3.06 GHz Pentium 4 and HyperThreading

By Johan De Gelas – November 2002

Improving TLP with HyperThreading

While the top of the line 3.06 GHz Pentium 4 is far too expensive for most hardware enthusiasts, it is one of the most anticipated Intel processors ever. The new flagship CPU from Intel is the first desktop processor to implement Simultaneous Multi-Threading or SMT. There have been quite a few papers written on the subject, including one from the University of Washington dating back to 1995. Later, in 1997, a very informative article on the subject, Simultaneous Multithreading: A Platform for Next Generation Processors was written by researchers from both DEC and the University of Washington, including Joel Elmer, a chief architect working at the time in DEC/Compaq's Alpha Processor Division. The Alpha 21464 was expected to be the first Alpha processor to implement this revolutionary technology, but those plans have since been laid to rest. While one might expect to see a high-end RISC architecture like the 21464 to lead the technological charge and be the first to introduce SMT, we are instead seeing this technology on a desktop CPU -- the Pentium 4.

Here at Ace's hardware we always have been a bit CPU centric, and back in December 2000, we found out that there was something called Jackson technology inside the Pentium 4 which was not enabled. At that time, we could not get any confirmation from Intel. But at IDF Fall 2001, Intel unveiled jackson technology, officially known as "Hyperthreading." Like out-of-order execution, Hyperthreading or SMT is a means to increase overall performance per clockcycle. Simply put, the goal is to maximize execution resources by executing instruction streams from multiple threads simultaneously, thereby exploiting parallelism at the thread-level rather than just the instruction-level.

Tests with the first revisions of the 2.4 GHz Pentium 4 Xeon revealed that Hyperthreading could push performance in server applications about 0-30% higher, and workstation applications typically get a -10 to +20% boost. So the first version of Hyperthreading could even reduce performance and therefore most Xeon workstations where shipped with Hyperthreading disabled. New revisions of, and gradual improvements on the Pentium 4 architecture made however sure that better Hyperthreading performance got better and better. Today, Intel feels that Hyperthreading performance cannot only improve the performance of workstations and servers but also the desktop.

SMT: Benefits and Disadvantages

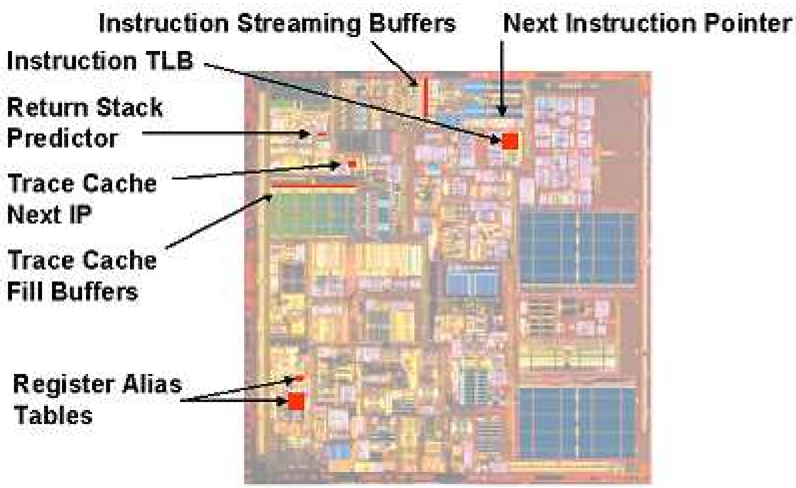
Ace's Hardware is read by many seasoned hardware veterans, but for the rest of you let me quickly introduce the concept of SMT. We will describe and discuss it in more detail in future technical articles. SMT is originally an idea of Dean Tullsen of the University of Washington, who described SMT back in 1995.

As we discussed in the future of x86 performance, few software applications can really use the potential of the current superscalar CPUs. A relatively easy example is the Athlon. The Athlon has 3 fully functional decoders and execution units for each type of instruction (integer, FPU, memory operation). So therefore, it should easily be able to decode, schedule, execute and retire slightly less than 3 x86 instructions per clockcycle on average. Code analyses shows that most applications are running at rates of less or slightly more than one single x86 instruction per clockcycle. This is clearly much less than the Athlon's superscalar architecture can sustain. Executing up to 3 x86 instructions happens only at brief peak moments, and therefore a lot of execution slots go to waste.

I used the example of the Athlon, as it is harder to make the same calculation for the Pentium 4, which can sustain a execution rate of 3 micro ops per clockcycle. The concept of the trace cache makes it much harder to calculate the number of sustained x86 instructions, but it is clear that the Pentium 4 is not running at its full potential either.

Enter Simultaneous Multi Threading or Hyperthreading as Intel likes to call it. The idea is to execute two threads, or "tasks," at the same time on the same processor (read more about threads and processes here). Normally a CPU will execute one thread, switch context (save the contents of the registers and CPU state in the cache), and then load the registers for another thread and execute it. A SMT processor is able to let two (or more) threads use its decoders, execution units and caches. The main objective is that a second thread would use the execution units that one thread cannot use at the moment and vice versa. This way, the processor should be able to sustain more instructions per clockcycle on average.

A dual processor system can of course execute two threads too, but the big advantage of SMT that it is very cost effective from a die size view. As you can see below, only a few additions to the die must be done. You need for example two instruction pointers ("Program Counters") of course as these registers point to the next instruction that must be executed, more data registers etc.



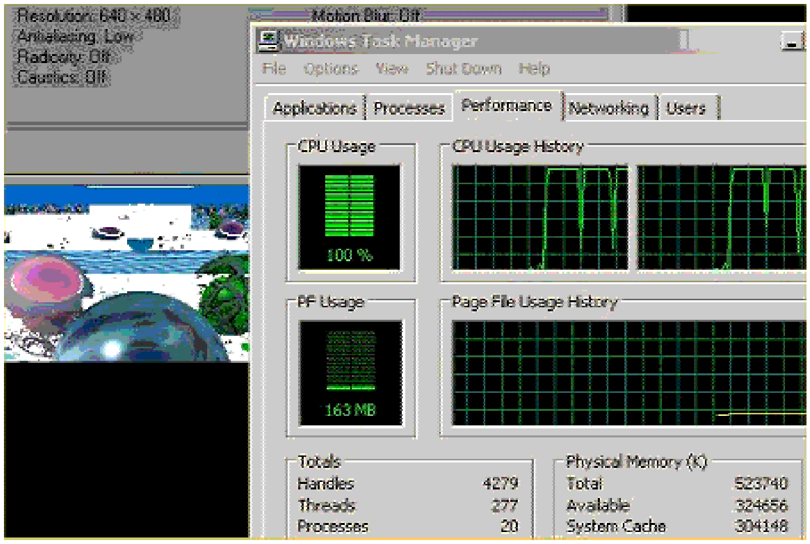
From a software perspective, an SMT-enabled processor appears as multiple logical processors on which threads can be scheduled. It's important to realize, however, that even though the there may be two logical processors, execution resources remain the same. Therefore, while Hyperthreading can maximize CPU resources and improve overall performance/clockcycle by exploiting thread level parallelism, it will not double performance.

Also important is the fact that the trace cache contains a special "thread" tag. This way, threads will not kick each other out of L1-cache. While SMT is a very cool way to increase the efficiency of a processor, it comes with a certain amount of trade offs. Caches will be shared among threads, while a real dual CPU system will have a cache for each thread.

A SMT processor can only be very efficient if the two threads do not compete for the same resources. It will be the most effective when one thread contains a lot of branches, and the other not, so when a branch misprediction occurs, the second thread can use the unused excution resources. Or one thread has a lot of memory accesses and the other not. Or one thread consists of mostly integer code, and the other of FP code.

To make sure that you get good Hyperthreading performance you need a chipset that is Hyperthreading enabled. As a SMT processor will present itself to the outside world as two logical processors, the chipset must be able to send IRQ request to both logical processors (two local APIC's). Intel told us that every (Intel) 533 MHz FSB capable chipset should be Hyperthreading-enabled. You just need to update the BIOS.

Next you need an operating system, which is optimized for Hyperthreading such as Windows XP and the latest 2.4 Linux kernel. If all these conditions are met, the operating system will see two processors.



Power Dissipation

A small disadvantage of more efficiently using the processor's execution resources is the fact that the CPU will dissipate more heat. Take a look at the table below:

Processor Model	Frequency (MHz)	Nominal Voltage	Typical Thermal Power (130 nm "Thoroughbred")	Maximum Thermal Power (130 nm "Thoroughbred)"
Athlon XP 1700+ (Palomino)	**1467**	1.75V	57.4 W	64 W
Athlon XP 2100+ (Palomino)	**1733**	1.75V	64.3 W	72 W
Athlon XP 1700+(T-bred)	**1467**	1.50V	44.9W	49.4W
Athlon XP 1800+	**1533**		46.3W	59.2 W
Athlon XP 1900+	**1600**		47.7W	60.7 W
Athlon XP 2000+	**1667**	1.60V	54.7W	60.3W
Athlon XP 2100+	**1733**		56.4W	64.3 W
Athlon XP 2200+	**1800**	1.65V	61.7W	67.9W
Athlon XP 2600+	**2133**	1.65V	62 W	68.3W
Athlon XP 2800+	**2250**	1.65V	62 W	68 W
Pentium 4 2.0 GHz (0.18 micron)	**2000**	1.7V	72 W	92 W
Pentium 4 2.0 GHz (0.13 micron)	**2000**	1.5V	52.4 W	66 W
Pentium 4 2.2 GHz	**2200**	1.5v	55.1 W	70 W
Pentium 4 2.8 GHz	**2800**	1.525V	68.4 W	85 W
Pentium 4 3.06 GHz	**3060**	1.55 V	81 W	+/- 105 W

As far as I know, no x86 CPU has dissipated so much heat as the 3.06 GHz Pentium 4 with Hyperthreading. The maximum power dissipation will never occur in the real world, as this can only be reached with special "thermal viruses." Luckily, the Pentium 4 has a very efficient heatspreader and a good clamping mechanism. We could not measure temperatures higher than 57°C with a decent cooler.

Nevertheless, the enormous amount of electrical current that the 3 GHz P4 requires will bring many motherboards and PSUs to their knees. This is probably the reason why we won't see the 3.2 GHz version arriving soon, as Intel will lower the power dissipation of its cores before moving to higher speeds. But right now, the current 3 GHz parts require 1.55V core voltage, and it is the most demanding processor of all. Intel has therefore reduced the recommended maximum processor ambient temperature from 45°C to 42°C and recommends 4 phase voltage regulators.

