant overhead involved for inter-processor communication.

As multicore density is set to increase in the future, traditional SMP and MMP designs will not be able to optimize multicore capabilities given the inherent system limitations outlined above. To continue to benefit from Moore's Law, high-performance system software must address the fundamental challenges associated with multicore architectures. Software architectures that were developed and optimized for single core CPUs will need to be re-architected based on parallel computing principles to minimize locking and communication overhead.

Specifically, the system memory architecture should be designed to optimize memory access and increase overall system utilization. The parallel processing systems must share certain system state information across all CPUs in order to ensure that security and service-level policies are accurately enforced across the entire system. Traditional system designs typically employ one of the following conventional design options to share memory across parallel processing cores in a multicore system design:

1. **Distributed Memory**: In this type of system, each processor has dedicated memory and any information exchange between the processors happens over an internal network. Typically, such system designs need inter-process communication (IPC) mechanisms to exchange shared data. IPC architectures require data to be prepared and transferred from one processor to another, which results in significant overhead. Additionally, each processor owns its own memory; therefore, data will have to be duplicated, leading to inefficient memory usage and limited scalability.
2. **Shared Memory**: In this type of system design, all processors have access to the same memory and can directly access that memory regardless of physical location. Since any CPU has direct access to all memory without the need for IPC and duplication functions, there's no associated overhead or performance impact. In addition, shared memory eliminates memory duplication to improve overall system performance. However, traditional shared memory systems are impacted by the fact that simultaneous access to the same memory blocks creates contention between the CPU cores that require locking access to a single core at any given time. As a result of this contention, traditional shared memory systems suffer from locking overhead, which also multiplies with increasingly dense multicore systems.

Most application networking systems use per-core dedicated memory and retrofit legacy software onto multicore processors, only moderately improving performance. But these techniques yield suboptimal results and cannot scale due to traditional IPC communication overhead, inefficient memory usage and data duplication. Furthermore, since dedicated memory systems have no shared visibility and control of system data, inefficient policy enforcement can result from synchronization gaps in the configuration memory (for example, dynamic enforcement of rate limiting and security policies).

Modern application networking systems must be able to handle increasingly high rates of application-layer traffic, which can only be implemented cost-effectively by implementing a multicore software system design that efficiently manages system memory. It is of utmost importance to keep the entire system state consistent, coordinated and easily accessible to all parallel processing resources. The consistent multicore system state ensures that configuration and security policies are accurately enforced. This benefits features such as caching and content acceleration that require rapid system learning and sharing of information.

ACOS High Speed Shared Memory Architecture

A10's Advanced Core Operating System (ACOS) was built from the ground up to optimize performance scalability of multicore CPU architectures. The ACOS architecture eliminates the drawbacks and overhead limitations of both traditional memory designs referenced above, which enables it to take advantage of the industry-wide technology trend towards multicore CPU system designs.

