



Hyper-Threading Technology on the Intel® Xeon™ Processor Family for Servers

Offering increased server performance
through on-processor thread-level parallelism

Contents

Introduction	3
Overview of Hyper-Threading Technology	4
Resource Utilization	5
Microarchitectural Details	6
How Server Platforms Use Hyper-Threading Technology	8
How Software Takes Advantage of Hyper-Threading Technology	9
Business Benefits of Hyper-Threading Technology	10
Summary	10

Introduction

Hyper-Threading Technology is a groundbreaking innovation from Intel that enables multithreaded server software applications to execute threads in parallel within each processor in a server platform. The Intel® Xeon™ processor family uses Hyper-Threading Technology, along with the Intel® NetBurst™ microarchitecture, to increase compute power and throughput for today's Internet, e-Business, and enterprise server applications. This level of threading technology has never been seen before in a general-purpose microprocessor. Hyper-Threading Technology helps increase transaction rates, reduces end-user response times, and enhances business productivity providing a competitive edge to e-Businesses and the enterprise. The Intel Xeon processor family for servers represents a significant leap forward in processor design and performance by being the first Intel®-based processor to support thread-level parallelism on a single processor.

Intel's commitment to providing businesses with world-class performance is long-standing. Each new processor introduction and microarchitecture shift has resulted in significant performance advances. The Intel Xeon processor family for servers has changed the landscape of processor design and performance by implementing support for multiple software threads on each processor core. Hyper-Threading Technology provides thread-level-parallelism (TLP) on each processor resulting in increased utilization of processor execution resources.

With processor and application parallelism becoming more prevalent, today's server platforms are increasingly turning to threading as a way of increasing overall system performance. Server applications have been threaded (split into multiple streams of instructions) to take advantage of multiple processors. Multithreaded applications are designed to be separated into distinct execution streams, or threads. These applications make the best use of multiple processors because they are designed with their own execution and performance in mind rather than relying on operating systems or compilers. For example, most Java Virtual Machine* implementations are multithreaded, as many of the

administrative tasks are separate threads. Multi-processing-aware operating systems can schedule these threads for processing in parallel, across multiple processors within the server system. These same applications can run unmodified on the Intel Xeon processor family for servers and take advantage of thread-level-parallelism on each processor in the system. Hyper-Threading Technology complements traditional multi-processing by offering greater parallelism and performance headroom for threaded software.

Dual-processing server applications in the areas of Web serving, search engines, security, streaming media, departmental or small business databases, and e-mail/file/print can realize benefits from Hyper-Threading Technology using Intel Xeon processor-based servers. Multi-processing server applications in the areas of customer relationship management, media servers, site management, enterprise databases, business intelligence, collaboration/e-mail and supply chain management can also reap tangible benefits from Hyper-Threading Technology using the Intel Xeon processor MP in 4-way, 8-way and above servers. All of these types of solutions, and more, can take advantage of the faster response times and higher transaction rates offered by Hyper-Threading Technology. As software applications continue to be optimized to take greater advantage of processor parallelism, Hyper-Threading Technology provides an additional boost for newer capabilities and the growing needs of today's businesses.

This paper explains how Hyper-Threading Technology works within the processor, and describes how server platforms, operating systems and server applications take advantage of it. While this focuses exclusively on servers, Hyper-Threading Technology is not restricted to servers. Intel has also introduced Hyper-Threading Technology into workstations, business desktops and consumer desktops. For a discussion of how Hyper-Threading Technology is implemented on workstations, and the benefits realized, please refer to the "Intel's Hyper-Threading Technology: Delivering Performance Boosts for Workstations" document.

Overview of Hyper-Threading Technology

Hyper-Threading Technology is a form of simultaneous multithreading technology (SMT), where multiple threads of software applications can be run simultaneously on one processor. This is achieved by duplicating the architectural state on each processor, while sharing one set of processor execution resources. The architectural state tracks the flow of a program or thread, and the execution resources are the units on the processor that do the work: add, multiply, load, etc.

Let’s use a manufacturing plant as an analogy of a processor. In a traditional manufacturing plant there are multiple assembly lines (execution resources) and one supervisor (architectural state) that keeps track of orders (threads) and keeps them moving through the plant. Our supervisor can keep the assembly lines within the plant working in parallel, but if he/she has to wait for a part, an assembly line may slow down while waiting for the part as well. In the case of a processor, an execution unit may have to wait for data from disk.

