Xeon Grows Up

Insight

Gordon Haff
30 March 2010

You might argue that picking this moment in time to declare the Intel Xeon server processor "grown up" is to trumpet as news something that is already in the history books. In a sense, you would be right. We started talking about "The Xeon Age" over a decade ago.[1]

x86 processors such as Xeon moved far beyond their roots as departmental or small business servers during the Internet build-out of the late 1990s. They were enterprise-ready by any reasonable definition of the phrase. Even if you focus the enterprise-ready lens tightly on multiprocessor servers that scale vertically as single SMP systems, rather than horizontally in the manner of most network-oriented computing, x86 was making inroads there as well. I worked on some of those scale-up products well over ten years ago.[2]

That said, the arrival of "Nehalem-EX," the new high-end of Intel's x86 server processor family, is still a watershed event. It's not the first Xeon to scale, but it makes scaling up far easier for Intel's server OEMs. It also significantly ups the performance bar, even absent custom OEM silicon. Add to this a doubling down on reliability and availability features and Nehalem-EX starts to look like a product that really is different in kind, and not just in degree, from what preceded it.

Before delving into the Intel Xeon Processor 6500 and 7500 Series—Nehalem-EX's official name—and the path that brought Xeon here, let's consider why we might want a product like this in the first place.

Scaling Up and Scaling Within

Historically, the answer to "Why scale up?" was straightforward. You had an application that needed to run within a single operating system image that

1 See our *The Xeon Age*

2 tinyurl.com/yz5hh94

required more compute power than a single microprocessor could provide. This dictated a symmetrical multiprocessing (SMP) server in which more than one processor had direct access to a single shared pool of memory. While there's no canonical definition of how big something has to be to be "scale-up," a good definition is anything larger than the ubiquitous dual-socket systems that make up so much of server unit volumes.[3]

This remains a common reason to buy larger SMP systems. Many applications either don't run, or don't run well, in a distributed, cluster architecture. Microsoft Windows and its databases like SQL Server have matured along with their associated applications, so scaling vertically on x86 becomes a more practical option with each passing year. Linux has gone through a similar process. At the very least, IT shops embracing x86 for their scale-out environments—as so many have done—find that there's often no intrinsic need to use a different architecture and operating system for a good number of their back-end systems.

Such servers weren't historically as large as the biggest RISC or Itanium boxes, although the limits of x86 scale have steadily fallen away over time.[4] However, even ultimate scale-up aside, midrange (which is to say four- to eight-socket) x86 servers commonly run databases and associated back-office applications. Such servers are enormously powerful by any historical standard.

Transaction processing, such as taking orders, remains an important scale-up workload. Indeed, many types of transaction volumes have increased enormously because of things like automated toll collection and tracking systems. However, more recently, business analytics has emerged as an even more voracious consumer of CPU cycles. Analytics was once something applied occasionally to copies of production data. Today, it's increasingly about using models, including neural networks, which look for patterns in a large volume of real-time and historical data. And using the results to make changes on-the-fly rather than days or weeks from now. These types of tasks take lots of cores, lots of threads, and lots of memory.[5]

However, today there's another big reason that people buy larger systems: server virtualization. Virtualization lets a single physical server be carved up into many logical servers, which can then be resized, archived, or moved to a different box. Although it might seem counterintuitive to purchase a larger server only to subdivide it, there are a number of reasons why this can make sense.

- While virtual machine image management is arguably the bigger challenge—many vendors have started using the term "virtual sprawl" as the hook to selling their management products—virtualization hasn't made management of the underlying physical server go away. Fewer boxes equals less management.
- Larger systems provide greater flexibility in allocating resources to virtual machines (VMs). A larger server means a larger pool of hardware with which to work. This starts to become particularly important as individual VMs increasingly encapsulate larger and more resource-intensive applications, such as database engines.
- Virtualization can also be used to simplify the delivery of composite applications. Oracle's midmarket Siebel CRM solution is a good example; the database runs in one VM, the Siebel application and Web server in another, on a single physical server.[6] But running all the pieces of an enterprise application on one box may require more than just a two-socket box.
- Larger servers don't just have more processors. They also support larger memory pools and stronger, higher-bandwidth I/O. They can also more economically amortize predictive failure

[3] Though in a multi-core world, the four, eight, or more cores of "just a two-socket system" look pretty "scaled up" to the operating systems that run them.

