The Advantages of IA-64 for Business Intelligence Applications

Information for Software Developers and IT Managers

Version 1.0 11/17/99

intel®

The Advantages of IA-64 for Business Intelligence Applications

November 1999

The need for business intelligence, or decision support system (DSS), applications to help businesses make decisions and analyze their data is growing as rapidly as the data being collected by modern e-Businesses. Databases ranging from hundreds of Gigabytes to tens of Terabytes can form the heart of these marts and warehouses. However, search-intensive decision support queries can slow even the most robust computing system.

To solve this problem, business intelligence applications must take optimum advantage of the best available computing technology. This is the case for either a standard query that the database schema handles, or an undisciplined query that may take many hours, or even days, to satisfy. Today, business intelligence applications are also a key part of e-Business solutions where customer behavior information leads to competitive advantage.

The IA-64 architecture and its first microprocessor, the Intel Itanium™ processor, provide many advantages that can improve the performance and scalability of business intelligence applications.

The Challenge

Compared to a memory-resident database, it is difficult to provide speedy access to mart or warehouse data that is disk-resident. With disk based data, there can be an intolerable delay to the many database accesses that a business intelligence application requires. Fortunately, it is possible to overcome the slow access problem.

Some of the most common data marts (with tens of Gigabytes or less) can be fully stored in memory. With the price of random access memory coming down to more affordable levels for many systems, servers can provide more memory for data-intensive applications. Moreover, if the value of the business intelligence application justifies the additional memory cost, much more data can be held in physical memory.

Entire databases can be stored in physical memory if the system processors can provide the needed linear address space. Traditional 32-bit processors typically address a maximum of 4 Gigabytes. A 64-bit processor addressing capability provides millions of Terabytes of address space.

However, 64-bit addressability alone is not sufficient. Business intelligence applications need more than just a large address space. Moreover, they need more than just the fast clock speeds and pipelining that many microprocessors offer. The problem with available microprocessors that offer just 64-bit addressing is that their computing cores can spend an inordinate amount of time waiting for data to arrive to fill the pipelines. The IA-64 architecture was designed to work closely with compilers to take advantage of the unique architecture features that improve overall performance of applications such as business intelligence.

IA-64 Architecture's Contribution to the Solution

In addition to its 64-bit addressing capability, the forthcoming Itanium processor provides additional architectural features for optimizing the critical code paths in business intelligence applications. These

features include a large number of registers, instruction level parallelism, predication, control and data speculation, and prefetch capability.

Designed by Intel to provide the best combination of RISC and CISC architectures, the IA-64 architecture also adds a number of unique features. The IA-64 architecture enables a greater number of instructions to execute in parallel through instruction level parallelism and an inherently scalable instruction set. In addition to these desirable features, the IA-64 architecture allows even more specific capabilities for enhancing business intelligence applications. These capabilities are summarized in Table 1.

Table 1: IA-64 Features and their Benefit for Business Intelligence Applications

IA-64 Architecture Feature	Business Intelligence Application Benefit
64-bit addressing	Large addressing space allows databases to be stored completely in physical memory or provides large databases caches for disk resident databases. Tens of gigabytes may be stored in physical memory for speediest access.
Large number of registers	Business intelligence application data and intermediate calculations stored in on-chip registers reduces the repetitive load and store of data values to improve response time of an application's database request.
Instruction level parallelism	Ability to execute many instructions in parallel allows simultaneous, speedy access and manipulation of data/intermediate values derived from multiple rows and columns of a large database table or tables.
Predication	Predication allows the conditional execution of instructions before it is known whether the execution is needed. Predication allows more code to execute in parallel, the performance penalty of branch-dependent code is less, and typical business intelligence applications with heavy branching speed up significantly.
Control and data speculation	Control speculation allows certain load instructions to be scheduled before conditional branch instructions, rather than after. Data speculation is similar to control speculation but allows loads to be scheduled above stores. Both allow a reduction in the processor wait states generated by branch-intensive code with high latency RAM accesses, thus increasing business intelligence application performance.
Prefetch	Instruction prefetches can be signaled on branch instructions. Data can be prefetched with explicit prefetch instructions. Both prefetches speed business intelligence application performance by reducing wait states.