With Hyper-Threading Technology, each manufacturing plant is enabled to have two supervisors (or two architectural states). They share the same common set of assembly lines within the plant. When one of the supervisors is waiting for a part to be delivered, the other supervisor can make use of the

assembly lines – resulting in a more efficient, higher performing manufacturing plant. Now the plant can turn out more goods per hour since the assembly lines are running and being utilized more often. There was no need to add assembly line resources; we’re just using them more effectively.

In Figure 1, the left-hand configuration represents a traditional multi-processor (MP)-based system with two discrete physical processors. Each processor has its own set of processor execution resources and its own single architectural state. The right-hand configuration in Figure 1 represents an Intel Xeon processor family-based multi-processor system where each processor features Hyper-Threading Technology. As you can see, the architectural state for each processor is duplicated, but each still has one set of execution resources.

When scheduling threads, the operating system treats the two separate architectural states as two separate “logical” processors. Multi-processor-capable software applications can run unmodified with twice as many logical processors to use. Each logical processor can respond to interrupts independently. The first logical processor can track one software thread, while the second logical processor can track another software thread simultaneously. Because the two threads share one set of execution resources, the

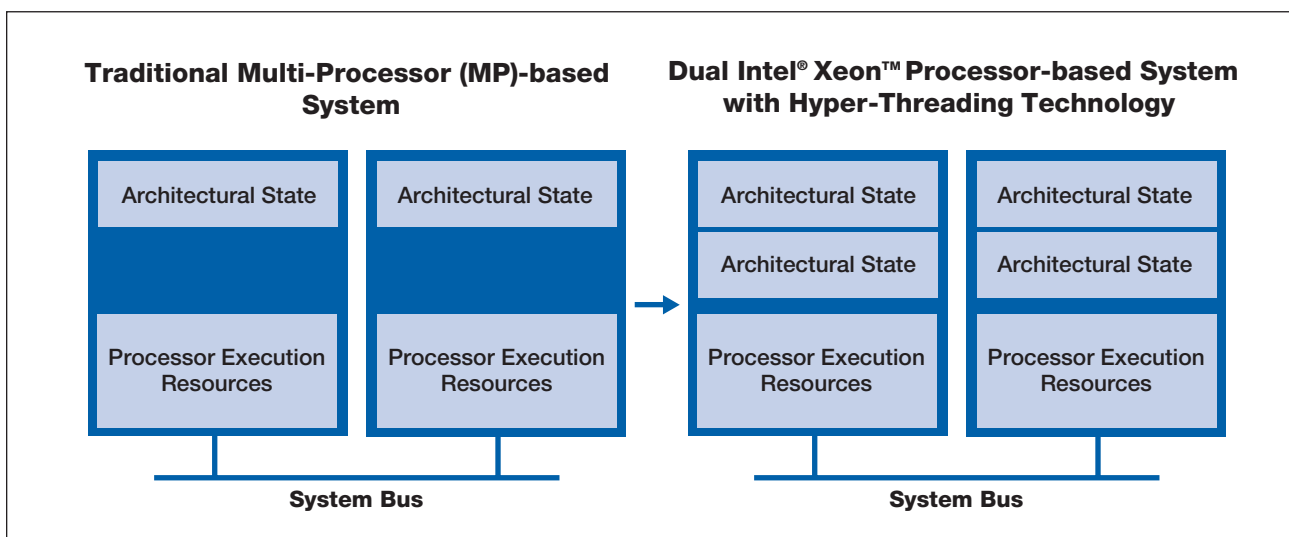


Figure 1: A multi-processor (MP)-based system with Hyper-Threading Technology duplicates the architectural state on each physical processor, providing two “logical” processors per physical processor.

second thread can use resources that would be otherwise idle if only one thread was executing. The result is an increased utilization of the execution resources within each physical processor package.