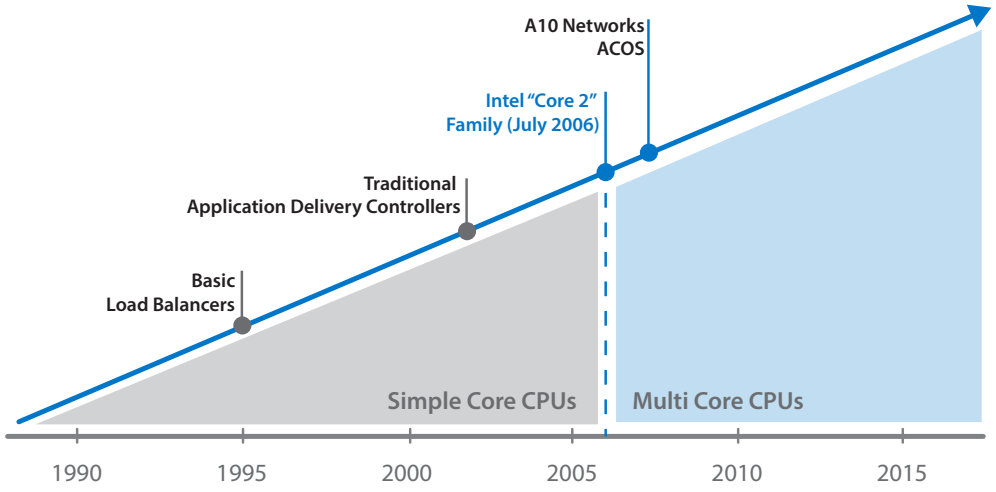

Figure 1: Multicore technology era

A10 Networks is the industry's first vendor to deliver a highly scalable 64-bit application networking platform optimized for multicore CPU system designs.

A10 Networks is the industry's first vendor to deliver a highly scalable 64-bit application networking platform optimized for multicore CPU system designs. The ACOS platform is based on innovative scalable symmetric multiprocessing (SSMP), which leverages supercomputing techniques for parallel processing to maximize the utility of multicore parallel processing architectures without the overhead of traditional designs.

SSMP in combination with A10's innovative "High-Speed Shared Memory Architecture" and "Flexible Traffic Accelerator" eliminates traditional design bottlenecks and linearly scales performance with increasing multicore CPU architectures. (Figure 2 shows key components for the ACOS architecture.)

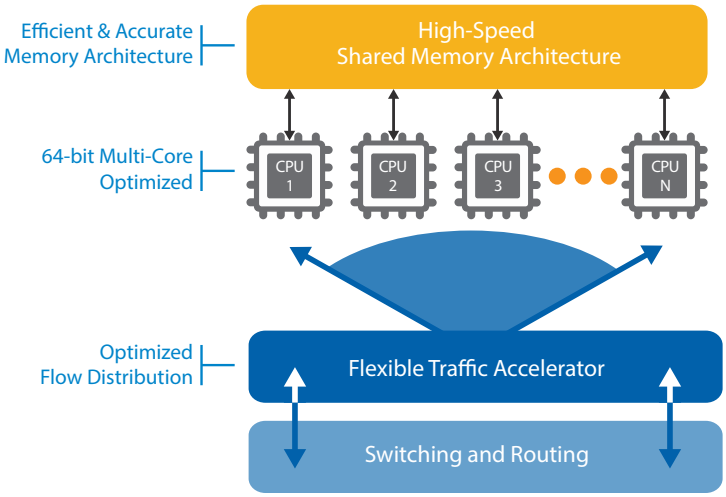

Figure 2: A10's ACOS architecture

ACOS High-Speed Shared Memory Architecture (SMA): Shared memory is available to all cores, which allows efficient and consistent access to shared data (configuration, policies, content caching and security learning). Consistent view of memory allows immediate enforcement of any policy or configuration changes without the need to copy memory blocks across CPU cores. In addition, real-time accounting and statistics will be accurately collected. Shared memory is also very efficient, since it eliminates the overhead and system costs associated with data duplication or replication. A10's High-Speed SMA allows all processors to share system state simultaneously, while also securing a portion of memory that does not require sharing to minimize locking overhead.

ACOS Flexible Traffic Accelerator (FTA): FTA is A10's high-performance intelligent network I/O technology that can distribute application flows intelligently across processor cores on deterministic paths. The Flexible Traffic Accelerator coordinates traffic distribution and efficiently utilizes multicore processors. FTA functionality is implemented in software or on specialized hardware. When using specialized hardware, FTA also performs certain hardware-based security checks and can discard unsecure traffic before it can impact system performance.

Additionally, the ACOS platform architecture separates forwarding and management planes to maximize concurrency. For traffic forwarding functions, multiple dedicated CPUs are allocated for highly efficient parallel processing of traffic. For management functions, ACOS allocates one or more dedicated CPUs for management purposes. As a result, administrators can reach devices and perform supervisory and health checks even under peak traffic conditions. The separation of management functions onto a separate core enables management flexibility and improves overall system robustness.