[4] IBM, NEC, and Unisys have been the most active in the Big Iron x86 space during recent years with designs that grow beyond what's possible with off-the-shelf chipsets. See our NEC Goes Big on Xeon and IBM's Uniquely Scalable Xeons.

[5] IBM talks about large composite applications with both transactional, analytic, and distributed sensor components under the "Smarter Planet" umbrella. See our *Horses for Courses: Optimizing for Workloads*.

[6] See our Hands On: Sun Puts Siebel CRM in a Box.

and other reliability and availability mechanisms, and therefore tend to have more features of this type.

In short, while big applications by themselves remain a motivation to scale-up, virtualization ("scale within") has emerged as another key reason —especially as "virtualization" and "big application" increasingly aren't mutually exclusive.

The Road to the Xeon 7500

The significance of the Xeon 7500[7] reflects, in part, Xeon's evolution, and, in part, the maturity of its ecosystem of operating systems, middleware such as application servers and databases, and the other software that runs on it.

Xeon—Intel's brand for "server level" x86 processors—was originally a product of the distributed computing world. This was distributed computing in the sense of small businesses, workgroups, or departments, rather than the huge clusters of relatively small servers commonplace in datacenters today.[8] As such, Xeon's design center was quite different from the large systems based on RISC or other types of processors that usually ran Unix.

Xeon grew up over time, and even found its way into "Big Iron" servers, although this was a secondary area of concern; Intel was understandably more interested in the far larger pot of dollars associated with the volume market. For example, Xeon chipsets and motherboards provided few features to let server vendors straightforwardly extend Intel's designs to add more processors. While Xeon steadily improved areas like low-level error detection, it didn't immediately tackle proactive failure handling and error recovery of the sort that vendors were increasingly adding to processors more focused on "mission-critical" environments or "fault tolerant" systems.[9]

When Intel introduced its 64-bit Itanium processor, that was the family that debuted Machine Check Architecture (MCA), which defined processor, chipset, firmware, and operating system responsibilities for advanced error handling. Over time, various reliability features migrated downward to Xeon. But Intel continued to position Itanium, even after Xeon was extended to 64-bits (and thereby able to support much larger memory complements),[10] as the processor of choice where robustness and scalability were paramount.

With the Xeon 7500, that split positioning effectively ends. Some of this is marketing. Intel is simply no longer saying that one processor family is *inherently better* than the other. The distinction is now mostly by operating environment. Itanium focuses on HP-UX, NonStop, and OpenVMS; Xeon focuses on Windows, Linux, and Solaris.

However, this shift in positioning is rooted in a significant technical evolution. With the Xeon 7500, Xeon now sports MCA as well.[11] Memory and I/O handling are also accelerated thanks to QuickPath Interconnect (QPI), Intel's first serial interconnect in a quad-socket processor design. This dramatically boosts scalability. The Xeon 7500 also re-enables[12] eight-socket system designs without the need for OEMs to design a custom chip set. For those who do want to go even bigger, the Xeon 7500 and QPI make explicit the ability to connect into OEM-designed node controllers.

---

7 Strictly speaking, the Xeon 7500 refers only to the processor series that scales to four-sockets and beyond. A less-expensive Xeon 6500 Series is two-socket only, but, unlike the Xeon EP line, otherwise shares the memory capacity and bandwidth of the Xeon 7500. However, we use Xeon 7500 throughout as a generic term for all Nehalem-EX processors.

8 And the norm for Internet-scale sites like those at Google and Microsoft.

9 I use the term "mission-critical" advisedly even though the reality is that many parts of a computing infrastructure can be critical to the proper functioning of an enterprise; it's not just the big, back-end systems that are important (although they may well be *individually* more important).

10 See our 64-Bit Goes Mainstream: Who Wins, Who Loses.

11 Although Itanium's current generation ("Tukwila"), still has more complete MCA coverage than Xeon.

12 In 1997, Intel bought Corollary in order to bring to market an eight-processor motherboard. However, the product wasn't especially successful and Intel didn't follow-up with a refresh.