This improvement in CPU resource utilization yields higher processing throughput for multithreaded applications. For example, one logical processor can execute a floating-point operation while the other logical processor executes an addition and a load operation. Hyper-Threading Technology is complementary to MP-based systems because the operating system can not only schedule separate threads to execute on each physical processor simultaneously, but on each logical processor simultaneously as well. This improves overall performance and system response because many parallel threads can be dispatched sooner due to twice as many logical processors being available

to the system. Even though there are twice as many logical processors available, they are still sharing one set of execution resources. So the performance benefit of another physical processor with its own set of dedicated execution resources will typically offer greater performance levels. In other words, Hyper-Threading Technology is complementary to multi-processing by offering greater parallelism within each processor in the system, but is not a replacement for dual or multi-processing.

Resource Utilization

To understand how Hyper-Threading Technology improves resource utilization, let's look at an MP-based system containing two traditional superscalar processors that can each execute three instructions every cycle and is running two separate threads. As shown in Figure 2A, each box within each processor represents an execution unit. Each

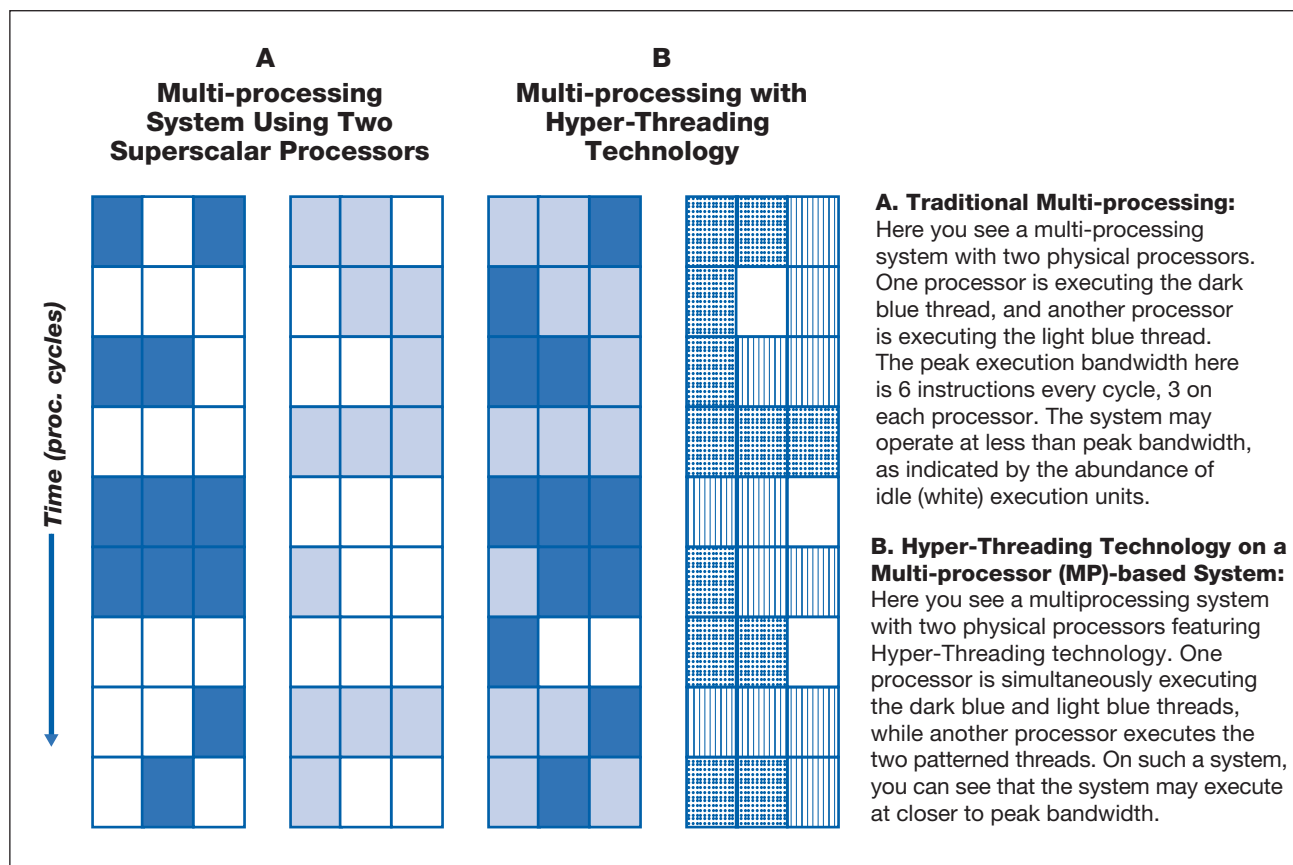


Figure 2: On a traditional multi-processor (MP)-based system, the idle (white) execution units indicate execution resource utilization is not optimum, while Hyper-Threading Technology not only allows multiple threads to execute simultaneously, but also reduces the number of idle execution units. This figure is a demonstration of the concepts underlying Hyper-Threading Technology and does not attempt to show the resource utilization patterns of any existing processor or software application.

