
White Paper
Research at Intel

# From a Few Cores to Many:

## A Tera-scale Computing Research Overview

**Editors:**

**Jim Held**

Intel Fellow, Director, Intel Tera-scale Computing Research Program

**Jerry Bautista**

Co-Director, Intel Tera-scale Computing Research Program

**Sean Koehl**

Technology Strategist, Intel Tera-scale Computing Research Program

Intel Corporation

**Contributors:** Ali Adl-Tabatabai, Pradeep Dubey, Dave Dunning, Mike Espig, Ed Grochowski, Antonio Gonzalez, Scott Hahn, Ram Huggahalli, Jay Jayasimha, Akhilesh Kumar, Partha Kundu,Tim Mattson, Derek McAuley, Alberto Munoz, Chuck Narad, Don Newell, R. M. Ramanathan, Thom Sawicki, Sebastian Schoenberg, Ioannis Schoinas, John Shen, Jim Sutton, Manny Vara, Mona Vij.

# 1. Introduction

Intel processors with two cores are here now, and quad-core processors are right around the corner. In the coming years, the number of cores on a chip will continue to grow, launching an era of vastly more powerful computers. These are the machines that will deliver teraflop performance with the efficient capabilities needed to handle tomorrow's emerging applications.

The Intel® Tera-scale Computing Research Program is Intel's overarching effort to shape the future of Intel processors and platforms. Intel researchers are already working on over 100 R&D projects worldwide to address the hardware and software challenges of building and programming systems with dozens of energy-efficient cores with a sophisticated memory hierarchy. Currently, projects in this program span circuit technologies, microarchitecture, interconnects, memory, and software technologies.

Some may wonder why Intel is putting so much emphasis on tera-scale research. The reason is simple: tera-scale computing is a two-fold revolution, both in the capabilities that devices will have, and in the amount of innovation that will be required to handle tomorrow's advanced applications. The term itself—tera-scale—refers to the terabytes of data that must be handled by platforms capable of teraflops of computing performance. That's a thousand times more compute capability than is available in today's giga-scale devices.

Why such a leap forward? Because incremental improvements in performance and capabilities simply won't support real-time data mining across teraflops of data; artificial intelligence (AI) for smarter cars and appliances; virtual reality (VR) for modeling, visualization, physics simulation, and medical training; and other applications that are still on the edge of being science fiction. Also, data stores are becoming larger and more complex. In medicine, a full-body medical scan already contains terabytes of information. Even at home, people are generating large amounts of data, including hundreds of hours of video, thousands of documents, and tens of thousands of digital photos that need to be indexed and searched. Tera-scale computing is the way to bring the massive compute capabilities of supercomputers to everyday devices, from servers, to desktops, to laptops.

For example, with a tera-scale computer, you could create studio quality, photo-realistic 3-D graphics in real time. Or you could manage personal media better by automatically analyzing, tagging, and sorting snapshots and home videos. Advanced algorithms could be used to improve the quality of movies captured on older, low-resolution video cameras. An advanced digital health application might assess a patient's health by interpreting huge volumes of data in a scan and aid in making decisions in real time.

The essential aspect of tera-scale technologies—and the heart of Intel's research—is being able to do such complex calculations in real-time, primarily through the execution of multiple tasks in parallel. That is the fundamental requirement for the complex and compelling applications we will see in the future.

## 1.1 Motivation

In the last twenty years, Intel has delivered dramatic performance gains by increasing the frequency of its processors, from 5 MHz to more than 3 GHz, while at the same time, improving IPC (instructions per cycle). Recently, power-thermal issues—such as dissipating heat from increasingly densely packed transistors—have begun to limit the rate at which processor frequency can also be increased. Although frequency increases have been a design staple for the last 20 years, the next 20 years will require a new approach. Basically, industry needs to develop improved microarchitectures at a faster rate, and in coordination with each new silicon manufacturing process, from 45 nm, to 32 nm, and beyond.