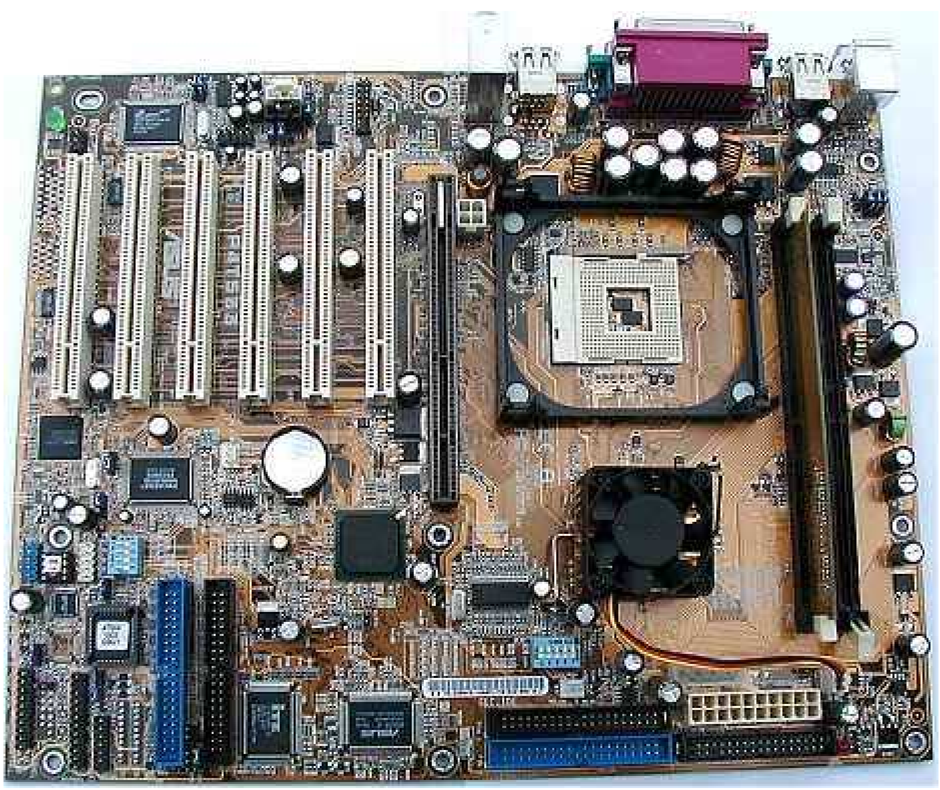
The Motherboards

P4T533

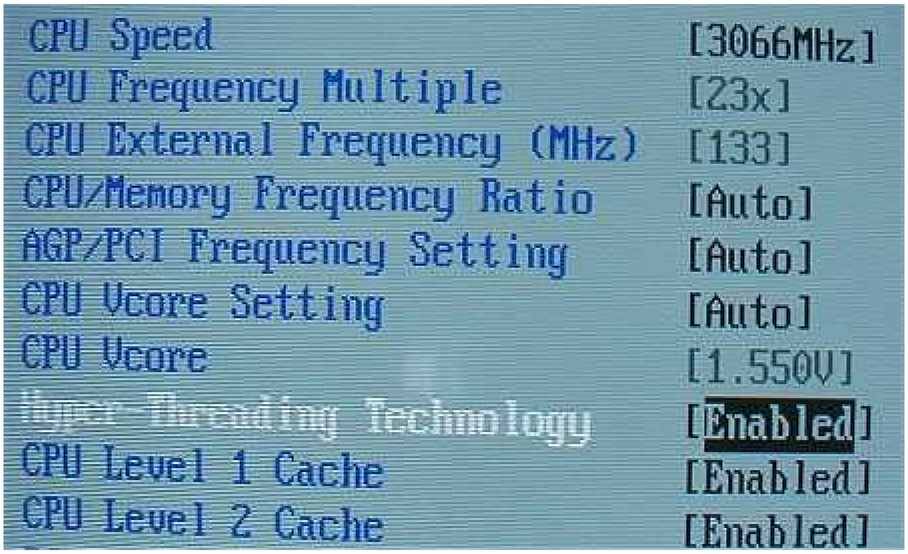
We tested the new 3.06 GHz Pentium 4 on two boards for now. We will provide more info in the next article. When we first tested the P4T533, we were pleasantly surprised. Our 2.4 GHz P4 overclocked to 2.7 GHz, with the 32-bit RDRAM running at no less than 600 MHz DDR (1200 MHz). The system was not fully stable, but we were probably pushing the limits of our Pentium 4. However, when we started testing with the 2.8 GHz Pentium 4, we experienced many lock ups when playing games. In fact, every game would lock up after a few minutes.

The RDRAM channels were not the problem. Stability was not perfect, but at 155 MHz x 16, the system ran most benchmarks. After trying out several BIOSs, we still faced the same problems. When we paid closer attention to the voltages measured, it became clear that the ASUS P4T533 power circuitry was not up to the job: the voltages (CPU core etc.) fluctuated too much. We tested several boards from several shops, and the results were all more or less the same.

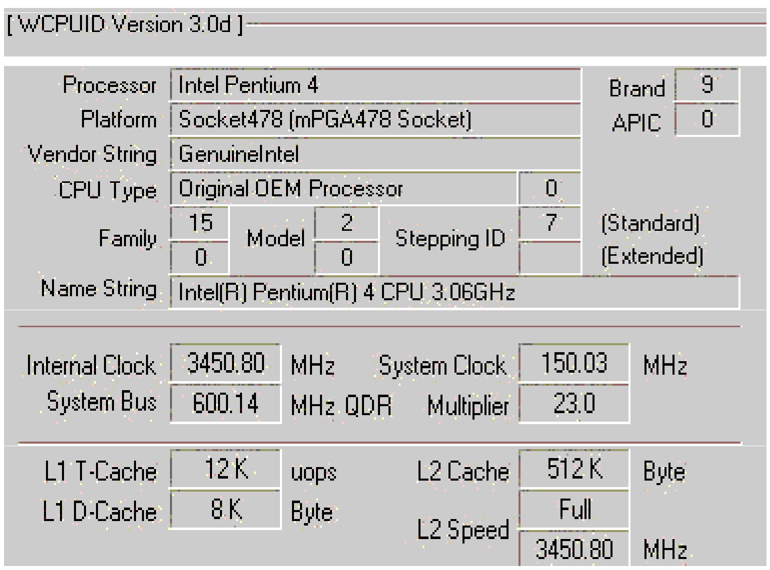
ASUS has, however, produced a new version of the P4T533, which comes with a very good power circuitry. Based on the data we have now, the new boards are still marked as revision 1.03.



However the "fixed power circuitry boards" can be recognized as the northbridge now features a fan, while the older boards did not.



With the BIOS upgraded to 1.005, the ASUS P4T533 support Hyperthreading. The 1066 MHz (533 MHz DDR) 32-bit RDRAM make the ASUS P4T533 the fastest performing Pentium 4 board on earth.



We'll show you more evidence in the next article. The ASUS P4T533 was able to push the P4 3.06 GHz to 3.45 GHz (1.75V) stable, which is very impressive.

MSI i845PE

The ASUS P4T533 is - if you make sure you get the fixed board - a screamer, but rather expensive. If you like something more affordable, the MSI i845PE MAX2 might be the right alternative. The board supported the 3.06 GHz Pentium 4 and Hyperthreading very well.

The board comes with CMEDIA sound onboard, an ATA-133 Fasttrack promise controller and 5 immediately accessible USB slots.

We will discuss the board and DDR333 memory performance in more detail in the next article.

Benchmarked Configurations

All systems were tested with NVIDIA's Detonator 30.82 drivers. The desktop was set at a resolution of 1024x768x32bpp with an 85 Hz refresh rate. V-sync was off at all times.

We used Corsair's XMS 3200 CAS 2 DDR (DDR400) for maximum overclocking possibilities and stability. We will discuss DDR400 in more detail in the next review. In this review, the XMS32000 were running at 166 MHz DDR (333 MHz) and configured in all systems to work with a CAS latency of 2.

Systems

Athlon XP 2000+, Athlon XP 2200+ , Athlon XP 2600+, Athlon XP 2800+

- **ASUS A7V333 BIOS version 1.013**
- **512 MB Corsair PC3200 XMS (DDR-SDRAM) running at 333 MHz CAS 2 (2-3-3-6)**

Pentium 4 (Northwood) 2 GHz (400 MHz FSB), 2.4 GHz (400 MHz FSB), 2.4 GHz (533 MHz FSB), 2.53 GHz (533 MHz FSB), 2.8 GHz and 3.06 GHz

- **DDR platform:**
- **MSI 845PE MAX2 (i845PE chipset) BIOS version 1.4**
- **512 MB Corsair PC3200 XMS (DDR-SDRAM) running at 333 MHz CAS 2 (2-3-3-6)**
- **RDRAM platform:**
- **ASUS P4T 533 BIOS Version 1.005 with 512 MB RIMM4200 at 1066 MHz**
- **512 MB Samsung RIMM4200 - 32**

Shared Components

- **Seagate Barracuda ATA III Model ST320414A (7200 rpm, ATA-100)**
- **ASUS Geforce Ti4400 128 MB**
- **AT 2700 10/100 Mbit NIC**
- **Sound Blaster Live!**

Software

- **Via 4 in 1 Drivers 4.42**
- **Intel chipset inf update 4.09.1011**
- **Windows XP Service Pack 1**
- **DirectX 8.1**

We'd like to thank the following helpful people for their support and important contributions to this review:

- Kristof Semhke, George Alfs and Jurgen Eymberts (Intel) made sure we could test the Pentium 4 2.8 GHz and 3.06 GHz HT.
- Thanks to Will Teng, Carol Chang (ASUS) and Sharon Tan (BAS computers Netherlands) for the ASUS A7V333 and ASUS P4T533.
- Saskia Verhappen of MSI provided us with the MSI i845PE
- Damon Muzny (AMD) made sure we were able to test the Athlon XP 2800+, 2600+ and Athlon XP 2200+ Thoroughbred.
- Robert Pearce of Corsair, provided us with Corsair's PC3200 CAS2 XMS.

Let us see some benchmarks!

Multitasking Benchmarks

At the keynotes at the IDF and during the 3.06 GHz NDA briefings, Intel stressed more than once that especially multitasking scenarios would be accelerated by Hyperthreading: . Brian reported the following demos:

> "Hyperthreading was a big topic for Intel and there were several demos of Hyperthreading on 3 GHz Pentium 4 CPUs. The demos during the keynote consisted of two programs running concurrently and illustrated the performance improvement associated with Hyperthreading in these kinds of situations. Not all of the situations were what you might consider common, but they did demonstrate an improvement nevertheless.
>
> The demos compared two systems, one with Hyperthreading and one without. One of the demos consisted of an excel macro that was running while Outlook was archiving email. Another was a 3D game that was running while video was being captured in the background from a TV broadcast. No precise performance details were indicated, rather the demos served to indicate a visual performance difference to those in attendance. In the case of the 3D game and video capture demo, the capture running on the non-HyperThreaded Pentium 4 system was dropping quite a few frames, while the capture on the HT-enabled system ran much more smoothly. In both cases, the 3D game ran at full speed, so the performance improvement with HT was seen in the performance of the background task. There were additional demos, including one involving video capture with on-the-fly MPEG4 compression, but as before, no hard performance figures were available."

We were and still are skeptical about these kind of benchmarks. Hardcore gamers are not going play anything in the background if it hurts their framerates, even if Hyperthreading somewhat negates the performance hit of running a background application. Productive office users schedule an Outlook archiving task when they are not using their desktop. And on-the-fly MPEG4 compression may probably only be possible with lower quality settings, only one pass, without 6 channel sound, etc.

So instead of recreating Intel's benchmarks, we turned to our readers at the General Message Board. We did not only want to test real world situations, but also situations where higher multi-tasking performance really matters. You told us about three important multi-tasking scenarios:

1. You run music decoders and/or compress files in background, while you play games.
2. You like to do something else while you are ripping videofiles from DVD (MPEG -2 decoding)
3. You like to do something else while you are encoding videofiles from MPEG -2 to MPEG-4 (Divx 5.02 encoding).