set of three horizontal boxes within each processor shows how the three execution units are being used during a given clock cycle. If a box is white, that execution unit is idle during the given clock cycle (e.g., if one box is white during a clock cycle, then one-third of the resources are idle and two-thirds of the resources are utilized during that clock cycle). The vertical sets indicate the utilization of the three execution units over time (during multiple clock cycles). The more white boxes, the less efficient the utilization of execution resources. Sometimes all the execution resources within a processor are being used, and sometimes there are idle resources that are waiting due to things like cache misses, branch mis-predictions or instruction dependencies for each thread.

Figure 2B shows the result of Hyper-Threading Technology with the two processors now running two separate threads simultaneously within each processor. It is easy to see that we are now using the execution units more often and reducing idle time on each processor. Resource utilization within each processor can increase significantly, but it is dependent on how each application is designed to use threading and the resources within the processor. If an application is multithreaded in such a way that different threads can use different resources in parallel (such as scheduling integer computation-intensive work on one logical processor while running floating-point computations on the other), then throughput can increase significantly within each processor and for the entire MP-based system.

Microarchitectural Details

Figure 3 shows a high-level block diagram of the Intel Xeon processor family for servers.

With Hyper-Threading Technology, the execution resources on the Intel Xeon processor family are shared by two architectural states, or two logical processors. The majority of the execution resources are in the Rapid Execution Engine and Integrated Cache Subsystem, which process instructions from both threads simultaneously. The Fetch and Deliver engine and Reorder and Retire block partition some of the resources to alternate between the two threads.

Fetch and Deliver Engine

The Fetch and Deliver engine alternates between fetching instructions from one logical processor and the other, and sends these instructions to the Rapid Execution Engine for processing. At the Level 1 Execution Trace Cache, one line is fetched for one logical processor, and then one line is fetched for the other logical processor. This continues, alternating back and forth, as long as both logical processors need to use the Execution Trace Cache.

If one logical processor is not requesting use of the cache, then the other logical processor can use the full bandwidth of the Execution Trace Cache.

Rapid Execution Engine

At the Rapid Execution Engine, both logical processors execute simultaneously. The Rapid Execution Engine takes instructions from the instruction queues and sends them to the execution units as fast as it can. The instructions are selected based only on dependencies and availability of execution units. The instructions may be selected out-of-order, meaning that later instructions that are independent can be scheduled before earlier instructions. The execution core is for the most part oblivious to what instructions belong to which logical processor, and the schedulers do not distinguish between instructions of different logical processors. The schedulers simply map independent instructions in the instruction queues to available execution resources. For example, a scheduler may map two instructions from thread #1, and one instruction from thread #2 in the same cycle.

Integrated Cache Subsystem

The Integrated Cache Subsystem delivers data and instructions to the processor core at a high speed with larger cache lines than previous-generation processors. Because the Integrated Cache Subsystem is clocked at the same rate as the processor core, as faster processors are released, the cache speed can increase correspondingly providing high-speed access to key data. The larger cache line sizes also decrease average cache misses. A large (up to 2MB), Integrated Three-Level cache is available only on the Intel Xeon processor MP for 4-way and above server platforms. This additional cache space is