For this new approach we can take advantage of Moore's law. Transistor feature size is expected to continue to be reduced at a rate similar to that in the past. For example, a 0.7x reduction in linear dimensions enables a 2.0x increase in the transistor density. Thus, we should assume that with every process generation, we will be able to build chips with twice the number of transistors as on the previous process generation. New technologies, such as 3-D die stacking, may allow even greater increases in total transistor counts within a given footprint, beyond the increases made possible by improvements in lithography alone.

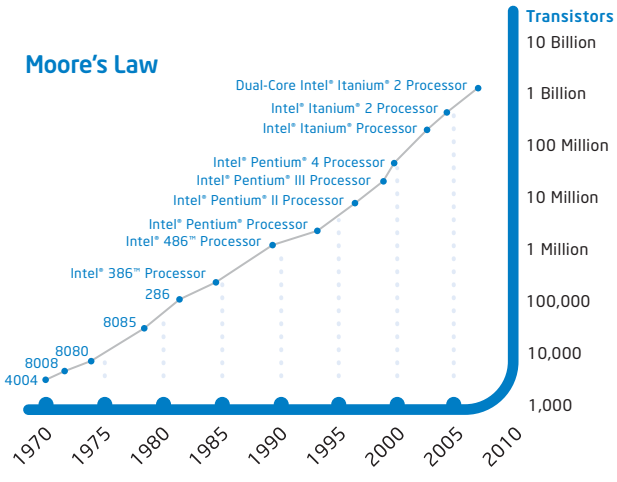



**Figure 1. Scaling transistors.** The number of transistors is expected to continue to double about every two years, in accordance with Moore's Law. Over time, the number of additional transistors will allow designers to increase the number of cores per chip.

With so many transistors available, we have already began to design chips with multiple processor cores (also called CMP or chip-level multiprocessing). Instead of focusing solely on performing individual tasks faster, we will execute many more tasks in parallel at the same time. We will also distribute those tasks across a grouping of cores that work in a coordinated fashion. Three distinct trends motivate this shift:

- **Performance:** We can no longer simply increase the clock frequency (processor "speed") at the same rate as we have in the past in order to increase performance. Power and thermal requirements are beginning to outstrip the benefits that faster clock frequencies offer. However, because the trajectory of Moore's law will continue well into the next decade, we expect to continue doubling transistors every 18-24 months for the next several years. Parallel execution in multi-core designs will then allow us to take advantage of these greater transistor densities to provide greater performance.
- **Power consumption:** Many simple cores can be built within the same area as a small number of large complex cores. In addition, power consumption can be optimized by using multiple types of cores tuned to match the needs of different usage models. Also, cores that are not busy can be powered down to reduce power consumption during idle times. These advanced power-saving techniques are enabled by multiple cores working in a coordinated fashion.
- **Rapid design cycles:** Building tera-scale processors from standard, repeated tiles that are then integrated into a common infrastructure should allow more reuse of designs between generations of processors. This will also allow us to highly optimize the design of these tiles to further improve power utilization and performance.

Since Moore's law is expected to continue to deliver more transistors every process generation, and since platform power and energy budgets will be increasingly limited, the trend is to deliver increased performance through parallel computing. Essentially, to achieve the desired improvements in performance without a corresponding increase in energy bills, we must increase the efficiency and number of cores on a chip, rather than increase clock frequency.

## 1.2 New Opportunities

Obviously, tera-scale architectures present many challenges. They also offer many unique opportunities through their highly integrated multi-core designs.

For example, consider on-chip core-to-core communication latencies and bandwidth. These will be orders of magnitude better than the chip-to-chip multiprocessing systems used today for parallel computing.

The number of cores in a chip can also reach levels well beyond those of traditional high-end servers. This can offer significant benefits in both performance and cost reductions over systems with traditional single or dual processor systems.