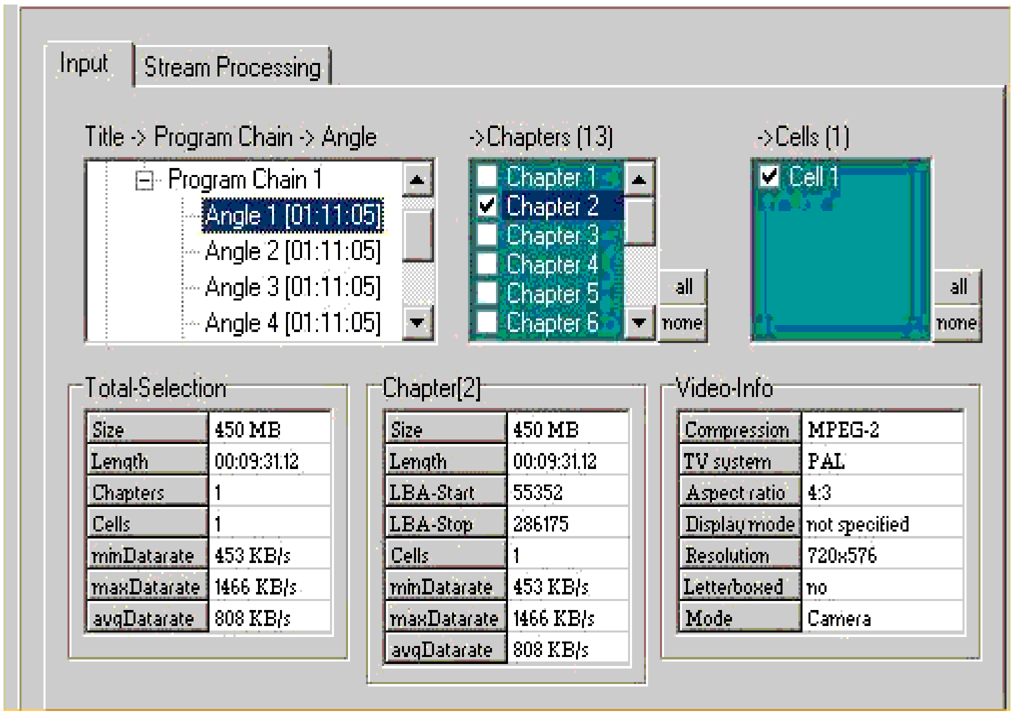
We decided to simulate these situations in our lab. Our first "multi-tasking experiment" consisted of starting WinAmp, playing some music ("Lords of the Rings Theme song") , then rarring a 340 MB MPEG-4 file, immediately followed by launching the Comanche benchmark. Comanche contains SSE-2 optimizations and runs very well on the Pentium 4, but our main objective is to see how much the framerates decreases - percentage wise - due to all the background tasks. We are less interested in the absolute numbers. In the following tests, all Pentium 4's are benchmarked on the 32 bit RDRAM platform and the Athlon XP 2800+ on the KT333 platform with DDR333.

MPEG-2 decoding/Ripping	Pentium 4 3.06 GHz Hyper Enabled	Pentium 4 3.06 GHz Hyper Disabled	Athlon 2800+	Hyperthreading Speedup
Only Comanche 4 (FPS)	**61.9**	60.3	48.4	+2.6%
fps Comanche: (Rar 340 MB file + Winamp + Comanche in foreground)	**27.83**	25.9	20	+7.1
Time in seconds to Rar 340 MB of MPEG4 file: (Rar 340 MB file + Windows Media Player + Comanche in foreground)	**560**	600	900	+7.5

The performance boost that Hyperthreading provides is relatively modest but still significant. The Hyperthreaded Pentium 4 is able to achieve 45% of the normal performance, while the non-HT 3.06 GHz only reaches 43% of its normal performance under heavy multi-tasking load. While that might not seem important, the WINRAR background task takes 7.5% less time.

So how did it "feel" to enable Hyperthreading? With Hyperthreading on, the music stuttered once. The same thing happened with Hyperthreading off. The biggest difference was the speed at which the Comanche benchmark started. With Hyperthreading off, it took about 10 to 12 seconds before the benchmark started, with Hyperthreading on, it took about 4 to 6 seconds. So response time gets a bit better. On the Athlon, performance was pretty mediocre, but the music did not stutter. This might have been a coincidence of course, considering that the Athlon performed poorly in WinRAR, even when we did not launch Comanche and ran only WinAmp .

The second multi-tasking scenario involves ripping a 450 MB MPEG-2 file of a DVD with Smartrip in the background, while running the Comanche benchmark. The reason why we always ran Comanche on the foreground is the fact that this way, all three multi-tasking benchmarks proved to be very repeatable. All our benchmarks had an error margin of about 1 to 2%.



The DVD ripping process consists of searching for a key to hack the DVD encryption followed by MPEG-2 decoding. Both the Athlon and Pentium 4 were ripping at rates far below the maximum transfer rate of the DVD player. As a result the DVD player was not the bottleneck, nor was the hard disk.

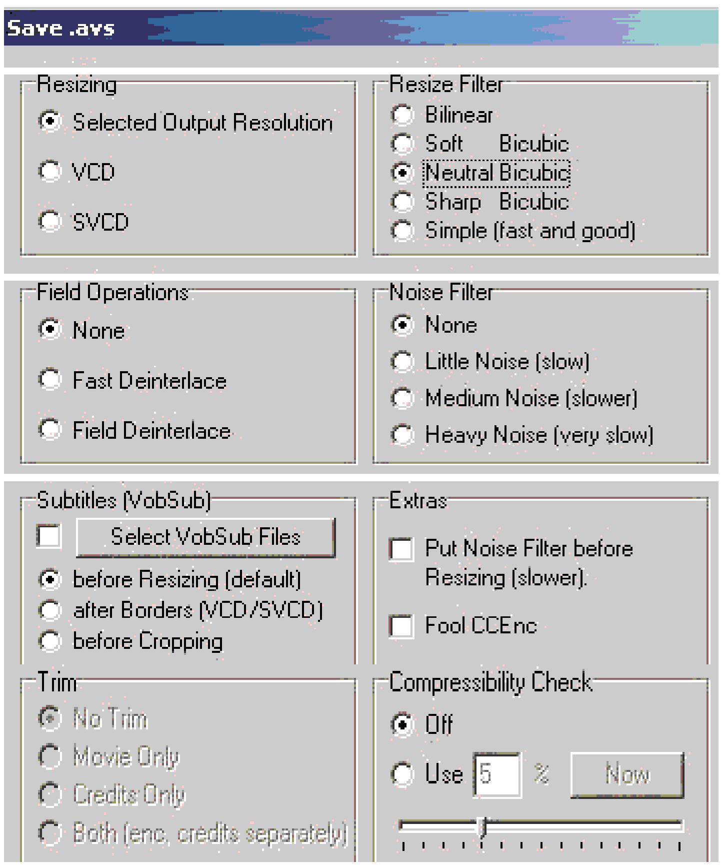
MPEG-2 decoding/Ripping	**Pentium 4 3.06 GHz Hyper Enabled**	**Pentium 4 3.06 GHz Hyper Disabled**	**Athlon 2800+**	**Hyperthreading Speedup**
Only Smartrip	**90**	93	82	+3%
DVD2AVI + Comanche in foreground	**92**	93	83	+1%
Only Comanche	**61.9**	60.3	48.4	+3%
DVD2AVI + Comanche in foreground	**53.2**	49.8	40.9	+7%

The Athlon XP looked pretty bad when compressing with WINRAR, but performs in our ripping test very well. Considering that the Pentium 4 runs 810 MHz faster, has a bigger and faster cache and a better memory subsystem, it is quite surprising that the Athlon still outperforms Intel's flagship processor.

I noticed that the first part of the ripping process, decrypting the DVD key was much faster on the Athlon. The used algorithm may be very similar to RC5 decrypting and other decrypting algorithms. These decrypting algorithms run much faster on the Athlon as it 3 superscalar ALUs, all capable of doing a rotate instruction with single-cycle latency. Given that rotates are the core of RC5 processing, and that the Pentium 4 has only two superscalar barrel shifters with higher latencies, I think we may speculate that is probably the reason why the Athlon handled the first part of the ripping process so much faster.

MPEG-2 decoding works also very well on the Athlon (Intervideo's WinDVD 4 was installed) and the it allows the Athlon to be able to keep the Comanche framerate average at 85% of its normal framerate. The Hyperthreaded Pentium 4 reaches 86% of its normal framerate, and 82% without the help of the second logical processor. This benchmark confirms again that the gains of a Hyperthreaded CPU are modest in multi tasking scenario's but significant.

In our last scenario, we compressed the 450 MB MPEG-2 file to an MPEG-4 file thanks to the DivX 5.02 codec, DVD2AVI and Virtual Dub which are all integrated in the superb software called [Gordian Knot 0.26](). Below you find the settings of DVD2AVI. Six Channel Sound was also encoded in a separate AC3 file.



The DIVX Pro 5.02 codec is optimized for SSE and SSE-2 and the program encoding with this CODEC, Virtual Dub 1.4.10 is multi threaded. The results were quite surprising.

MPEG-4 Encoding	Pentium 4 3.06 GHz Hyper Enabled	Pentium 4 3.06 GHz Hyper Disabled	Athlon 2800+	Hyperthreading Speedup	Difference
Only Virtual Dub, time:	**11:42**	14:04	16:07	+20 %	38%
Time Virtual Dub 1.4.10: (Virtual Dub 1.4.10 + Comanche)	**12:21**	15:04	17:14	+22%	40%
Only Comanche, Comanche framerate:	**61.9**	60.3	48.4	+3%	+28%
Comanche framerate: (Virtual Dub 1.4.10 + Comanche)	**39.2**	56.9	47.1	-31%	-19%

The Pentium 4 HT enabled is by far the fastest MPEG-4 encoder but at a price: Comanche performs rather mediocre in foreground. I have been repeating this benchmark over and over again, and the results were remarkably repeatable: the comanche frame rate fluctuated between 39 - 39.7 fps, and the MPEG-4 encoding times only varied a few seconds.

As the MPEG4-encoding is multithreaded, it tries to use both logical processors, and this probably artificially increases the number of cache misses of the Comanche benchmark. The MPEG4-encoding is - so to say - pushing the Comanche benchmark away. As both other CPUs switch from one task to the other from time to time, they probably suffer less from this cache trashing problem.

Conclusion on Multitasking and HyperThreading

More research need to be done, but we will formulate a few temporary conclusions. Both video ripping and encoding contain chains of instructions which depend on each other ("Dependency chains") and have to go a lot to the memory. In other words, these kind of applications will not keep the CPU busy all the time, as the CPU is waiting a lot for instructions and data to arrive, or it can not execute other instructions in parallel as they are depend on the result of the currently executed instruction. So video ripping and encoding combined with another task (comanche in our example) is ideal for a hyperthreaded processor which is able to fill in the gaps (the open execution slots) with another application.

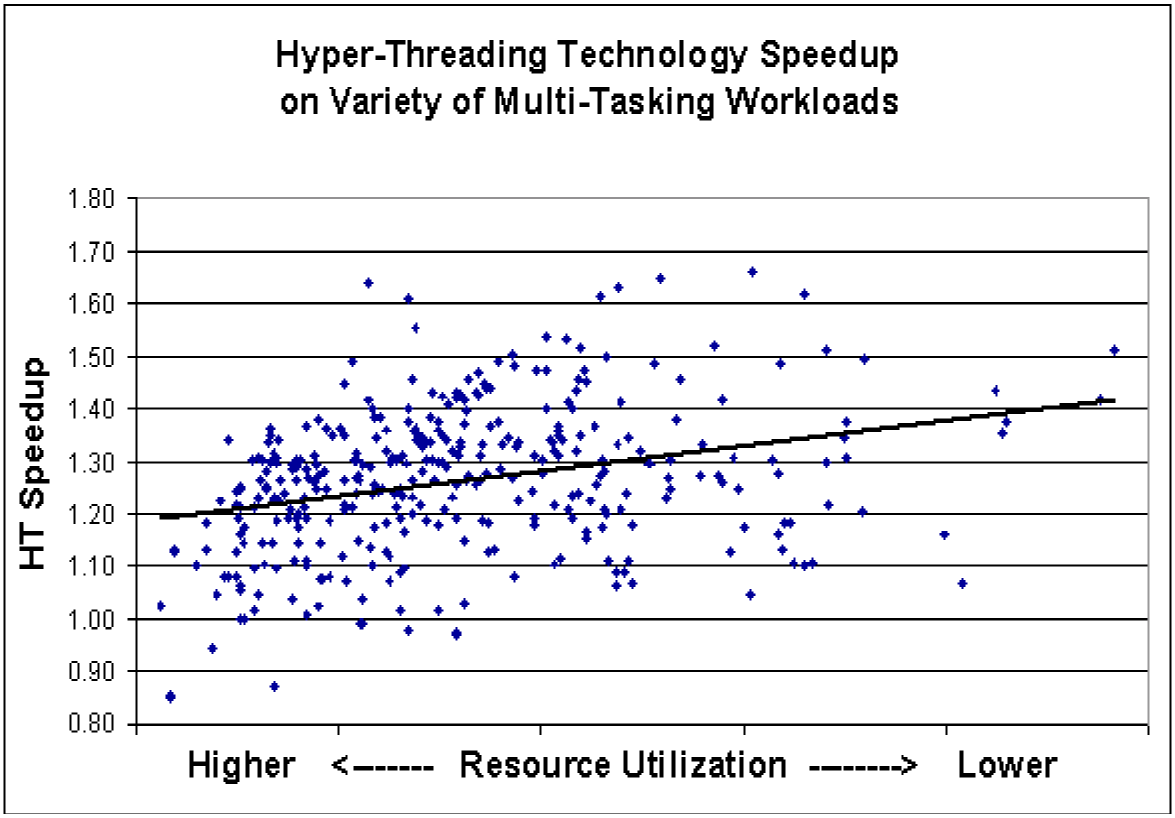
However, our results do not firmly confirm this. It seems that if the Hyperthreaded CPU can not reach its full potential if one of the applications in the multitasking scenario is multi threaded (Virtual Dub) and tries to keep both logical CPUs to itself.