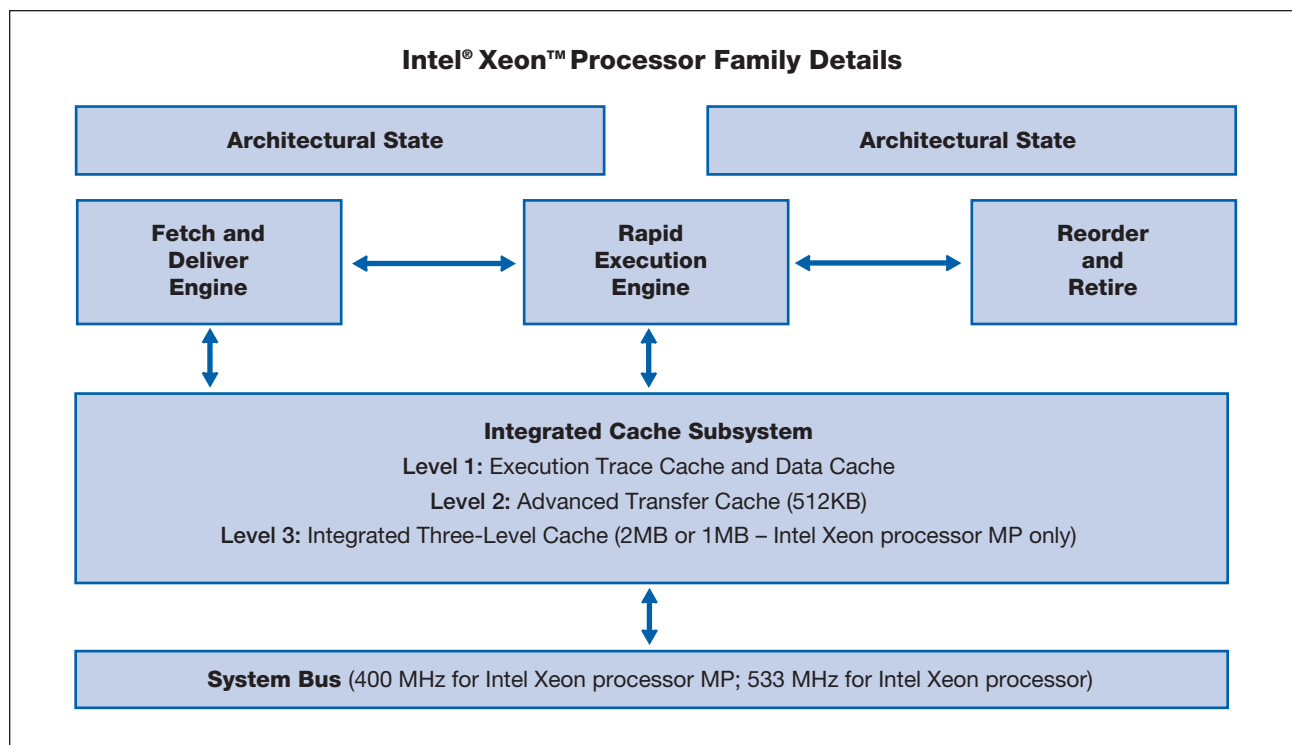


Figure 3: High-level block diagram of the Intel® Xeon™ processor family for servers shows how the various pieces of the microarchitecture relate to each other with Hyper-Threading Technology.

available for larger instruction and data sets, and significantly reduces the average memory latency, which improves performance for mid-range and high-end server applications.

The caches are shared by both logical processors and are designed to minimize potential cache conflicts through a high level of set-associativity, which helps ensure data is well-retained in the caches. The shared caches also provide the potential for super-linear speedups for some applications, where one logical processor may be able to prefetch instructions or data into the cache for the other logical processor, so that the other logical processor does not have to go to the system bus and fetch the instructions or data again from system memory.

Reorder and Retire Block

The Reorder and Retire block takes all the instructions that were executing out-of-order, and puts them back into program order, then commits the state of those instructions in program order. Instruction retirement alternates between logical processors. Instructions are alternately retired for one logical processor, then the other.

System Bus

The System Bus is designed to increase the throughput of multi-processing and multithreaded server applications and provide the necessary bandwidth for Hyper-Threading Technology when accessing system memory. It uses signaling and buffering schemes that allow for sustained data transfers. When one of the logical processors cannot find the data it needs in the Integrated Cache Subsystem, then the data must be transferred over the system bus from memory.

In systems with four or more processors, the system bus runs at 400 MHz and provides up to 3.2 GB/s bandwidth, which can be up to four times the previous-generation MP-based processor. For dual-processor (DP)-based server applications, which typically are not as data intensive or cache hungry and derive a greater benefit from increased frequency and I/O, the system bus now runs at 533 MHz. This provides up to 4.27 GB/s bandwidth, over four times that of earlier generations of DP-based processors.