Finally, the very high computational power of a tera-scale processor, combined with the modularity of many cores, gives designers the flexibility to dedicate hardware resources (such as one or more cores) to specific functions. These could be system management or single-use devices, such as a Voice-over-IP appliance (for internet telephony), cryptography, or media encoding/decoding.

# 2. Tera-scale Computing Research Areas

Intel has identified several key attributes that will be required of future tera-scale platforms:

- **Programmability.** Without optimized software, tera-scale platforms will not live up to their potential. Platforms must effectively address the needs of new and existing programming models. This also includes software development/debug tools, as well as new performance benchmarks consistent with highly parallel execution.
- **Adaptability.** The platform must be able to change configuration to match varied usage and workloads, as well as adapt to changes in the hardware environment, such as from power and thermal factors.
- **Reliability.** The platform must preserve current levels of reliability or increase its reliability despite the increased complexity inherent in these platforms.
- **Trust.** The platform must provide a trustworthy environment, despite its flexibility and the complexity of its design.
- **Scalability.** The platform must deliver performance that increases in proportion to the number of cores, with hardware and software that also effectively scales.

To help develop platforms that meet these requirements, the Intel Tera-scale Computing Research Program encompasses projects around the globe, and addresses challenges in the three critical areas of microprocessors, platforms, and software development. Among others, these challenges include optimizing designs for highly parallel and multithreaded workloads, developing scalable on-die interconnects, redesigning cache and memory hierarchies, improving I/O, and identifying and exploiting parallelism via software development.

## 2.1 Microprocessor Research Areas

One of Intel's goals is to deliver a highly modular, scalable tera-scale processor architecture that can address the broad range of Intel platforms within future power constraints. There are four keys to delivering this architecture: optimized core designs, fixed-function units, scalable on-die interconnects, and effective energy management.

### 2.1.1 Core Designs

One of the differentiating aspects of tera-scale architectures is the use of a core microarchitecture that is optimized for highly-parallel, multithreaded workloads. Simultaneous multithreading can help deal with memory latencies and help reduce or eliminate out-of-order complexity.

At Intel, researchers are working to determine the optimal core design or designs needed for tera-scale architectures. Researchers are also working to identify the number of cores than can be effectively exploited in a given manufacturing process generation (such as 45 nm, 32 nm, or 22 nm). Of special importance is research aimed at including in each core special features/mechanisms that will facilitate the exploitation of thread-level parallelism.

Legacy software and inherently single-threaded algorithms present another challenge to tera-scale architectures. To address that issue, Intel researchers are exploring ways to incorporate heterogeneous general-purpose cores (both single-thread and multithread optimized cores) into tera-scale architectures. In addition, researchers are exploring the use of ensembles of simple cores to accelerate single threads through thread-level speculation and other advanced techniques.

By varying the mix of functional elements, the architecture can allow designers to create multiple implementations that match specific market needs. If the processor uses regular tiles that consist of one or more cores, cache, a router, and other supporting hardware, different aspects of a layout could be reused from processor to processor.

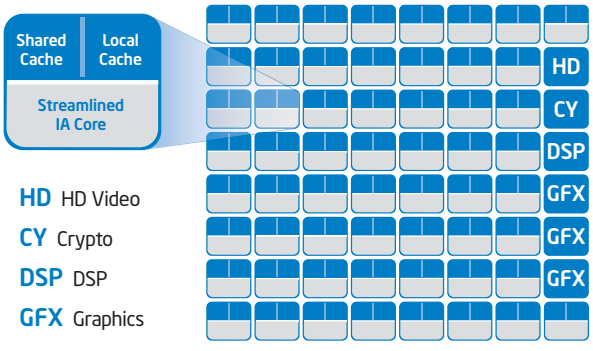



**Figure 2.** Future tera-scale chips will use an array of tens to hundreds of cores with reconfigurable caches, as well as special-purpose hardware accelerators.