To resume:

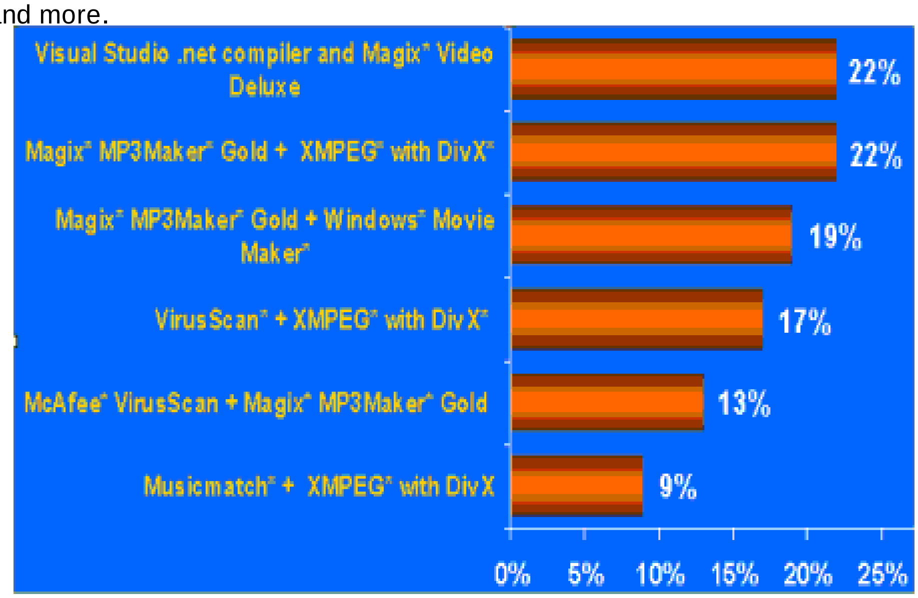
- Hyperthreading speeds up **single threaded applications a little bit** by handling the OS tasks in the background on the second logical CPU (See Comanche framerates for example: +2.6%)
- Hyperthreading speeds up **multiple single threaded applications quite a bit** (See Comanche (+7.1%) and RAR compressing results (+ 7.5%))
- Hyperthreading **speeds up multithreaded applications a lot** (Virtual Dub MPEG4 encoding + 20%)
- But seems to have a little trouble with a multi thread applications and some single threaded applications at the same time.

It is clear that while multi tasking can benefit quite a bit from Hyperthreading, this will vary a lot on the used applications. We will try to do as much relevant tests as we can and we advise you to ask yourself each time if this is the way you use your PC as the results vary greatly on the usage pattern.

Our finding



Notice how Hyperthreading seems to boost two intensive tasks (like Comanche and Virtual Dub, high resource utilization) with "only" 10 to 20%. Notice also that it gets more difficult for Intel to find scenarios with two lower resource utilizing applications. Intel's own examples do not feature these "exceptional" benchmarks, which show 35% better performance and more.



So based on our experiences and Intel's, we think it is reasonable to assume a 5 tot 15% speed up of multi-tasking on average. Let us investigate our third point, multi threaded applications a bit more.

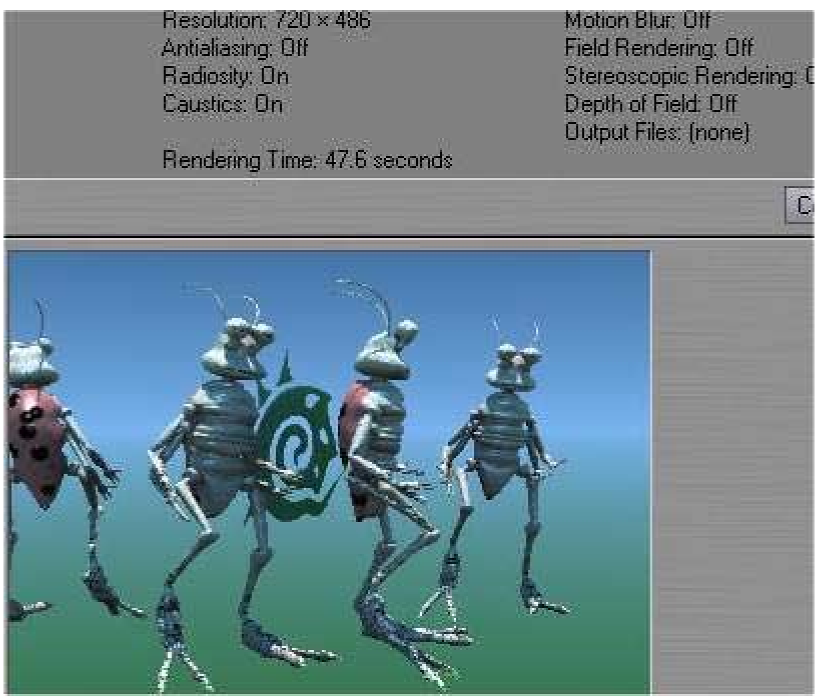
Multithreaded Applications: Lightwave

Does Newtek's product need much introduction? It is one of the big names in the 3D animation realm and it is very well optimized for the Pentium 4. As it one of the most popular packages, and rendering can be done in one to 8 threads, it is an excellent benchmark. We decided to benchmark the raytracing benchmark included in all Lightwave 6.0/7.0 and 7.5 packages. This way you can compare your configuration and results with ours. This article provides our rendering settings in the Render Options panel. We also tried out 1, 2, 4 and 8 thread rendering.

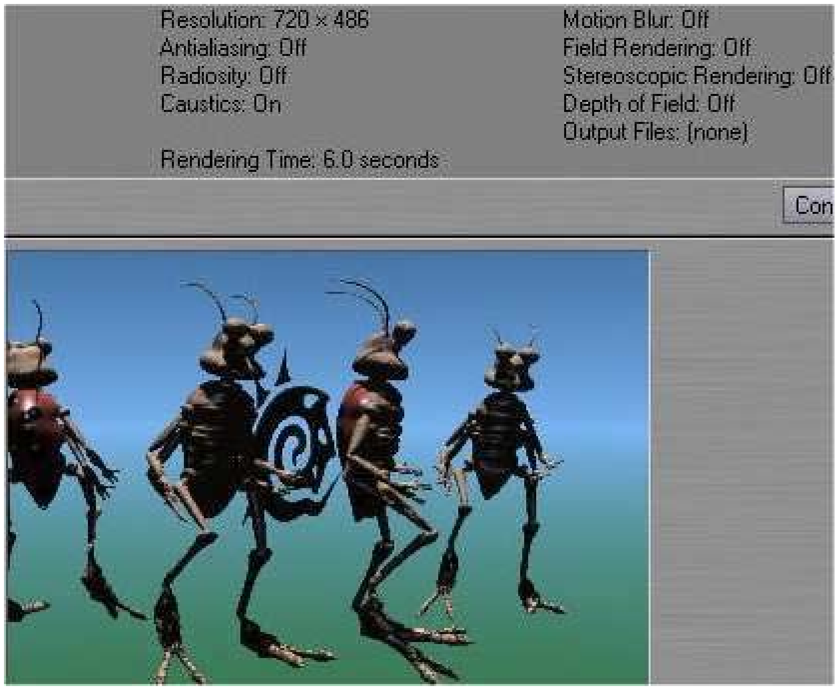
Raytracing standard lightwave benchmark	Pentium 4 3.06 GHz Hyper Enabled	Pentium 4 3.06 GHz Hyper Disabled	Athlon 2800+	Hyperthreading Speedup
1 Thread	**99.5**	98.2	134.5	-1%
2 Threads	**91.3**	105.9	141.5	+16%
4 Threads	**88.8**	109.3	141.4	+23%
8 Threads	**91.2**	106.3	141.2	+17%

Four thread rendering seem to be ideal for the Hyperthreaded Pentium 4, while all other CPUs prefer one single thread. Raytracing gets up to 23% faster thanks to Hyperthreading. While the Athlon was able to keep up with the Pentium 4 in raytracing tests with previous versions of Lightwave (7.0 and older), the new Lightwave 7.5 is even better optimized for the Pentium 4 and the Athlon XP 2800+ is up to 50% slower than Intel's 3 GHz showpiece.

Radiosity is a lighting technique which makes the lighting of a scene much more realistic but which results in extremely slow rendering times, even on the fastest setups. So people tend to enable Radiosity only in final stage of a project. The "SKULL_HEAD_NEWEST" benchmark is Newtek's and Intel showcase for SSE-2 optimization and includes Radiosity. However, following Intel Guidelines about benchmarking might make the picture maybe slightly too rosy for the Pentium 4. So I turned to Newtek's site and compiled a radiosity scene myself based on a example at Newtek's site about Radiosity and Animation. Below you see the scene with Radiosity enabled…



…and disabled:



Notice how Radiosity makes the scene very alive, but rendering times get up to 8 times (!) slower. We rendered with 4 Threads.