How Server Platforms Use Hyper-Threading Technology

Server platforms based on the Intel Xeon processor family have implemented the necessary changes in the platform BIOS in order to recognize the logical processors so that the operating system and software can utilize Hyper-Threading Technology.

Figure 4 depicts a high-level block diagram of platform components. Within each Intel Xeon processor in the system, the two logical processors appear to the BIOS and multi-processor-aware operating system (OS) as processors available to the system and software applications. During the MP-based system initialization process (also called system boot), the BIOS counts and records the number of logical processors available in the system. The BIOS records only the first logical processor on each physical processor into the MPS (Multi-processor Specification) table to preserve backward compatibility with legacy operating systems. These legacy operating systems will use the MPS table only, and will not recognize the second logical processor on each physical processor. They will work as if Hyper-Threading Technology is not there.

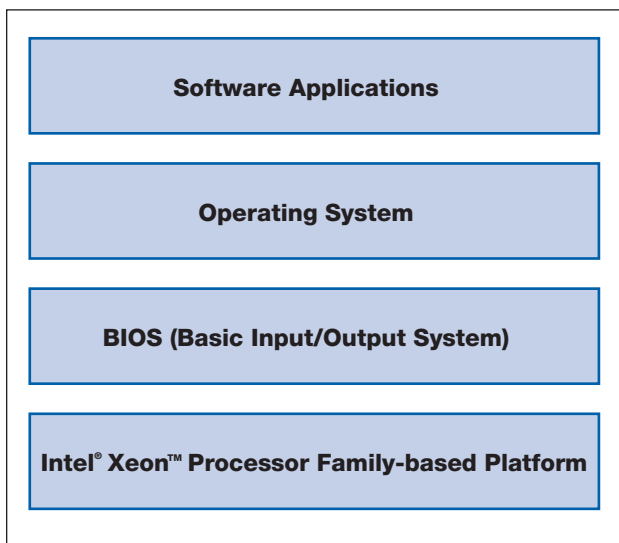


Figure 4: Platform components of Intel® Xeon™ processor family for servers include the processor and other system hardware components, BIOS, operating system, and software applications.

The BIOS then records two logical processors for each physical processor into the ACPI (Advanced Configuration and Power Interface) table. Operating systems that use the ACPI table can then proceed to schedule threads onto the logical processors that the software license allows the system to use. The BIOS and OS may report only the number of logical processors available in the system. Platform management software can typically be used to determine the number of physical processors available in the system.

Multi-processor-aware operating systems can use the same methods that they use today to schedule threads on MP-based systems. Figure 5 illustrates how the OS schedules the threads to the logical processors.

The Intel Xeon processor family with Hyper-Threading Technology is fully backward compatible with existing operating systems and applications. That is, legacy multi-processor-aware operating systems can run unmodified on Intel Xeon processor-based platforms. Some of these legacy operating systems (such as Windows NT*) may not recognize the second logical processor and may not take advantage of Hyper-Threading Technology, but are compatible with the Intel Xeon processor family and will run unchanged. Today's current operating systems (such as versions of Windows* 2000 Server, Linux* and Novell NetWare*) can recognize two logical processors and can utilize Hyper-Threading Technology, depending on OS license configurations (check with your operating system vendor for how Hyper-Threading Technology effects licensing in these versions). Many of the newer operating systems, either released or soon to be released (such as versions of Microsoft Windows*.NET Server, Microsoft Windows XP* Professional, and certain distribution versions of Linux such as Red Hat 2.4*), include further optimizations for Hyper-Threading Technology. These newer operating systems can distinguish fully between physical and logical processors, allowing them to offer improvements in scheduling and idle loops in order to realize optimal performance gains. Moreover, in these operating systems Hyper-Threading Technology enabled processors are recognized as single physical processors for licensing purposes (e.g. a four-way, Hyper-Threading Technology enabled server is recorded and licensed as four processors, not eight logical processors).