ACOS High-Speed Shared Memory Architecture eliminates IPC communication overhead and requires no memory duplication, allowing multicore resources to execute the most important task of processing application traffic. (Figure 3 depicts ACOS shared memory benefits.)

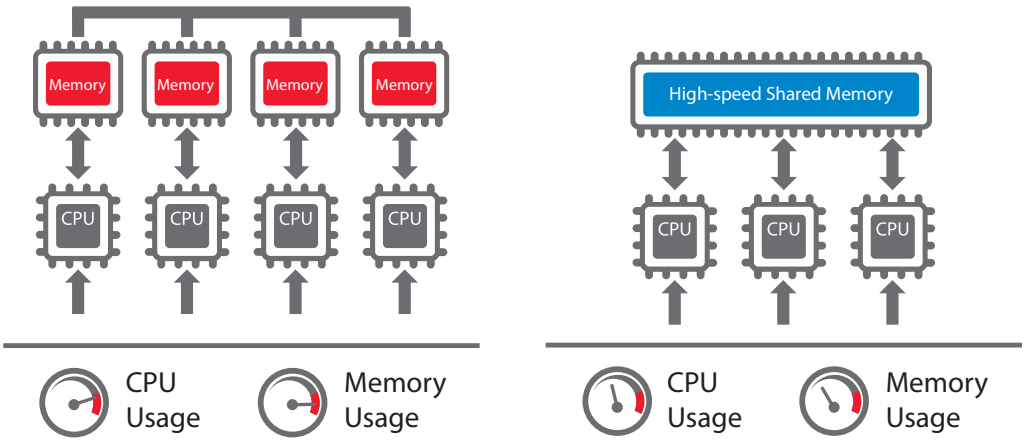

Figure 3: ACOS High-Speed SMA Benefits

ACOS Shared Memory Architecture optimizes the use of expensive multicore CPU and memory resources and requires fewer hardware components compared to conventional systems. Fewer components improve operational efficiency and overall reliability of the system.

ACOS Benefits

As a direct result of the ACOS High-Speed Shared Memory Architecture, A10 Networks Application Services Gateway (ASG) appliances require dramatically fewer resources (processors and memory) to achieve a comparable capacity relative to traditional designs with legacy software architecture. As a result, A10 Networks ASG appliances are dramatically more efficient, requiring significantly less power, cooling and rack space compared to traditional products (Figure 4 shows resource efficiency of different architectures). And seeing is believing, so to speak: A10 Networks' products can process two to five times more Web transactions in certain head-to-head product comparisons per unit of computing and memory resources, power or rack space.

As a direct result of the ACOS High-Speed Shared Memory Architecture, A10 Networks Application Services Gateway (ASG) appliances require dramatically fewer resources (processors and memory) to achieve a comparable capacity relative to traditional designs with legacy software architecture.