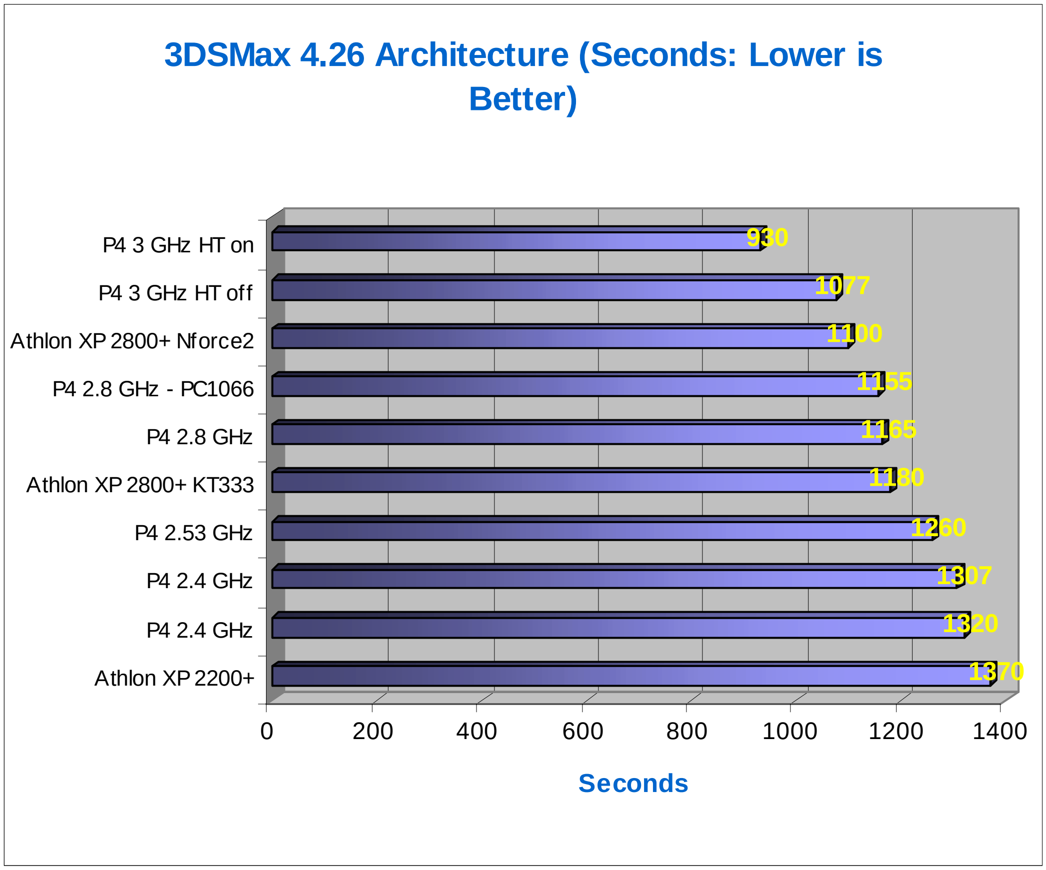
Grashopper Acebench	Pentium 4 3.06 GHz Hyper Enabled (seconds)	Pentium 4 3.06 GHz Hyper Disabled (seconds)	Athlon XP 2800+ (seconds)	Hyperthreading Speedup (%)
Radiosity on	**28.9**	28.4	47.4	-2%
Radiosity off	**4.4**	4.9	6.3	+11%

This time, Hyperthreading does not pay off. We got the best results with 1 thread on the both 3.06 GHz configurations (28.1 seconds). Hyperthreading benefits do not only vary from software to software, but also depend strongly on the filters you use and even the models, as we will see further.

Radiosity is extremely fast on the Pentium 4, up to 67% faster than the Athlon XP.

Multithreaded Applications: 3DSMax

We tested the architecture scene from the SPECapc 3DS MAX R4.2 benchmark. This test has a moving camera that shows a complicated building, a virtual tour of a scale model. This complex scene has no less than 600,000 polygons and 7 lights. It runs with raytracing and fog enabled. Frames 20 to 22 were rendered at 500x300 to the virtual frame buffer (memory).

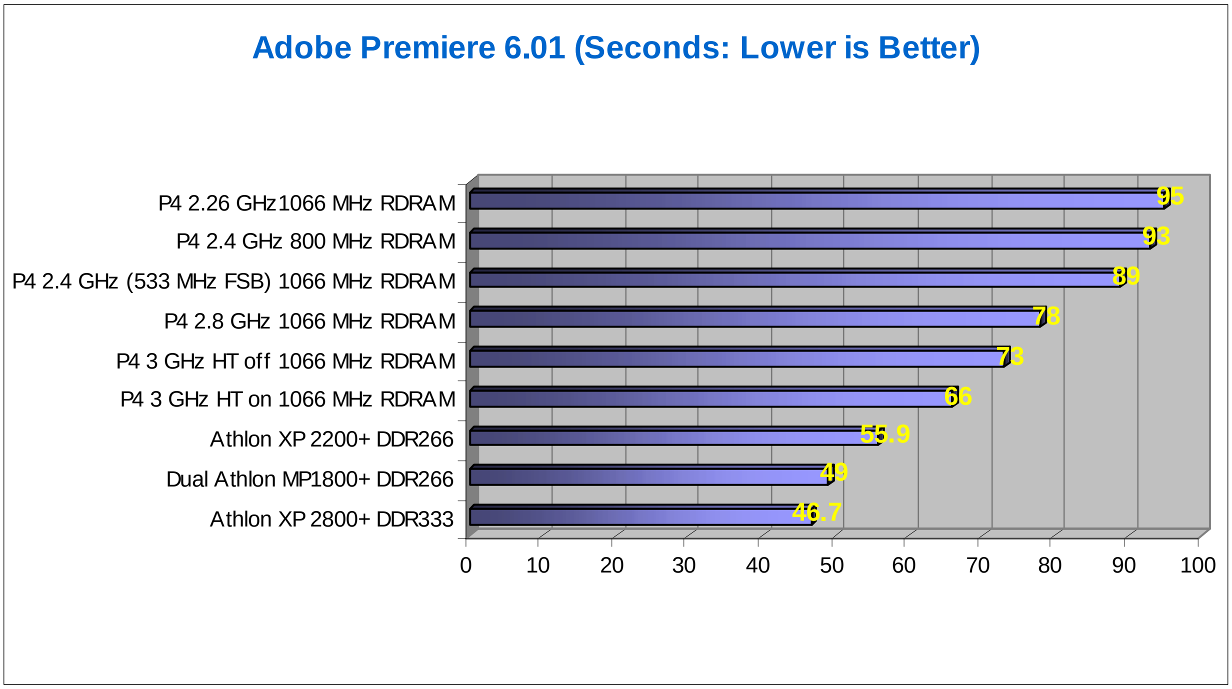


3DSMax performs 25% better thanks to the second logical processor. In this case, the 3 GHz HT processor is almost a Dual CPU, making it a very nice workstation processor.

Multithreaded Applications: Adobe Premiere

Digital cameras, PCI firewire cards, CPUs... they all get cheaper. Therefore the times that semi professionals and amateurs had to invest a lot of money in specialized analogue to digital videocards might come to an end. So we feel that software video editing rendering should make a good test. We included a few results that we borrowed from our upcoming video editing guide. More information later...

We used the sample project "Z-tour.ppj" of Adobe Premiere 6.01 and video settings were set at MS Digital Video (PAL 720x576), best quality (100%). Rendering times were slightly higher with the NSTC format, but the results were almost identical.



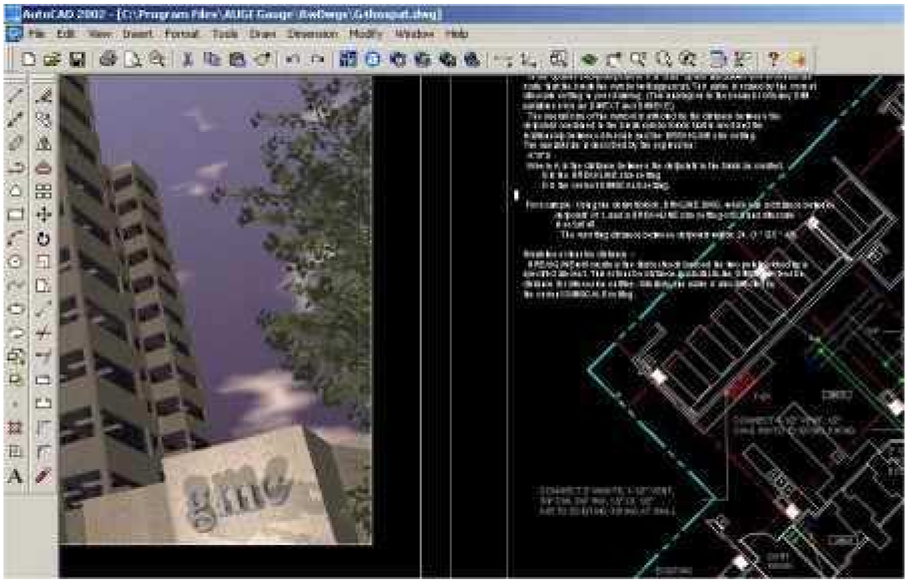
MPEG-4 encoding, Lightwave and compressing with WinRAR tests were landslide victories for the Pentium 4. This time, the Athlon fights back, and performs no less than 40% faster than the best Pentium 4. Hyperthreading helps with a 11% boost, but is not enough to beat the Athlon.

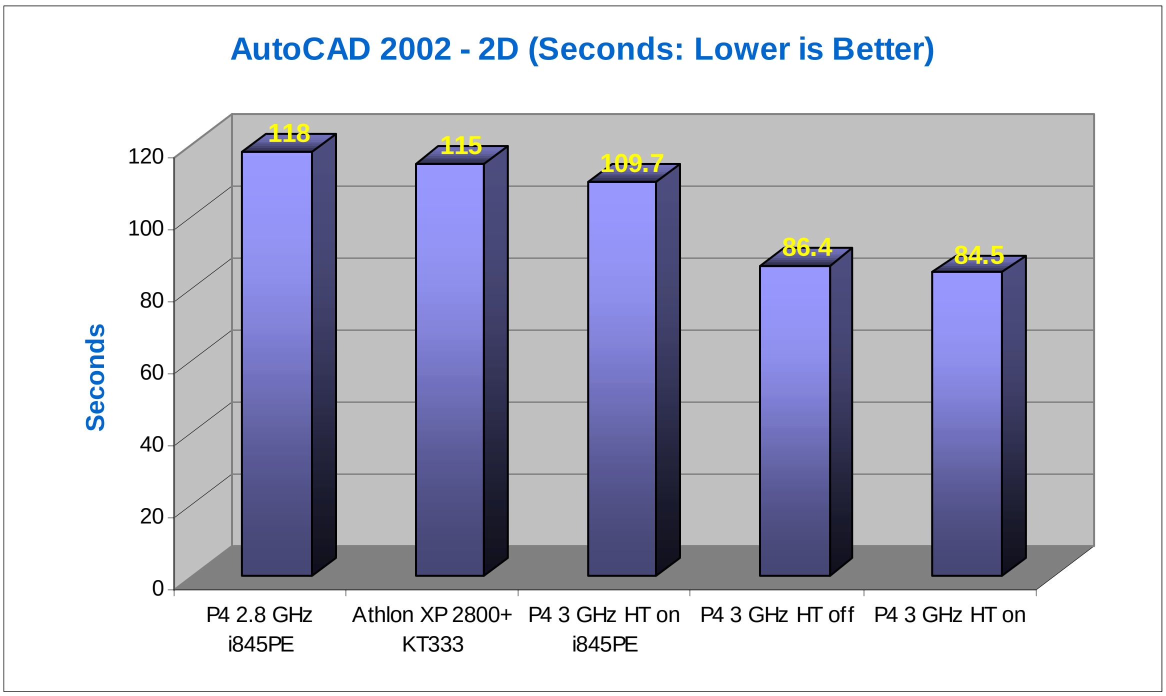
Multithreaded Applications: AutoCAD 2002

To benchmark AutoCAD, we used the AUGI Gauge benchmark from Autodesk Users Group International. From the AUGI Gauge site:

> The AUGI Gauge is a performance-testing tool that can be used to develop benchmark scripts for testing different operations and different drawings. The testing tool comprises a Visual Basic front end and an AutoLISP testing engine. The AUGI Gauge prints completion times for each test operation to a text file, which can be imported into a spreadsheet for data manipulation. The original AUGI Gauge testing tool was designed to work with AutoCAD Release 12 (DOS), Release 13 (Windows) and Release 14. The current version works with AutoCAD Release 14 and AutoCAD 2000.

The benchmark itself consists of two sections, an we have used the real-world test that performs various file, edit, and display operations (totaling 30) on a series of 15 drawings that each average 2 MB in size.