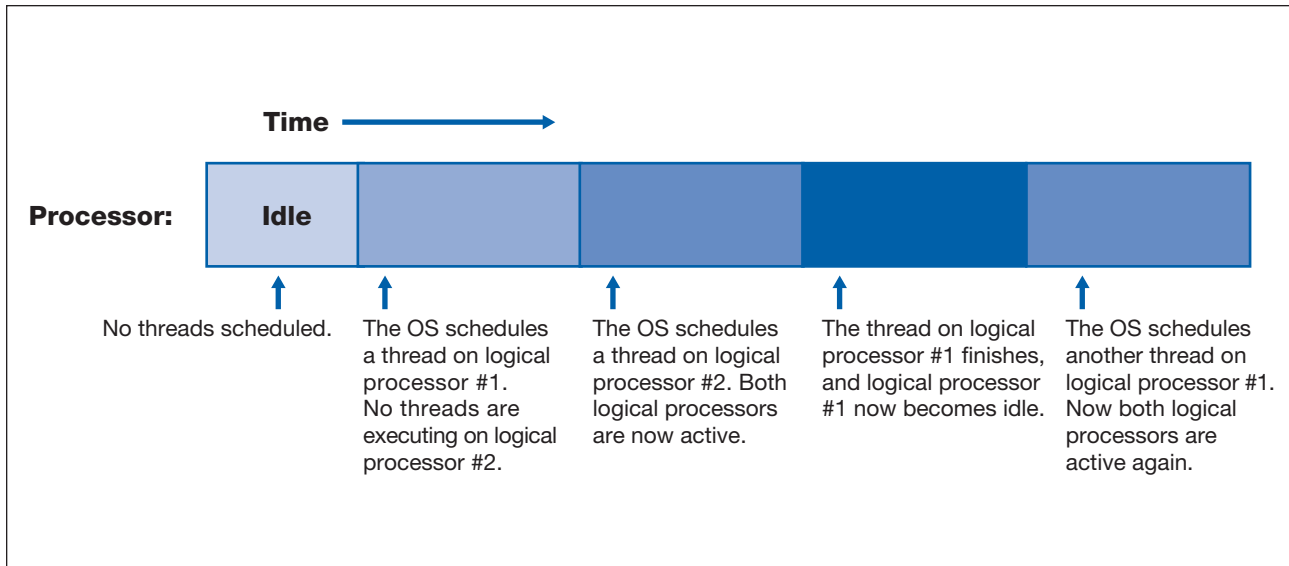


Figure 5: The operating system schedules threads on the two logical processors, alternating back and forth as threads complete.

How Software Takes Advantage of Hyper-Threading Technology

Today’s server software is multithreaded and can take advantage of Hyper-Threading Technology without changes. The software application uses the operating system to schedule threads across the multiple logical processors available in the system, much like it does on traditional MP-based systems. It’s the system BIOS and operating system that determine which logical processors are available for use. When considering Hyper-Threading Technology and its logical processors, software vendors that use per-processor license models should continue to base their models on physical processor count. Consult your software vendor to determine which versions and system configurations utilize Hyper-Threading Technology on Intel Xeon processor family-based platforms.

Applications that exhibit good threading methods and scale well on MP-based servers today are likely to take advantage of Hyper-Threading Technology. The performance increase seen is highly dependent on the nature of the application, the threading model it uses, as well as system dependencies. It has been shown that some server applications can experience up to 30 percent additional performance due to Hyper-Threading Technology in the Intel Xeon processor family implementation.

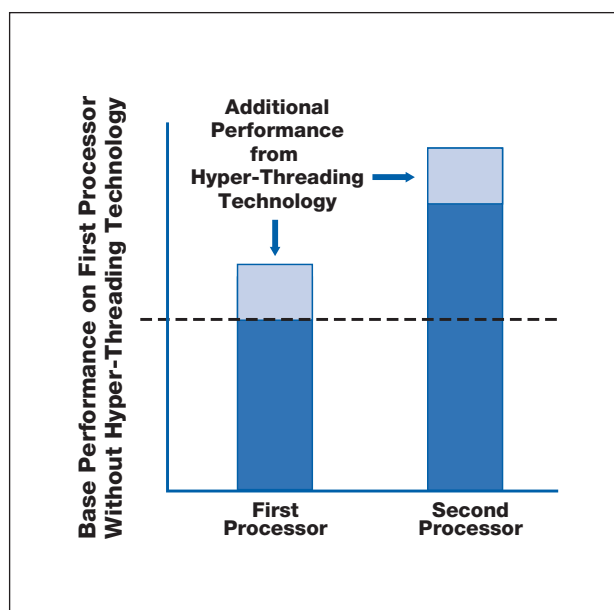


Figure 6: Performance gains attributed to Hyper-Threading Technology on a dual-processor (DP)-based system.