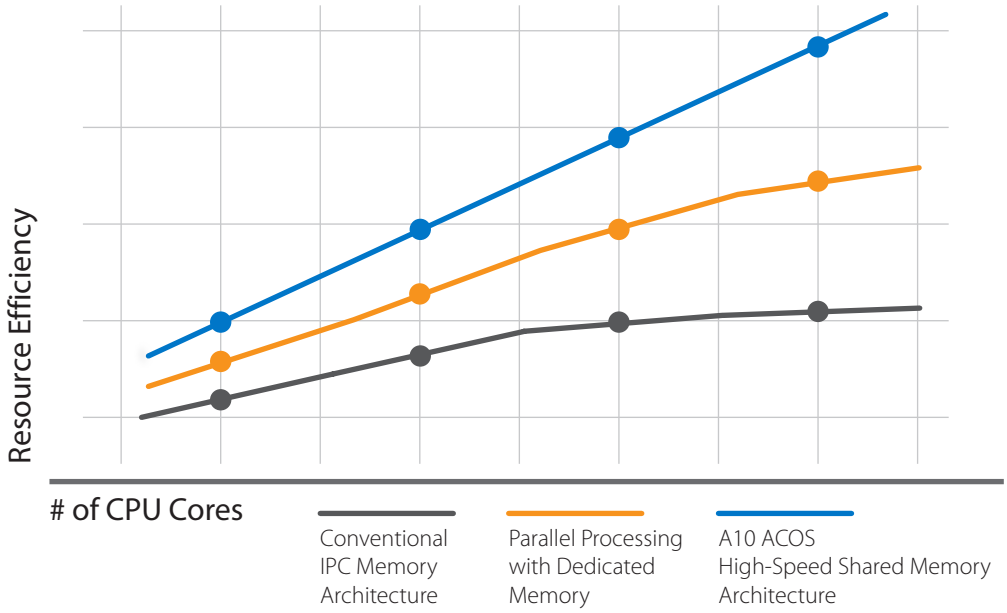

Figure 4: Linear scaling with ACOS High-Speed Shared Memory Architecture

What this really means to your business is better end user experience today and into the future; with superior ROI for your IT budget.

The ACOS platform delivers linear scale and provides many customer benefits:

- Improved Application Performance: ACOS significantly improves application availability, accelerates responses, increases connection speeds and improves end user experience.
- Predictable Performance: ACOS linearly scales performance and helps deliver appropriate service levels even with ever-increasing data center traffic growth.
- Reduced Complexity: High performance and scalability will help consolidation of legacy devices and reduce overall network complexity.
- Investment Protection: As network traffic grows, customers can scale up application services while protecting investing with existing infrastructure.
- Reduced CapEx: Higher performance and efficiency from similar hardware allows customers to realize more value from their capital investment
- Data Center Efficiencies and Better ROI Results:
  - Less power, cooling and space requirements will reduce costs.
  - ACOS platform can also offload resource-intensive functions such as SSL and compression from the server infrastructure, thereby reducing the load on servers and ultimately reducing the number of servers required.

Conclusion

A10 Networks' Advanced Core Operating System (ACOS) platform was built from the ground up based on supercomputing principles with High-Speed Shared Memory to meet rapidly growing application networking demands. ACOS linearly scales performance in increasingly dense multicore CPU systems and consumes dramatically fewer resources to achieve a comparable capacity relative to traditional designs. As a result, customers get a much more efficient system that can provide dramatically more application and security performance, and scale with much greater value. What this really means to your business is better end user experience today and into the future; with superior ROI for your IT budget.

About A10 Networks

A10 Networks is a leader in application networking, providing a range of high-performance application networking solutions that help organizations ensure that their data center applications and networks remain highly available, accelerated and secure. Founded in 2004, A10 Networks is based in San Jose, California, and serves customers globally with offices worldwide. For more information, visit: **www.a10networks.com**

Corporate Headquarters

A10 Networks, Inc
3 West Plumeria Ave.
San Jose, CA 95134 USA
Tel: +1 408 325-8668
Fax: +1 408 325-8666
www.a10networks.com

Part Number: A10-WP-21104-EN-01
Mar 2014

Worldwide Offices

North America
sales@a10networks.com
Europe
emea_sales@a10networks.com
South America
brazil@a10networks.com
Japan
jinfo@a10networks.com
China
china_sales@a10networks.com
Taiwan
taiwan@a10networks.com
Korea
korea@a10networks.com
Hong Kong
HongKong@a10networks.com
South Asia
SouthAsia@a10networks.com
Australia/New Zealand
anz_sales@a10networks.com

To learn more about the A10 Thunder Application Service Gateways and how it can enhance your business, contact A10 Networks at: **www.a10networks.com/contact** or call to talk to an A10 sales representative.

©2014 A10 Networks, Inc. All rights reserved. A10 Networks, the A10 Networks logo, A10 Thunder, Thunder, vThunder, aCloud, ACOS, and aGalaxy are trademarks or registered trademarks of A10 Networks, Inc. in the United States and in other countries. All other trademarks are property of their respective owners. A10 Networks assumes no responsibility for any inaccuracies in this document. A10 Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.