### 2.1.2 Fixed-Function Units

The demands of power efficiency suggest that in some instances it will be worthwhile to create fixed-function units. Fixed-function units are processing elements dedicated to a particular workload. They will perform better for that task than a general-purpose execution unit would perform (for example, a variable-length bit manipulation or video decode logic). These functions may be fixed logic or programmable engines that are optimized to a particular purpose, such as graphics engines to accelerate texture processing. Other possibilities under research include various network accelerators, cryptography engines, and physics engines.

With fixed-function units, a given tera-scale platform can have a processor with a mix of element types matched to the significant workloads in the segment it is to serve. Both the number of each and the relative number of each type could be varied.

### 2.1.3 Scalable On-Die Interconnect Fabric

A tera-scale processor needs to connect, not only to a large number of cores, but also to caches and special-purpose hardware, such as graphics units. Basically, as more elements are packed onto a tera-scale chip, there is a correspondingly greater need for each of these units to communicate with each other. Studies have shown that the interconnect design and microarchitecture will play a significant role in determining the net performance of a tera-scale architecture.

In particular, tera-scale architecture will require a modular on-die (or on-chip) interconnect "fabric" that is scalable to support a variable number of cores. Interconnects must also be robust in the face of failures, and have an architectural lifetime that can span several generations of processor designs. Finally, the interconnects must have efficient power management that scales down power consumption to match the more efficient utilization of the cores. On-chip interconnect networks are expected to have several advantages over today's off-chip core interconnects:

- On-chip wires are cheap and plentiful compared to off-chip networks in printed circuit boards.
- On-chip interconnects can often be routed on top of some types of circuit structures (such as caches), so that they consume little on-die real-estate.
- Given the smaller distances on-die interconnects must span, such interconnects are generally more responsive and power efficient than off-die interconnects.
- On-die bus widths can be much wider than those on printed circuit boards, allowing for more efficient, lower modulation speeds. In other words, bandwidths can be increased by widening the bus and actually slowing the modulation speed to save power.

Because on-die interconnects can support a wide number of cores, they can serve many market segments, ranging from large server devices to small mini-notebook processors. The interconnect architecture would support modular designs (with a standard interface to the various compute elements), and offer performance headroom to scale over a decade in order to support several process generations. Such characteristics are typical of platform interconnects such as PCI, which further highlights the essential nature of a tera-scale processor as a platform/system on a chip.

### 2.1.4 Energy Management

A tera-scale architecture would offer new opportunities for energy management. For example, the availability of numerous, modular cores will allow the system to migrate work as needed to balance out utilization across the tera-scale processor, or to match workload to the type of core. Effective management of thermal loads in individual components of a tera-scale processor will also improve the overall reliability, as well as improve overall performance by evenly spreading out the thermal load.

At the circuit level, Intel is researching many critical areas. These include designs for circuits for on-die caches and on-die memory that are much more efficient in their use of power. Prototype register file circuits are already showing up to a 3X improvement in the speed of memory accesses, and up to 4X to 5X reduction over today's circuits in the power they consume. Intel's goal is a 10X improvement in performance per watt over the next 10 years.

Tera-scale processors may also have the ability to throttle the energy efficiency for a given compute task. This means noncritical or highly parallel workloads could be run at a reduced frequency, and thus much lower energy consumption. For critical code or segments of highly serial code, performance could be boosted by temporarily throttling up the frequency.

## 2.2 Platform Research Areas

Balancing a system with the tremendous compute density of a future tera-scale processor will be challenging. For example, getting data in and out of a tera-scale processor in an efficient manner requires balancing cache and memory bandwidth, I/O bandwidth, and connectivity. Essentially, a tera-scale processor needs to be able to move tera-bytes of data onto and then later off the chip. As the number of cores grows, it will be increasingly challenging to provide I/O with a performance level that scales to the number of cores.

Tera-scale platform hardware will also need to support an execution environment which provides power management, partitioning, resilience, and other features of the full system.