Figure 6 represents an example of additional performance offered by Hyper-Threading Technology in a multi-processing environment. Let’s start with base performance on one traditional processor in a DP-based system. With Hyper-Threading Technology enabled, performance on multi-threaded applications on the first processor can see an increase. If we added a second physical processor to the

traditional dual-processing system, we see that performance can increase above the level obtained using Hyper-Threading Technology on the first processor. Performance gains attributed to additional physical processors are typically greater because you have more dedicated processor execution resources to use (this has been shown to be as much as 80 percent in some cases). If we were to add a second physical processor with Hyper-Threading Technology enabled, we can see an additional performance gain on top of the traditional dual-processor performance. Similar to traditional dual-processor and multi-processor performance scaling results, the performance increase realized from Hyper-Threading Technology is highly dependent on the nature of the application, the threading model it uses, as well as system dependencies.

Hyper-Threading Technology complements the scalability offered in a traditional multi-processing environment offering higher levels of performance headroom. With parallelism and threading becoming more prevalent, server applications are expected to be further optimized for Hyper-Threading Technology and the Intel NetBurst microarchitecture to yield optimal levels of performance going forward.

Business Benefits of Hyper-Threading Technology

Hyper-Threading Technology can result in many benefits to e-Business and the enterprise:

- Improved reaction and response times for end-users and customers
- Increased number of users that a server system can support
- Handle increased server workloads
- Higher transaction rates for e-Businesses
- Greater end-user and business productivity
- Compatibility with existing server applications and operating systems
- Headroom to take advantage of enhancements offered by future software releases
- Headroom for future business growth and new solution capabilities

Solutions that run on both dual-processing and multi-processing servers can see benefits from the Intel Xeon processor family with Hyper-Threading Technology.

Dual-processing servers based on the Intel Xeon processor deliver compute power at unparalleled value and versatility for departmental servers, Internet infrastructure (for example, Web-caching, streaming media, search engine and security), mail server and Small & Medium Business (SMB) server applications. The combination of the Intel NetBurst microarchitecture and Hyper-Threading Technology provides outstanding performance and headroom for peak Internet server workloads, resulting in faster response times and support for more users.

Multi-processing servers (4 processors and greater) based on the Intel Xeon processor MP provide industry-leading compute power and throughput for critical e-Business and enterprise server workloads; such as customer relationship management, supply chain management, middleware and application servers, collaboration/e-mail, media servers, site server management, enterprise databases, and business intelligence. The combination of the Intel NetBurst microarchitecture, Hyper-Threading Technology, and the additional Integrated Three-Level Cache delivers outstanding throughput and headroom for larger server workloads, resulting in increased transaction rates, faster response times, support for more users, and improved availability and scalability for your e-Business and enterprise.

Summary

Hyper-Threading Technology is groundbreaking innovative technology that Intel has seamlessly integrated into today's industry-standard server infrastructure via the Intel Xeon processor family. With two logical processors available per each physical processor, multithreaded server applications can now take advantage of thread-level parallelism on each physical processor for additional performance on dual-processing and multi-processing servers.

Today dual-processing and multi-processing-aware operating systems and threaded server applications can run unchanged on the Intel Xeon processor family, and many can take advantage of Hyper-Threading Technology for immediate business benefits. Hyper-Threading Technology complements today's DP and MP-based servers by providing additional headroom for future growth and new server capabilities just on the horizon. Today and tomorrow's enterprise can utilize the Intel Xeon processor family with Hyper-Threading Technology for a competitive advantage.

Additional Information:

For additional technical details on Hyper-Threading Technology, please visit the Hyper-Threading Technology Home Page for a collection of white papers and presentations at: developer.intel.com/technology/hyperthread/index.htm.

You can also find additional information on the Intel Xeon processor family at the Intel® Business Computing Web site, at: www.intel.com/ebusiness.



Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, or life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.

THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

Intel disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein, except that a license is hereby granted to copy and reproduce this document for internal use only.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The Intel Xeon processor may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Copyright © 2002 Intel Corporation. All rights reserved. Intel, the Intel logo, Intel NetBurst, and Intel Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.