Also consider the Xeon 7500 in the context of the software and systems associated with it. Windows and Linux continue to mature in reliability, security, and the level of workloads that they can handle. So do the middleware and applications that run on them. x86 virtualization, aided in part by processor features that provide explicit assists, can now comfortably run even so-called "production workloads" such as back-end databases. That wasn't true ten—or even five—years ago. In short, both the Xeon 7500 and its associated ecosystem are today fully capable of running a wide range of workloads that were historically more associated with mainframes or RISC/Unix systems.

A New Architecture

The Nehalem codename refers, overall, to Intel's current desktop and server x86 microarchitecture. Debuting on servers in early 2009, it built on the 45nm process technology earlier introduced with the Core microarchitecture. To that, it added pretty much all of the major architectural features of modern microprocessors, providing not only a big leap forward in performance, but also a foundation largely free of legacy encumbrances such as the front side bus (FSB).

The Xeon 7500 builds on this foundation. Its integrated memory controllers, serial interconnects between processors, and simultaneous-multi-threading (SMT) are all new features relative to its quad-socket predecessor. Many of this generation's features are even more significant in larger-scale systems. High bandwidth, low latency, large caches, and accomplished reliability, availability, and serviceability (RAS) become even more important as systems get bigger, whether in processor count, memory capacity, or both. This is evidenced by Intel's performance estimates for the Xeon 7500, which it describes as the "biggest performance jump ever in Xeon history."

The details of a processor's design interrelate in various ways. Nonetheless, we can think about the Xeon 7500 as most focused on tackling three broad areas: scalable performance, virtualization, and reliability.

Performance

Performance is the yardstick by which processors have been traditionally measured. It's not the *only* important aspect of their design, but it certainly is a big part. Simply on a per-processor basis, the Xeon 7500 delivers in that regard. Intel estimates that, compared to its quad-socket predecessor, the Xeon 7500 will have up to 3 times the database performance, more than 2.5 times the integer throughput, and over 3.8 times the floating-point throughput. Early benchmark data from OEMs Fujitsu and NEC show between a 1.54x and 1.75x scaling ratio in moving from quad-socket to eight-socket configurations, including I/O-intensive workloads like business analytics.

These gains derive in part from a wide range of low-level microarchitectural enhancements. However, the Xeon 7500 also applies its 2.3 billion transistors to implement several top-level features that directly relate to performance.

Its eight cores per chip is double the count of recent predecessors.[13] Hyper-Threading, Intel's version of simultaneous-multi-threading, doubles again the number of threads the processor can handle simultaneously, to 16.[14] 24MB of shared cache keeps a large pool of data close to the cores working on it. That data can also be brought in from memory more quickly than in previous generations, because the Xeon 7500 integrates its memory controllers onto the same die as the processing cores.

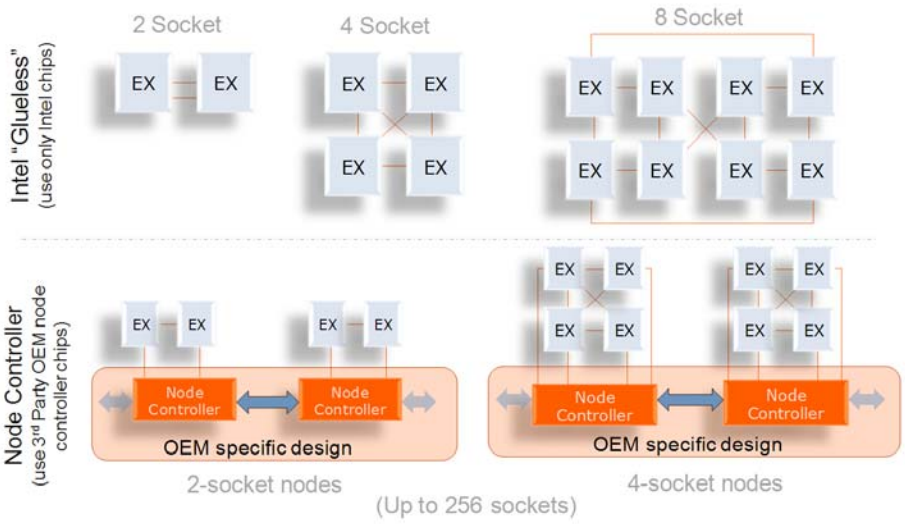
Yet, for all that, what's most significant about the Xeon 7500 isn't the speediness of an individual chip, but rather how scalable systems can be constructed using that chip as a foundation.