### 2.2.1 Cache and Memory Hierarchy

One major set of changes to platform design will be in the memory hierarchy. Specifically, the cache hierarchy for a tera-scale device will include private first-level caches (L1 caches) per core, as well as several levels of shared cache (L2 cache). Research in these areas includes work on shared distributed caches, cache policies (including data-specific policies), and cache partitioning.

In addition, a tera-scale processor will most likely require an additional level in the memory hierarchy to match its bandwidth requirements to the system memory characteristics. Here, researchers are exploring the use of 3-D stacking to provide a large, low-latency, last-level cache using SRAM or DRAM.

### 2.2.2 I/O

Tera-scale platforms also create new challenges associated with I/O. For example, the I/O of a tera-scale device will have to scale to match the compute density (the number of transistors for a given footprint) of its architecture. This means the device must be able to handle the potential aggregate bandwidth required when all cores are operating at the same time.

While providing the potential aggregate bandwidth raises considerable challenges, issues of connectivity and sharing will be even more difficult to resolve. This is because the level of I/O sharing is exaggerated due to the presence of more cores and threads per processor.

At the physical layer, copper I/O interconnects must scale to support hundreds of gigabits per second. This performance is required to support the expected tera-scale workloads. Intel is already researching ways to scale existing copper I/O interconnects to tens of gigabits per second per line.

Hardware and software mechanisms for device replication and sharing will also need to scale. These mechanisms will have to support dozens to hundreds of threads or virtual machines that are concurrently sharing a device. The challenge is that most practical configurations incorporating tera-scale processors will be limited to one or a small number of disks. To address this issue, Intel researchers are also looking at ways to incorporate into storage subsystems the hardware and software mechanisms that would provide the necessary bandwidth and I/O per second.

A tera-scale I/O architecture must also be flexible and versatile enough to balance performance, cost, and power across the many workloads, benchmarks, and usage models. Balanced I/O refers to providing I/O services in proportion to the compute demands of applications running on the platform, so that the I/O subsystem does not critically limit the application's performance. To be balanced, I/O as a subsystem must also meet the above requirement with a minimal cost in silicon, software, and energy consumption.

Intel has been conducting extensive research on ways to provide performance I/O for tera-scale devices through technologies such as CMOS radio and silicon photonics. CMOS radios would provide the capability to integrate flexible, multistandard wireless capabilities into components such as chipsets. Likewise, Intel's recent advancements in silicon photonics are aimed at making high-bandwidth fiber-optic links affordable for connections in and around desktops and servers.

#### 2.2.3 Virtualization and Partitioning

Intel's vision for tera-scale platforms includes support for both hardware partitioning and virtualization. These capabilities will support special usage models, as well as improve robustness and trust through tamper-resistant, hardware-based isolation. The virtualized platforms will have full support for efficient access to such partitions from and to I/O devices.

Applications that process real-time media may require performance guarantees in order to provide the required quality of service (QoS) to the user. Because of this, performance isolation among partitions or among virtual machines will be important, especially for client tera-scale platforms.

#### 2.2.4 Execution Environment

The execution environment encompasses the system software and firmware that controls access to shared resources on the platform. These resources include compute resources (cores), storage resources (file systems and disk), and memory.

The software and firmware elements of the execution environment include the operating system, hypervisor or virtual machine monitor, and any firmware that is used to partition the system. As the tera-scale architecture increases the parallel nature of workloads, the way elements access shared system resources must change to ensure the availability of the resources. Each element is responsible for making sure that its workload receives access in such a way that meets the needs of other workloads running on tera-scale platforms. In other words, each element must share resources appropriately in order to allow other components to access the same resources.

As with other aspects of a tera-scale architecture, the elements of the execution environment must scale with the number of cores. In addition, the elements must be able to adapt to platforms where resources are being dynamically added or removed.

### 2.3 Software Development Research Areas

Developing software to take advantage of tera-scale hardware includes two of the greatest challenges for tera-scale computing.