Large Number of Integer Registers

While other RISC processors typically only have 32 registers, the Itanium™ processor has 128 integer, 128 floating point, 8 branch, and 64 predicate registers. The use of all these registers allows more selected data mart or data warehouse data and intermediate calculations to be stored in on-chip registers and

reduces slower and repetitive memory load and store of intermediate data values. These capabilities combine to improve the response time of a business intelligence application's query or calculation.

The IA-64 architecture's innovative register model provides great flexibility compared to traditional registers since they can be both "rotated" and "framed" registers. Rotating registers enable efficient loop execution important to business intelligence applications. They achieve the equivalent of software pipelining so that many loop iterations execute simultaneously with reduced cache misses and subsequent greater performance. Register frames easily handle the call-intensive code found in many business intelligence applications. Variable size blocks can be set up in the registers which minimizes performance penalties due to excessive register save/restore operations and ensures efficient use of the register resources.

Instruction Level Parallelism

Traditional hardware-based instruction execution scheduling cannot efficiently schedule parallel instruction execution. With the IA-64 architecture's explicit instruction level parallelism, an IA-64 compiler automatically produces parallel machine code. This ability to execute instructions in parallel allows the IA-64 architecture to simultaneously speed access to, and manipulate the data derived from, multiple rows and columns of a large database table or tables. Multi-million row table access and manipulation is common in certain business intelligence applications.

To best take advantage of this kind of explicit instruction parallelism, the IA-64 architecture provides the previously mentioned large number of registers. To accommodate the high level of instruction parallelism, there are multiple replicated functional units for integer and floating point calculations. These work to keep the microprocessor instruction pipeline as full as possible which, in turn, speeds business intelligence application performance.

Predication and Speculation

Predication and speculation are two of the key features of the IA-64 architecture that will enhance business intelligence application performance. Implemented in hardware, they work with the compiler to provide the most efficient instruction utilization. The predication feature allows the conditional execution of instructions before it is known whether the execution is needed.

Predication reduces the number of branches a business intelligence application must execute since branch control-dependent code is replaced by parallel scheduled and executed code. Since predication allows more code to be executed in parallel, the performance penalty of branch-dependent code and branch mispredicts is decreased. Business intelligence applications with varying or non-predictable workloads or queries, such as data mining applications, and the subsequent heavy branching, will benefit most from the predication feature of the IA-64 architecture.

Since processor speeds are far in excess of memory access speeds, pre-fetching code or data before conditional branch instructions need to be executed (control speculation) overcomes a major reason why the instruction pipelines of current microprocessors are not full. As is the case with predication, typically,

business intelligence applications with heavy branching (in contrast to well-ordered, repetitive technical calculations) will benefit most from control speculation.

Control speculation allows certain load instructions to be scheduled before conditional branch instructions, rather than after. In most existing architectures, load instructions typically cannot be moved ahead of conditional branch instructions due to the possibility that the load instruction may not be executed at all after the branch.

Variations on control speculation may be found in other microprocessor architectures. However, the microprocessor resource cost is often too high. In contrast, the number, variety and flexibility of the IA-64 registers (i.e., rotatable and framed), the large number of functional units found in the IA-64 architecture, and the power of the control speculation algorithms used by the IA-64 architecture (which enables the compiler to decide what to do in each speculation case), will allow previously unrealized business intelligence application performance.

The IA-64 architecture also provides data speculation. It is similar to control speculation but allows loads to be scheduled above stores wherever possible to reduce CPU wait states generated by branch-intensive code - also speeding business intelligence application performance.

Finally, the IA-64 architecture provides a prefetch capability that the compiler will automatically use. Instruction prefetches can be signaled on branch instructions. Data can be prefetched with explicit prefetch instructions. Here too, microprocessor wait states are reduced and business intelligence applications will experience performance gains.

Summary

The growth of e-Business solutions has allowed businesses to collect rapidly growing amounts of data on their customers' behaviors, needs and interests. Business intelligence applications are increasingly critical for corporations who need to transform this mass of data into insight and action. The challenge for business intelligence application developers and users is to select computing platforms that offer the best performance, head room, scalability, availability and price.

The Intel Itanium processor, the first processor in Intel's IA-64 architecture family, offers the 64-bit addressing capability, large number of registers, inherent instruction level parallelism, predication, control and data speculation, and prefetch capability needed to best accommodate these business intelligence applications.