One important aspect of this is that the Xeon 7500 connects processors using the QuickPath

[13] Though "Dunnington" (the Xeon 7400) had six cores.

[14] See our Gradations of Threading. Unlike adding cores, Hyper-Threading doesn't actually add more execution units. However, it helps a processor avoid idling the execution units it does have while waiting for data to arrive from memory.

Interconnect (QPI), a point-to-point link, rather than the parallel FSB used in its predecessors. QPI avoids much of the contention for resources that can limit scalability when a bus connects processors with each other and with memory. Intel introduced QPI with its dual-socket Xeon 5500 but, with the Xeon 7500, we see both the flexibility and the scaling possible using this sort of approach.



A two-socket Xeon 7500 system supports 32 DIMM slots, 512GB of memory, almost twice the complement of the Xeon 5500—handy for workloads that need a lot of memory, such as virtualization. A two-socket-only version, the Xeon 6500, isn't upgradeable but supports the same memory capacity and bandwidth per socket. Alternatively, by fully populating a Xeon 7500 node with four processors, memory capacity increases, in concert, to 1TB.

QPI also enables an 8-socket "glueless" design. This means that system OEMs can now go all the way to eight sockets without having to design their own custom chipsets. However, for those OEMs who do want to go larger, the Xeon 7500 architecture also explicitly incorporates the notion of a node controller that connects together dual-socket or quad-socket "nodes." While OEMs have scaled prior Intel Xeons to high processor counts, it was always relatively difficult to do so given that the architecture wasn't really designed with that in mind. The Xeon 7500 is therefore ready to support up to 8-socket (64-core) designs out of the box, and is designed to support up to 256-socket servers with OEM extensions. Those are enormously Big Iron configurations—larger, in fact, that almost all RISC and Itanium servers.

Nor are node controllers limited to functioning as a way to connect more processors. For example, IBM's MAX5 option on its new eX5 server lineup is actually a node controller for the Xeon 7500 that doubles as a way to further boost system memory capacity using a "scalable memory buffer."[15] OEMs can also add their own reliability features, whether in software or hardware, such as memory mirroring.

Flexible Virtualization

Per-chip performance and performance running large workloads such as databases is the traditional way of thinking about system performance. However, performance in a virtualized environment, including running back-end production applications, is increasingly important and will only become more so. Leading-edge datacenters are moving from the mindset in which virtualization is something you did here and there to a mindset in which running everything virtualized is the norm.

One aspect of this shift is processor features that explicitly support virtualization. The first such delivery under the Intel Virtualization Technology (VT) umbrella eliminated the need for the virtual machine monitor (VMM)—i.e. the hypervisor—to listen for, trap, and handle certain instructions from the guest operating systems.[16] More recently, VT-d

[15] See our *IBM System x: From Boxes to Workloads*.

[16] The essential problem was that guest operating systems couldn't be allowed to directly execute "sensitive" instructions that could change the system state in a way that could affect other guests. VT adds virtualization-specific instructions that make it easier to switch modes and otherwise deal with the ways

(Virtualization Technology for Directed I/O) reduces the VMM's involvement with I/O traffic by letting the VMM assign specific devices to a given guest OS.

Not all VT features optimize at the single-system level. Technologies like Microsoft's Live Migration and VMware's VMotion can move a running guest OS from one physical server to another. This is a hugely popular feature because it allows for scheduled maintenance without shutting down applications. It's also at the heart of the more flexible and dynamic datacenter operations that are the goal of the current wave of computing. VT FlexMigration helps hypervisors establish a consistent set of instructions across the servers within a migration pool. This prevents, for example, a guest operating system from trying to execute a newer-generation Streaming SIMD Extension (SSE) instruction on a system where it isn't supported. The result is more flexibility in configuring a virtual infrastructure.

Memory is also key to virtualization performance. The Extended Page Tables[17] in the Xeon 7500 can significantly improve memory performance under virtualization, especially under the heavy loads associated with enterprise applications. The page tables in a processor store the mapping between physical memory addresses and virtual ones; they're a fundamental aspect of a virtual memory system. Virtualization requires an extra level of indirection. The guest OS virtual memory now maps to what the VMM presents as "physical" memory but which, in fact, is a software abstraction. This "physical" memory then maps to the machine memory, that is, the actual memory hardware. This double-mapping can be handled in software using Shadow Page Tables. However, it's much more efficient to handle it in hardware, as Extended Page Tables do.