The first challenge is to ensure that there are compelling applications and workloads that exploit the massive compute density, using parallel algorithms and potentially masses of data. Intel researchers are already working to identify and enable such workloads and algorithms. The goal here is to distill the characteristics of potential workloads and algorithms, then develop technologies that facilitate application development and scale them onto tera-scale architectures platforms.

The second challenge is that multiprocessing adds a time dimension that is extremely difficult for software developers to cope with and extremely difficult for validation engineers to test. The large-scale concurrency of such devices will only increase software development challenges—such as implementing correct synchronization among threads without deadlock and race conditions in accessing shared memory. These are very subtle errors that are hard to reproduce. Intel is researching software technologies such as transactional memory to address these types of programming needs. In addition, Intel is working with universities worldwide to expand curricula to include multithreaded software development and train future generations of developers.

Certainly not all legacy applications will be rewritten to take advantage of a parallel programming environment. To address the legacy issue, Intel researchers are working on compiling techniques and runtime environments that automatically parallelize existing, single-threaded code. Such techniques have limited benefits relative to building the code from the ground up with a parallel execution model in mind. However, the techniques may allow single threaded applications to take some advantage of the higher performance capabilities of multi-core, tera-scale devices.

#### 2.3.1 Future Workloads

Future usage models will define the requirements of tera-scale platforms. As such, industry needs effective models for what the software workloads will be and how those workloads will fit together. Intel researchers are currently working to identify and model the emerging workloads and how they will interact with specific features of the tera-scale architecture.

While existing single-threaded workloads are important, new applications will be enabled by tera-scale processors. Compute densities on future tera-scale processors will be significantly greater than the densities found on current microprocessors. This will help developers shift applications that were once possible only on high-end supercomputers, onto laptops or even handheld devices.

To this end, Intel is investigating classes of processing capabilities that will be needed to handle tera-scale workloads. Intel classifies these processing capabilities into three fundamental types: recognition, mining and synthesis (RMS).

- **Recognition:** Machine-learning capabilities that allow computers to examine data and images and construct mathematical models based on what they identify. An example might be a model for the face of a specific person.
- **Mining:** The capability to sift through large amounts of real-world data related to the patterns or models of interest. Put more simply, it is the ability to find an instance of a specific model amidst a large volume of data. For example, mining could mean finding a particular person's face from a large number of images of various resolutions, lighting environments, and so on.
- **Synthesis:** The capability to explore theoretical scenarios by constructing new instances of a model. For example, this could be projecting what the target person's face might look like if they were younger or older.

An example of an RMS workload is computer-assisted care of the infirm or the elderly. With standard web cameras wirelessly connected and distributed around a house, a tera-scale computer could recognize the person under supervision and model expected behavior. For example, the computer could help make sure the person self-medicates at appropriate intervals, gets a glass of water to drink after so much time has passed, or doesn't leave the stove on and unattended for more than a few minutes. Such modeling applications could help reduce health risks for the elderly and allow for increased self-reliance. The challenge with such applications is that the compute requirements are intense. Advanced image processing, inferential logic, anticipation of potentially dangerous scenarios, and the security to ensure privacy for sensitive information—these require extremely intensive compute performance.

RMS workloads are of particular interest to Intel since they show a great deal of promise to deliver user value via a common set of algorithms that scale. The RMS application model appears to be general in nature, applying to a very large number of compute needs that span high-performance computing, digital content creation, computer vision, and artificial intelligence. Essentially, these workloads have certain system and architectural commonalities. By deeply studying the underlying RMS algorithms and kernels, Intel researchers can deduce the key architectural requirements that benefit this class of applications. In turn, this will help researchers turn an understanding of RMS into improved platform architectures for the future.

Regardless of how computing evolves over the next decade, people will still use web servers, office productivity applications, and databases. Intel continues to study these more traditional workloads to understand how they relate to future tera-scale architectures.