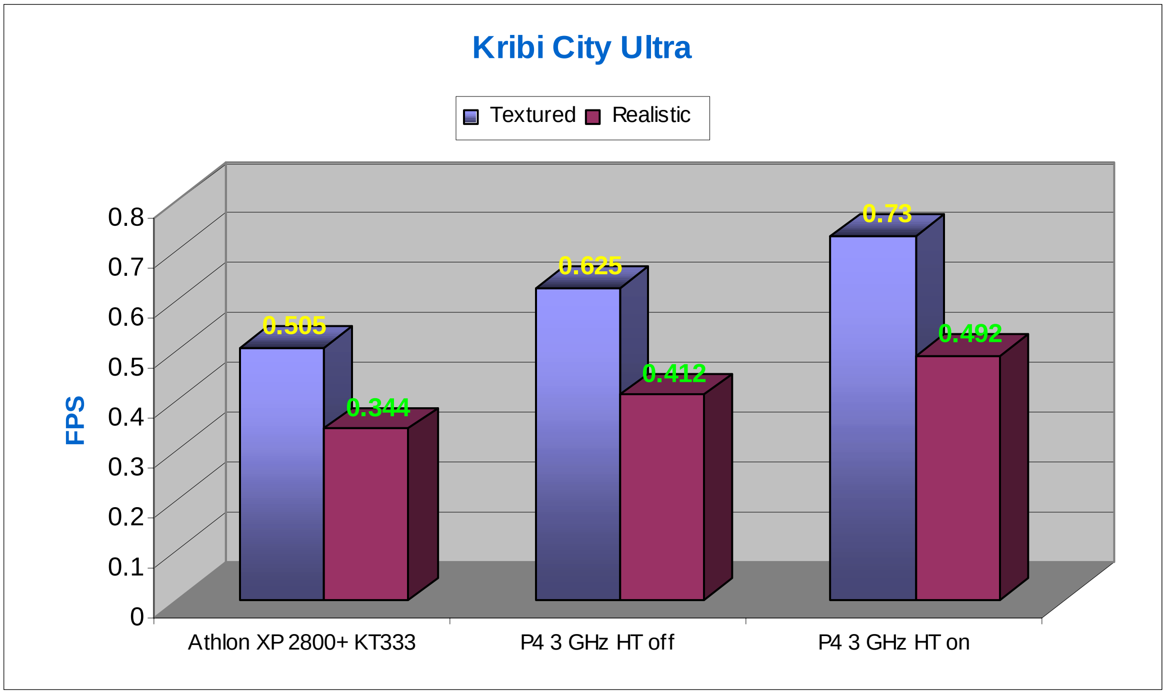


The Hyperthreading gain is quite modest, but the boost from RDRAM over DDR is spectacular. We were surprised by the huge 30% performance gain. The complex drawings have to be loaded from the memory of course.

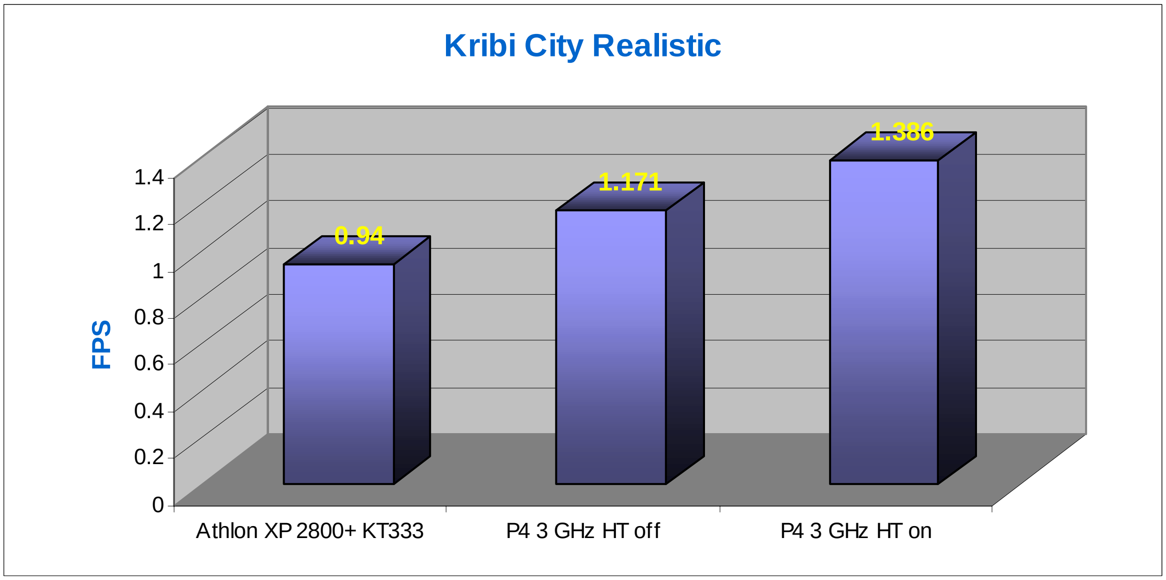
Multithreaded Applications: Kribi Software Rendering

Save yourself an OPENGL card is the philosophy of Kribi. Kribi, a product of Adept Development, is quite a an ultra powerful software rendering 3D engine. Originally developed by Eric Bron, a regular visitor of Ace's Hardware, is is designed to handle up to 10 billion polygons and enable real-time photorealistic rendering. The Kribi engine uses 100% software rendering (a pure CPU benchmark) and cannot work without SSE instructions. It is thus a sort of SSE and FPU benchmark. This time we used Kribi version 1.1 which is much faster, and carefully optimized for Hyperthreading.

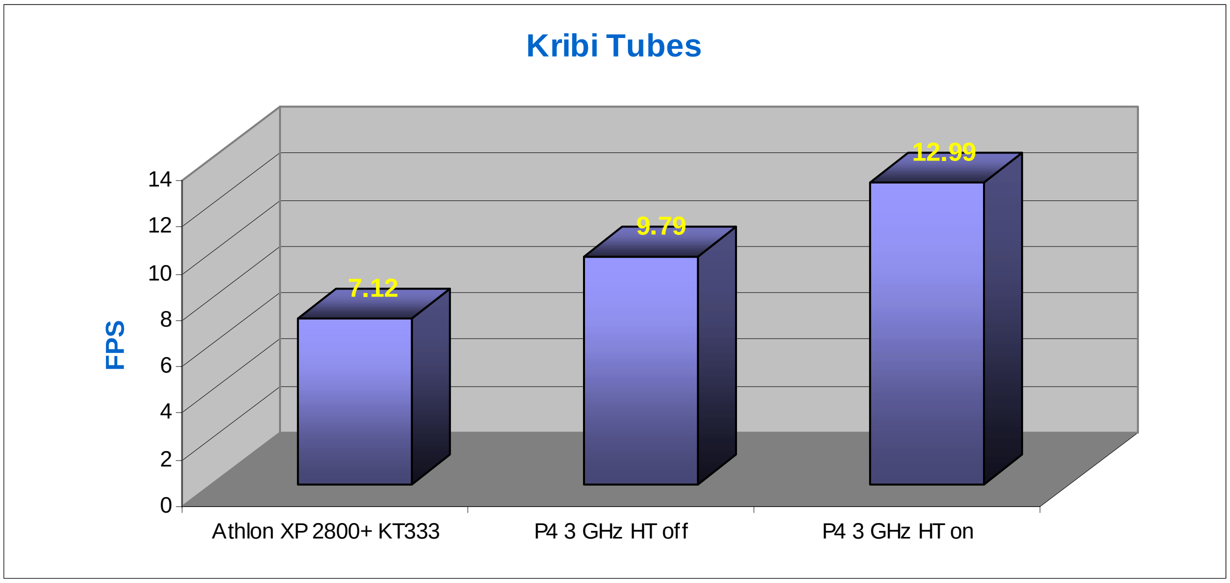
We tested with several models to evaluate whether or not the used model has a significant influence on performance. The first scene - City Ultra - is the most spectacular: no less than 16.7 billion polygons in total. All results are expressed in frames per second (fps)



Hyperthreading provides a 17% boost in texture mode and up to 19.4% better performance in Realistic mode. The next scene - city - contains about 107 million polygons.



Again, an 18% boost. Next up, a special scene with a lot of texture mapping, no less than 12 lights but only 29 k polygons.



This benchmark is spectacular: Hyperthreading boost performance no less than 32.6%! The picture below shows the model.

Eric Bron explains the exceptional boost that Hyperthreading provides:

"in the case of tubesmap.d, my explanation of the good H-T speedup is as follow : the scene is based on texture mapped cylinders, the memory access pattern for texture lookup is more random than with planar texture so the latency hiding effect of H-T kick in. It helps also the CPU the scale in frequency so we can expect better H-T scaling just with higher core clock. "

Well, so far the Kribi Tubes benchmark shows the highest boost from Hyperthreading ever seen. Don't be surprised if you see the Kribi Tubes benchmark in BAPCO SysMark 2003... ;)

All joking apart, our hats off to the Adept Development team who have shown how much performance can be gained by optimizing carefully for SSE and Hyperthreading. At the same time, we think that the 32% boost we have seen is probably the upper limit for multithreaded applications with the current implementation of Hyperthreading. A while ago, we found out that a second CPU (Athlon MP in this case) can push Kribi performance up to 81% higher, quite a bit higher. But it must be said that Hyperthreading's 20-32% boost is incredibly high, considering that it cost only 5% extra die space.

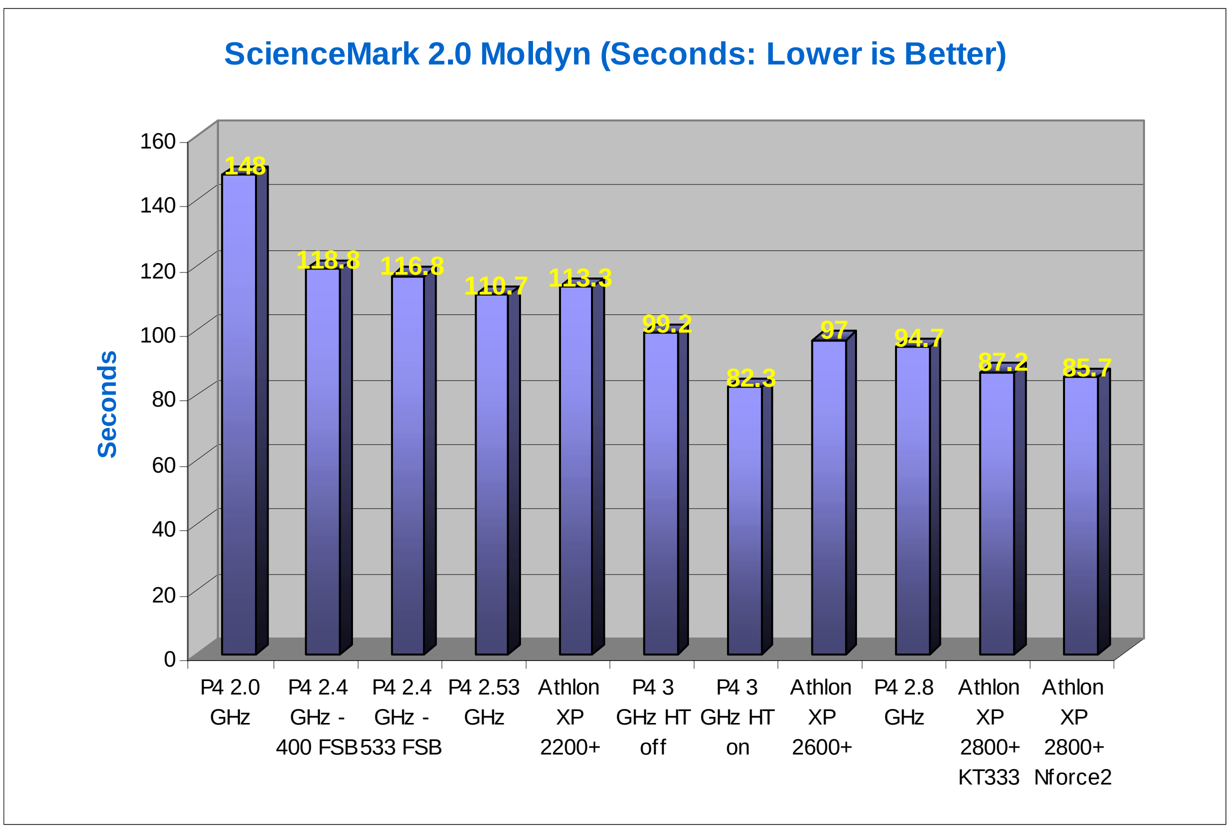
The Hyperthreaded CPU has a big advantage over DUAL Athlons: it performs very well in single threaded applications too. In those applications the Dual Athlon configurations is hampered by the higher latency and lower bandwidth of the AMD760MP chipset and its memory subsystem.

