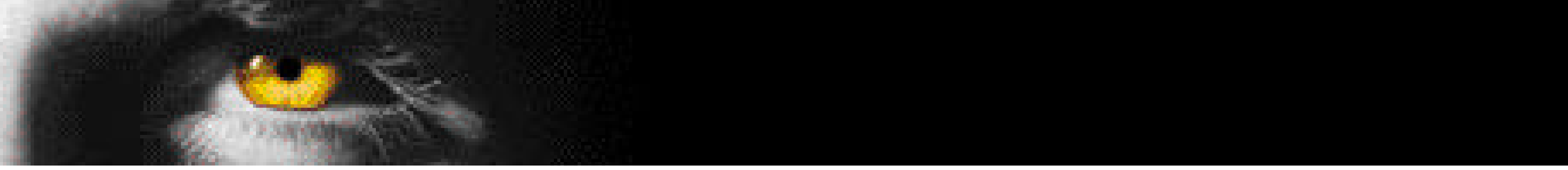


64 versus 32-bit processors: which benefits for numerical simulation?

e-Tips
Nr. 19

More and more, one hears speaking about 64-bit processors and it seems reasonable to ask the following question: which benefits this new technology can bring to the end user of numerical simulation software? The aim of this e-tip is to give a point of view on this issue.

Bits and bytes

Everyone knows that computers represent any information, whatever its nature is (text, numbers, sound, image ...), as a succession of 0 or 1. This smallest piece of information is called a bit, a short cut for binary digit. Imagine now that we have 3 bits at hand: what is the largest integer number that we can represent? Figure 1 gives the answer: we can represent numbers from 0 up to $2^3 - 1$ (7).

In the main memory of a computer (RAM), bits are grouped by byte. A byte is a set of eight successive bits. On a 32-bit processor, four bytes are the amount of memory used to represent integer numbers. This means that the range for positive integers is from 0 up to around 4 billions (2^{32}-1).

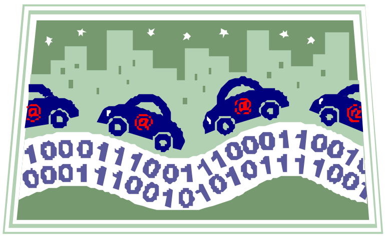
While the computer is in operation, information is continuously exchanged between the processor and the memory. In order to speed-up the execution, 32-bit processors are designed to exchange data with memory by bunch of four bytes at once: from a hardware point of view, this means that the data path between the processor and the memory is 32-bit wide (like a 32 lane motorway).

binary			decimal
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Figure 1: Binary representation of integer numbers with 3 bits (left) with their equivalent decimal value (right).

Memory size

Each byte in the memory is known to the processor through its address, in the same way an element in a vector is located by its index. Addresses being coded as integer numbers, the maximum amount of memory which can be addressed on 32-bit platforms is 4 billions

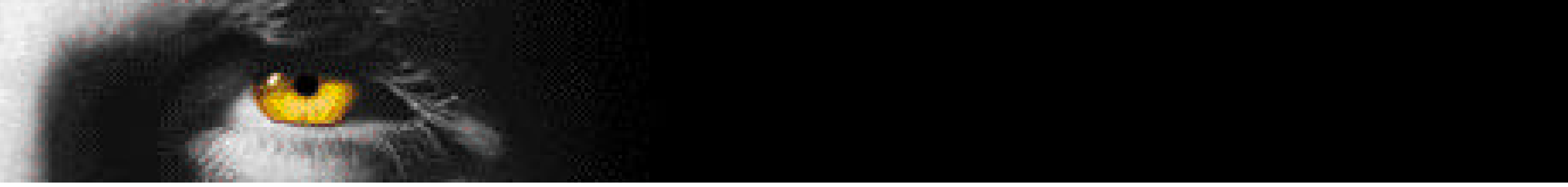


of bytes *(4 GB)*. Moreover, on some systems, only half of this value can be monopolized by a single process.
Today, the tremendous increase of the processor speed allows treating in acceptable computing times numerical models with larger and larger numbers of unknowns (such as for instance the numbers of nodes in a mesh) and it is henceforth not unusual to come up against this barrier of 2/4 GB.

In these cases, the migration to 64-bit processor will change the picture dramatically. A 64-bit processor can store integer numbers with 64 bits – which means it can express numbers or addresses from 0 to 18 × 10^{18} – and is therefore capable to theoretically address a colossal amount of 18 billion gigabytes of virtual memory. In practice, it is worth noting that, depending on the processor model and manufacturer, the physical address space is less than 64 bits wide (in fact, 40 to 48 bits) and give access to a real physical memory of 1 to 280 terabytes (which is already not so bad!).

File size

This limitation to 2 GB stands also for the size of the files which can be created on a 32-bit platform, the reason being the same: the offset which allows accessing data at the right place in a file is a signed integer whose value is less than 2 billions.



This limit is reached quickly when one wants to store the results of a calculation involving a large number of unknowns and a lot of time-steps, even if the results are split on several files. However, on many 32-bit platforms, large files (greater than 2 GB) are already supported providing that some special input/output functions are implemented in the source code of the application. If such is the case, the move to 64 bits will have no effect.

Precision and speed

On a 32-bit processor, calculations requiring a high level of precision are already done with double precision floating point numbers. These numbers being 64 bit wide, they have to be split to go through the data path bottleneck – an operation which adds an extra cost in terms of speed. In contrast, a 64-bit processor can execute instructions on 64-bit numbers in a single clock cycle. For such calculations, the results will have the same precision on both platforms, but will be obtained faster on a 64-bit processor.

	Pentium Xeon @ 3.2GHz (32 bits)	Itanium 2 @ 1.5 GHz (64 bits)
CINT2000	1306	1077
CFP2000	1220	2055

Table 1: Performance comparison between a well-known 32-bit processor and its 64-bit successor.

The width of the data path is obviously not the unique factor that influences the computing speed of a processor. It is nevertheless interesting to compare the performances as reported by the Standard Performance Evaluation Corporation (www.spec.org) for two processors sold by the same manufacturer (Intel): the Pentium Xeon @ 3.2 GHz (32 bits) and the Itanium 2 @ 1.5 GHZ (64 bits).

Table 1 gives two values for each processor: the CINT2000 measures the performance for computations involving integer numbers whereas the CFP2000 refers to floating point computations (the greater the value, the better the performance). It can be observed that both processors have almost equivalent performances for integer calculations. This is not the case for the floating point numbers wheras the Itanium processor (64-bit) is already 70% faster than the Pentium (32-bit). This is a promising result for people running numerical simulations.

To sum up, the main benefits 64-bit computing will bring to numerical simulation are the following:

- Push central memory above the limit of 4GB,
- Opens the perspective for new processors with improved performance.

calcom

Calcom ESI SA
Parc Scientifique EPFL, CH-1015 Lausanne, Switzerland
phone: +41 21 693 2918 / fax: +41 21 693 4740
mail@calcom.ch / www.calcom.ch / www.esi-group.com