The amount of memory also matters. Virtualization tends to improve the utilization of a system's processors—sometimes dramatically. Memory requirements tend to grow apace, both quantity and the bandwidth needed to support the larger memory complements. The Xeon 7500 uses a "scalable memory buffer" on each of four links out of the processor. Each of these buffers has two DDR3 memory channels with two DIMMs per channel. This provides for a hefty complement of 1TB of memory on a 4-socket server using 16GB DIMMs.

Advanced Reliability

As a type of system runs larger and more important workloads, the more important it is that those workloads be protected from failures. Even where workloads are not individually larger, server virtualization tends to "put more eggs in every basket." Much reliability is the responsibility of software, of course; the operating system, virtualization software, and the applications themselves bear a lot of the responsibility for maintaining uptime. However, the underlying processor and server hardware have their roles to play as well.

In all, Intel counts some 20 new reliability, availability, and serviceability (RAS) features in the Xeon 7500. Some are a feature of silicon, and don't need additional pieces to work. For example, the QuickPath Interconnect can retry packets and failover to another clock on its own. Many of these reliability features relate to memory. The memory interconnect can switch to another lane or another clock. Memory can be mirrored within a socket or to another socket, or a DIMM channel can be allocated as a spare.

Others require operating system and/or server hardware support, such as the hot addition of physical I/O hubs or dynamically changing memory capacity. In this regard, RAS for the Xeon 7500 starts to look a lot more like how the designers of RISC/Unix, Itanium, and mainframe systems think about RAS. In those and other systems with relatively integrated hardware and software stacks, silicon RAS features are often designed to explicitly work in concert with other components. For example, the silicon may detect

---

that running virtualized is different from running on bare metal.

[17] AMD calls this feature Nested Page Tables.

and even isolate an error, but recovering from that error may be better handled at the system software layer.

Machine Check Architecture (MCA) Recovery provides a good example of cooperation between the hardware and the software. MCA Recovery is a mechanism in which the silicon works with the operating system to allow a server to recover from uncorrectable memory errors which would have caused a system crash in prior generations. This type of capability has been available on RISC, mainframe, and Itanium systems for some time, but this is the first time it has been implemented in a Xeon-based system.[18]

Many detected errors can be corrected by hardware error correction mechanisms like ECC. In that case, the Corrected Machine Check Interrupt feature notifies the OS of the location of these errors so they can be tracked for preventative maintenance. However, up until now, when multi-bit errors that could not be corrected in a Xeon-based system were detected, the system had no alternative but to shut down.

In this first implementation, MCA Recovery allows the OS to recover when uncorrectable errors are discovered during either an explicit write-back operation from cache, or by a memory patrol scrub which examines every server memory location daily. When an uncorrectable error is detected, the silicon interrupts the OS and passes it the address of the memory error. The OS then determines how best to recover from the error and continue operation of the system. The OS marks the defective memory location so it will not be used again, resets the error condition, and the system can keep running in many cases.

The concept behind MCA Recovery is not new. What's new is that it's now been brought to the volume server space.

Conclusion

Change often happens incrementally; the accumulation of capabilities over time is more significant than any one leap forward. Xeon's advances over the years are no exception. Each new generation brought with it more performance, an up-leveling of RAS, and more features relevant to servers operating in a datacenter. Think of this as evolutionary gradualism applied to processors.

However, some generations are little hops, while others are leaps. Collectively, Nehalem represents what's arguably the biggest leap in Xeon's history. In evolution, it's called punctuated equilibrium, a model for discontinuous tempos of change. This was true of the dual-socket Xeon 5500. But an even stronger case can be made for the Xeon 7500 given that Nehalem features such as QPI and integrated memory controllers bring even more benefit to big systems than to smaller ones. What's more, the Xeon 7500 debuts many reliability and other features that we hadn't seen in Xeon before.

In short, Xeon 7500 is a chip for serious workloads. It can't be shrugged off as a mere "volume chip." It is that, of course. But it also has the scalability, the support for virtualization, and the RAS to go toe-to-toe with any other processor for the job of running serious workloads. Xeon has grown up and become High Volume Big Iron, in many respects a new class of server.

[18] Itanium's MCA capabilities still exceed what are available in Xeon, but this generation's RAS is a big step forward nonetheless.