Therefore when hyperthreaded CPUs become available at lower prices, they might be very attractive for enthusiasts who only need dual CPU power from time to time.

Multithreaded Applications: Scientific Workloads with ScienceMark 2.0

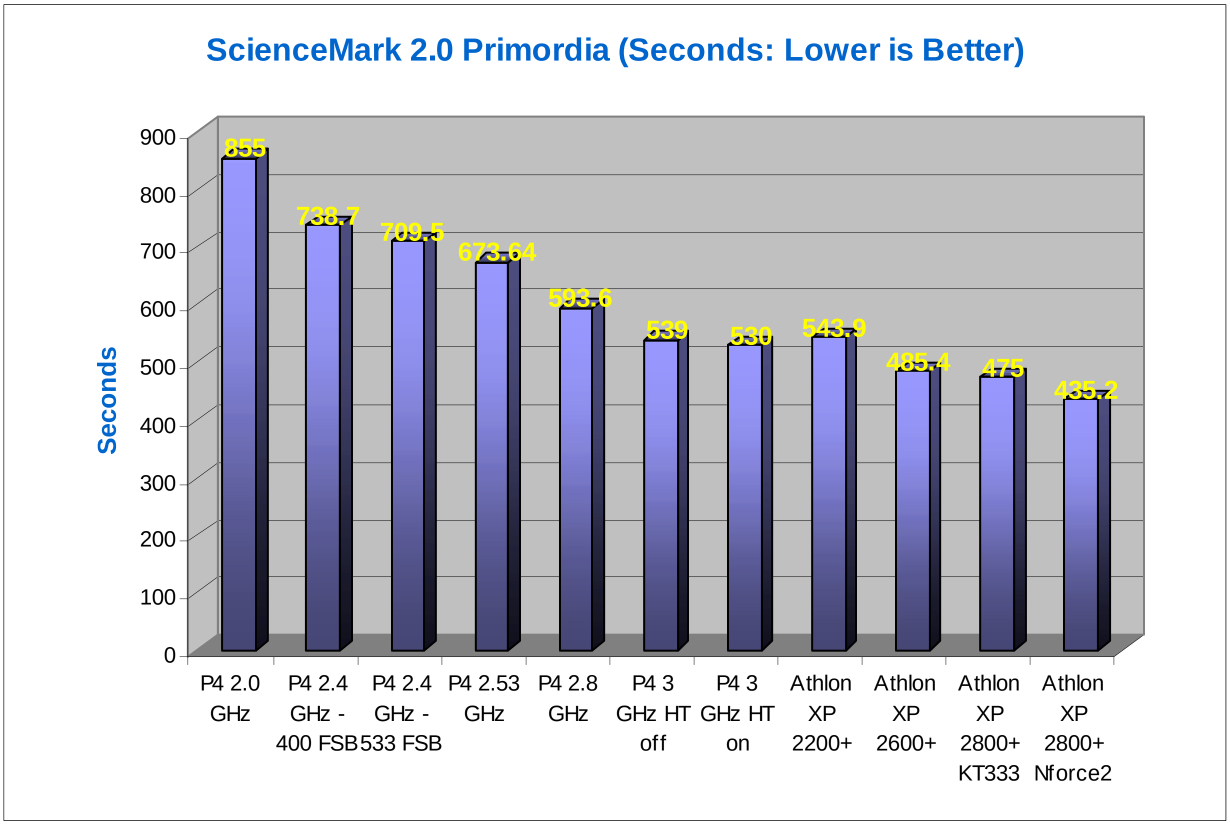
The ScienceMark 2.0 (16/08/2002) is of course more than Membench. It is a well optimized benchmark for both the Athlon and the Pentium 4 that gives us an idea of how the different CPUs handle various scientific workloads.

Molecular Dynamics is a method for simulating the thermodynamic behavior of materials using their forces, velocities, and positions. The most important of these is the force. Moldyn performs a molecular dynamics simulation of 216 Argon atoms at 140 Kelvin. More information can be found here.



Next benchmark is Primordia. From Sciencemark.org:

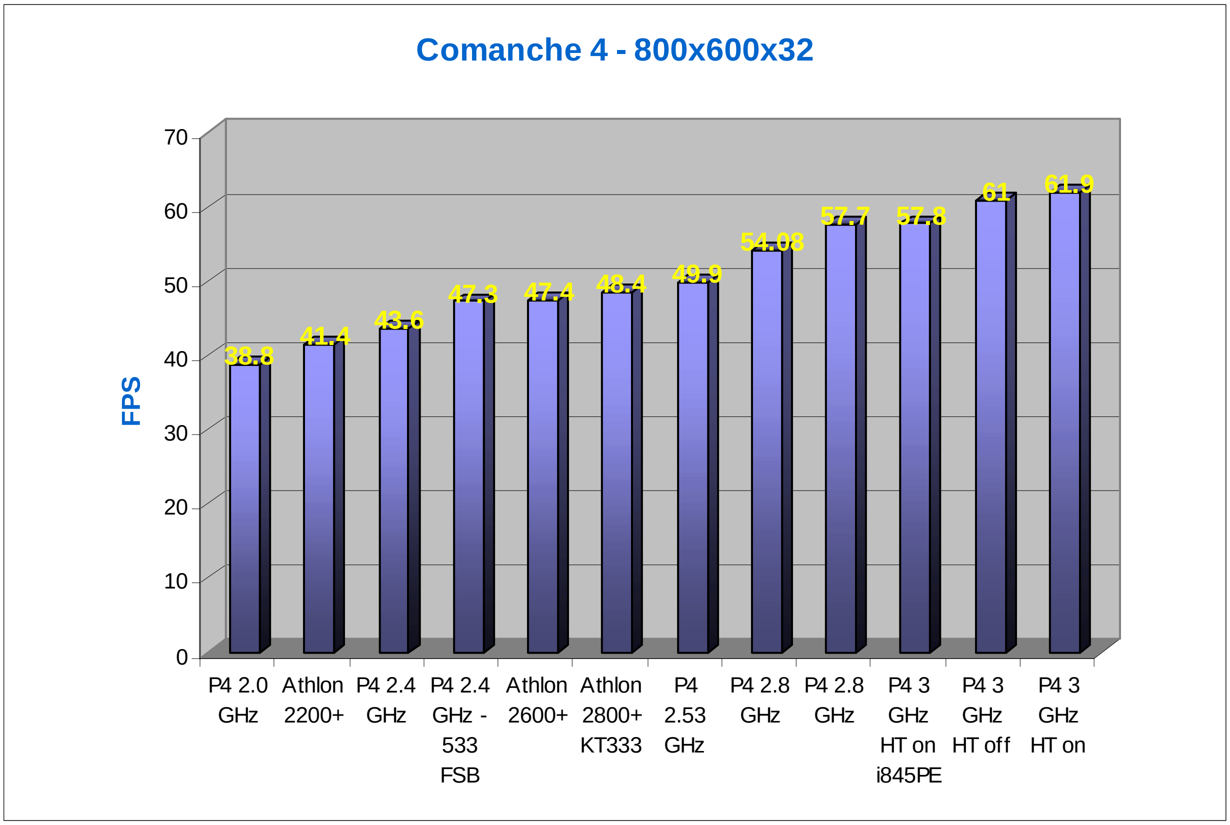
> "This code calculates the Quantum Mechanical Hartree-Fock Orbitals for each electron in any element of the periodic table . The problem involved in solving for the orbitals is discussed in great detail here. A Self-Conistent loop is performed. At each step in the loop the hartree, exchange, and the correlation potentials for each orbital are evaluated. The user has a choice of a variety of algorithms with which to evaluate these potentials."



Hyperthreading cannot help a lot if the applications bottleneck is the lack of pure x87 power. While Sciencemark is properly multi threaded, the FPU unit is simply occupied all the time. Hyperthreading can only help if one threads can fill in the executions slots that the other one doesn't need at the moment.

Gaming Benchmarks

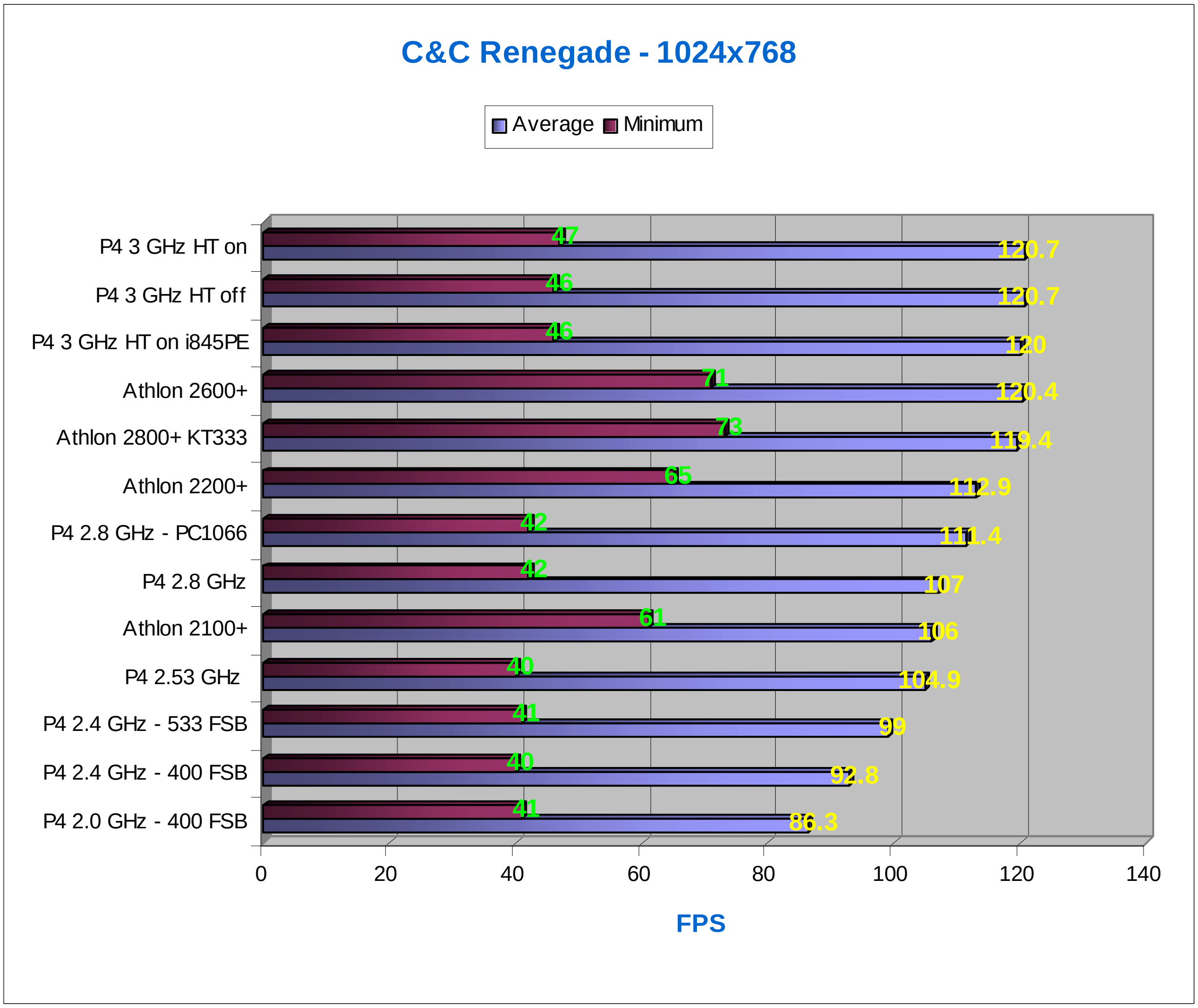
We have already discussed Comanche 4, the military helicopter simulator, earlier in the review. Here you can see all the results:



The Pentium 4 simply rules in this game.