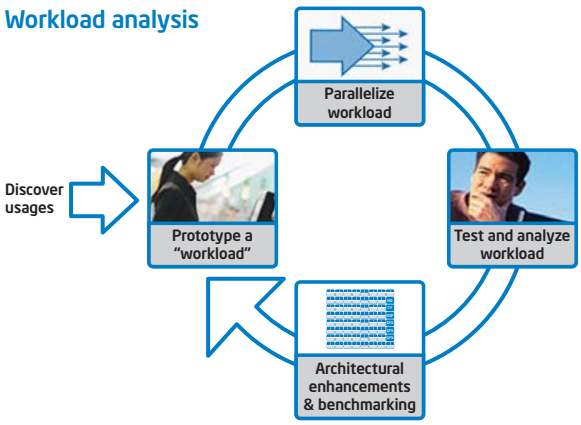



**Figure 3.** Designing realistic prototype applications allows Intel researchers to test multithreaded scalability and feed key learning into future hardware designs.

### 2.3.2 Programming Environment

One of the more difficult challenges may be efficiently writing software that gains the benefit of multi-core processing. In the tera-scale future, software should be designed to use available parallelism to gain the performance benefit of the increased numbers of cores. This requires that software developers design parallel programs, a traditionally time-consuming and error-prone task which requires developers to think differently than the way they do today. Teaching mainstream and future developers to identify and then effectively exploit parallelism is something Intel must foster if these skills are to move from a narrow domain of high-performance computing (HPC) experts into the mainstream.

One of the primary issues is that almost all of the elements associated with the programming environment will have to change in order to accommodate scalability and concurrency. Compilers and tools such as debuggers and performance analyzers will need to support new concurrency constructs. The programming environment itself will need to provide new language constructs, as well as new tools to help the programmer uncover parallelism and avoid parallel programming bugs (such as deadlocks and race conditions). Language runtimes must also scale, provide new concurrency primitives for the compiler, and expose new interfaces to tools. Finally, libraries will have to provide new APIs that make it easier to develop robust and scalable applications. Beyond the more common shared-memory model of SMP (symmetric microprocessor) programming, Intel is investigating tools to support programming models that have been successfully employed in existing parallel programming environments. These environments include the streaming programming model used with GPUs (graphics processing units), and the distributed-processing model based on message processing that is common in HPC.

### 2.3.3 New Technologies to Support Parallel Programming

To simplify software development, Intel is researching new features for tera-scale devices. For example, transactional memory simplifies parallel programming by reducing the need for software developers to manage explicit locks. A key focus of Intel's research is finding other hardware features that will simplify parallel programming.

One of the features of tera-scale architectures will be dedicated partitions that appear as devices to regular software. These partitions will provide functions such as system management and I/O acceleration (such as network protocol processing). The architectures may also include hardware support for lightweight message passing for a distributed-computation model that is familiar to HPC software developers. This type of streaming has proven to be an effective programming model for workloads such as graphics and media processing. Tera-scale architectures will exploit that familiarity by providing hardware support. For example, the architectures will support the cache behavior desired, as well as coprocessor instruction access to fixed-function media acceleration.

In addition, as with the MMX extensions for multimedia applications, there will be opportunities to extend the ISA (Instruction Set Architecture) to better support emerging RMS workloads.

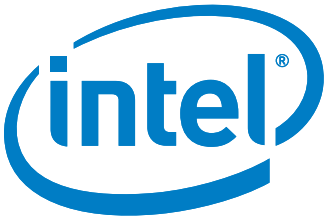
# 3. Summary

Intel foresees future devices revolving around tera-scale architectures that use tens to hundreds of cores to process massive amounts of information in parallel. By making these devices scalable, adaptable, and programmable, Intel can help lead industry into a new era where more immersive, interactive, and useful applications become possible.

In a tera-scale world, there will be new processing capabilities for mining and interpreting the world's growing mountain of data, and for doing so with even greater efficiency. Intelligent agents could advise users in real-time on stock trades and other financial decisions. Such agents could search massive collections of digital videos to find specific people or events, and even edit a new video based on what the user wants to see. For gamers, there is the obvious benefit of photo-realistic, real-time graphics. However, even those benefits aren't just for gamers anymore. Interactive virtual environments are now being developed for both collaboration and education, such as learning a language by interacting with virtual native speakers, or training a doctor to deal with an emergency situation on a simulated human body.

There are people who still question a multi-core, tera-scale future, and why tomorrow's applications need so many threads. The answer is that those advanced, intelligent applications require supercomputing capabilities, and the accompanying parallelism that allows those applications to be processed in real-time. This is a massive shift in what mainstream devices—servers, desktops, and mobile—can do, and it requires an equally massive shift in hardware and software.

Tera-scale computing is not a new goal, but the fulfillment of Platform 2015. Intel's long-range vision for the collective evolution of computational technologies, interfaces, and infrastructures, and the architectural innovation and core competencies will enable that evolution. Through the Tera-scale Computing Research Program, Intel has dedicated a significant amount of effort towards developing the technologies to realize this vision. The resulting discoveries and successes will not only shape the future of Intel micro-architecture, but will guide the capabilities of the underlying platforms, and allow the possibilities of the future to become reality through revolutionary applications.



**www.intel.com**

## References


Xiao-Feng, Li Zhao Hui Du, Chen Yang, Chu-Cheow Lim, Tin-Fook Ngai, "Speculative parallel threading architecture and compilation", pp 285- 294, International Conference Workshops on Parallel Processing (ICPP 2005), 2005.

Murali Annavaram, Ed Grochowski, John Shen, "Mitigating Amdahl's Law through EPI Throttling," pp. 298-309, 32nd Annual International Symposium on Computer Architecture (ISCA'05), 2005.

D. Nelson, C. Webb, D. McCauley, K. Raol, J. R. II, J. DeVale, and B. Black, "A 3D Interconnect Methodology Applied to iA32-class Architectures for Performance Improvements through RC Mitigation," in Proceedings of the 21 st International VLSI Multilevel Interconnection Conference, Waikoloa Beach, HI, USA, September 2004.

Paul Reed, Gus Yeung, and Bryan Black. "Design Aspects of a Microprocessor Data Cache using 3D Die Interconnect Technology." In Proceedings of the International Conference on Integrated Circuit Design and Technology, pages 15–18, Austin, TX, USA, May 2005.

L Hsu, R Iyer, S Makineni, S Reinhardt, D Newell, *"Exploring the cache design space for large scale CMPs"*, pp 24-33, ACM SIGARCH Computer Architecture News, Volume 33, Issue 4 (November 2005).

Berna L. Massingill, Timothy G. Mattson, and Beverly A. Sanders; "Reengineering for Parallelism: An Entry Point into PLPP (Pattern Language for Parallel Programming) for Legacy Applications"; Proceedings of the Twelfth Pattern Languages of Programs Workshop (PLoP 2005), 2005.

Pradeep Dubey, "Recognition, Mining and Synthesis Moves Computers to the Era of Tera" in Intel Technology Journal, February, 2005.

Ali-Reza Adl-Tabatabai, Brian T. Lewis, Vijay Menon, Brian R. Murphy, Bratin Saha, Tatiana Shpeisman, "Compiler and runtime support for efficient software transactional memory" pp 26-37, Proceedings of the 2006 ACM SIGPLAN conference on Programming language design and implementation.

Y. Chen, Q. Diao, C. Dulong, W. Hu, C. Lai, E. Li, W. Li, T. Wang, and Y. Zhang. Performance scalability of data mining workloads in bioinformatics. Intel Technology Journal, 09(12):131–142, May 2005.


## More Info

Learn more about tera-scale computing research. Visit www.intel.com/go/terascale.

Learn more about other Intel research. Visit www.intel.com/research.



Printed in the United States. 0906/MLG/HBD/PDF 315297-001US