More Gaming Benchmarks

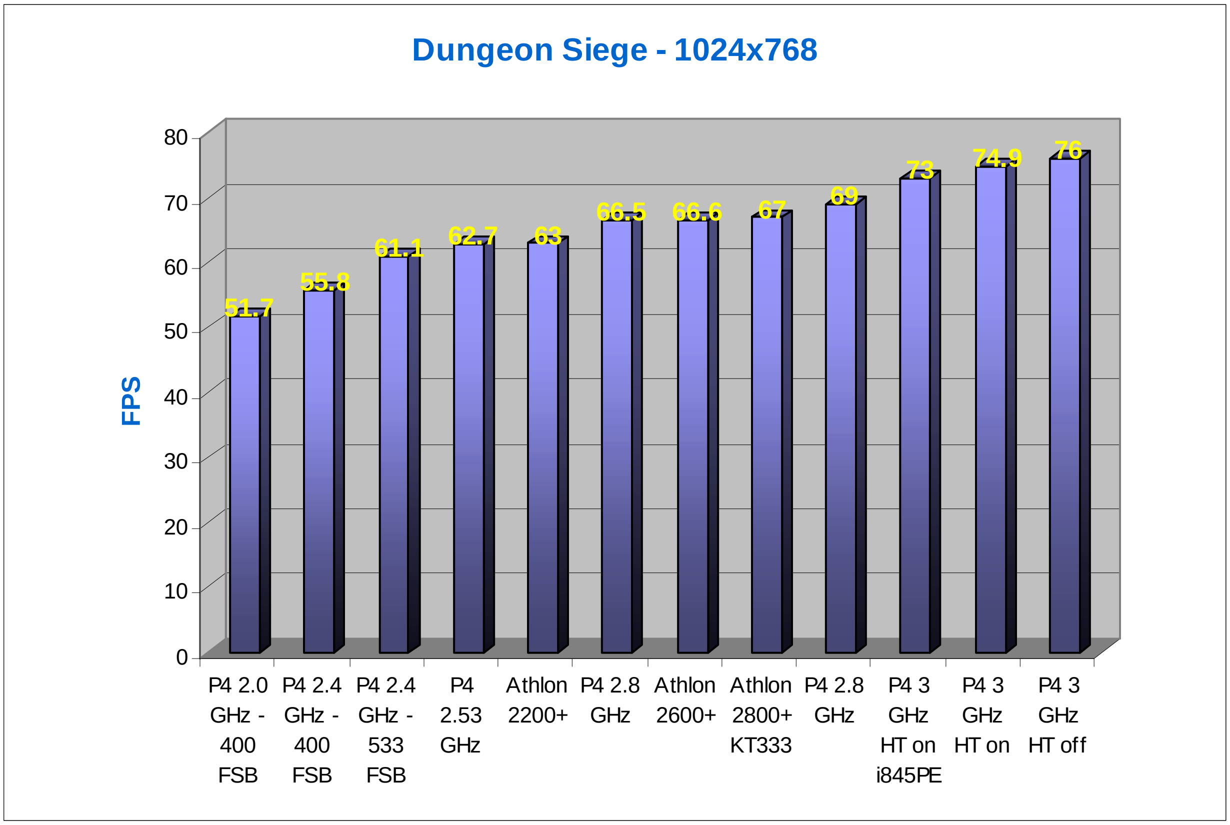
Being one of the most popular first person shooters, we felt C&C makes a nice addition to our benchmark suite, even though it relies slightly more on the videocard than the CPU (compared to Comanche).



Finally, the Pentium 4 is also able to reach the fillrate limits of our Geforce 4 Ti4400 card. Before we test any faster CPUs, we will be force to upgrade the videocard of our testbed.

Dungeon Siege

Let us try out Dungeon Siege:

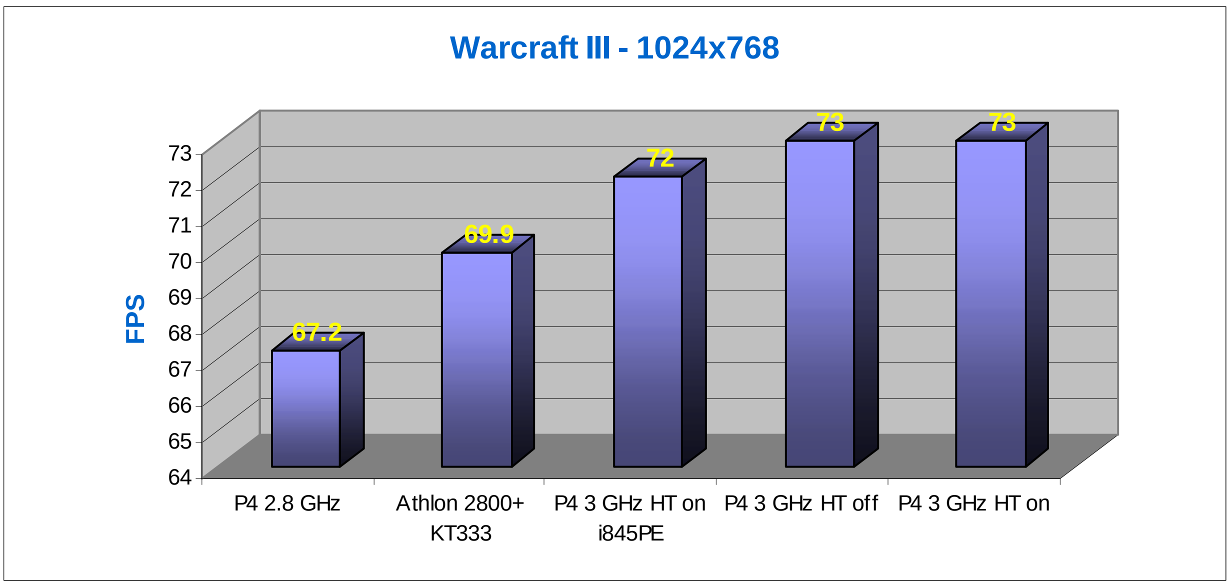


Dungeon Siege was one game that sees practically no benefit from Hyperthreading at all. One frame less is of course nothing to write home about.

Warcraft III

We benchmarked an in game cutscene of Warcraft III with fraps in the beginning of the "Human campaign".

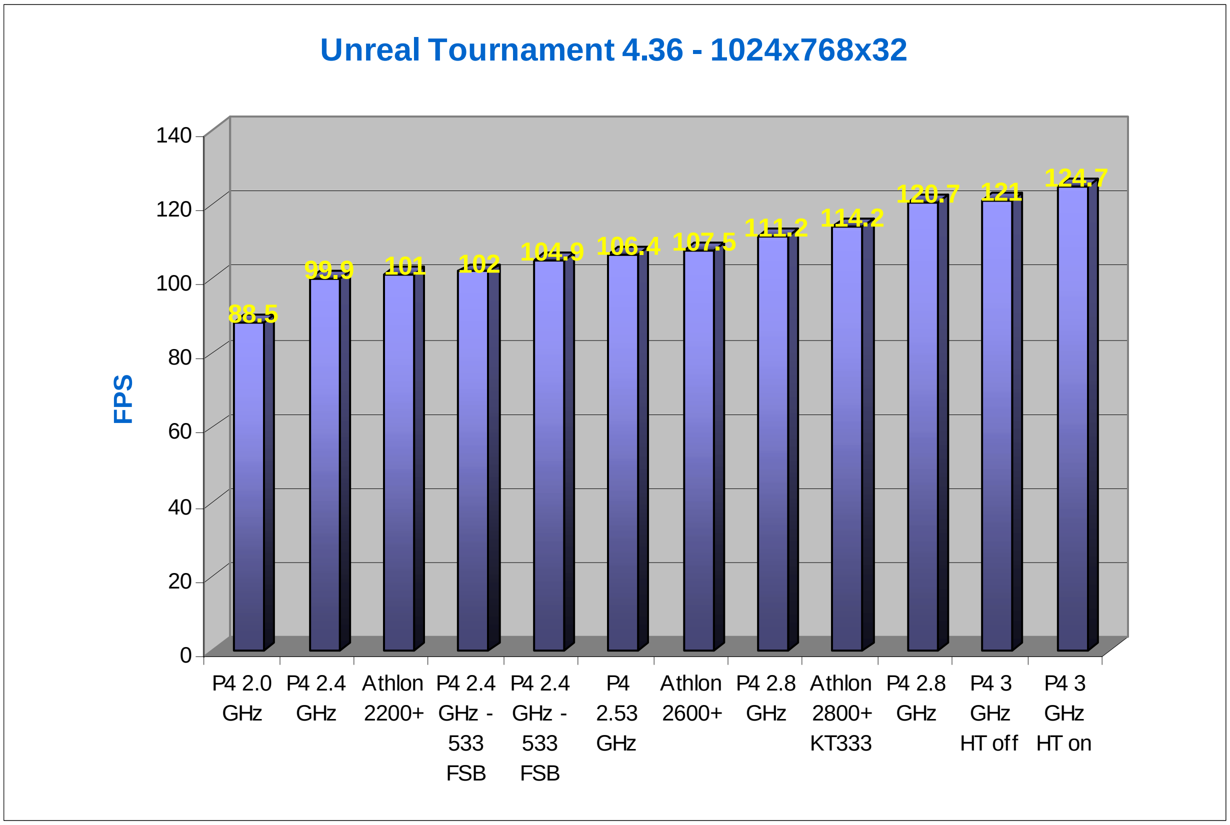
We benchmarked one of the in-game cut scenes



Nothing spectacular here, there are no real gains resulting from HT in Warcraft III. The 3.06 GHz Pentium 4 turns in the top score anyway, but the Athlon XP is not far behind.

Unreal Tournament 4.36

Being one of the most popular online games ever, we tested with Unreal Tournament 4.36 ("Game of the Year Edition") and our own "Demoace2" demo.



Several people at the message board told us that the Unreal engine might already been multi threaded for quite some time. We see a decent 3% performance improvement, which might indicate that a few less important things are handled in a second thread. A quick test with Unreal Tournament 2003 could not really confirm this. With Hyperthreading on we achieved 82.7 frames in the Asbestos deathmatch benchmark and 81.3 with Hyperthreading disabled. Rest assured that we will publish more gaming benchmarks later.

Conclusion

It is quite remarkable how almost every single threaded benchmark still got a small performance boost from Hyperthreading, between 1 and 5%. This shows that Hyperthreading has matured as it almost never decreased performance, as it did in the first hyperthreaded Xeons.

Most multi-tasking scenarios were measurably faster with Hyperthreading on, and Hyperthreading is a very smart way to improve CPU performance. But is it more responsive? In some situations yes. Applications tend to load a bit faster (see the Comanche 4 + Winrar scenario), and performance of the foreground task tends to suffer a bit. Still, don't expect Hyperthreading to enable you to run two intensive tasks on your pc. With Winrar in the background, the Comanche framerate went from 61 frames to less than 30, hyperthreaded 3 GHz CPU or not. Like we said in the introduction, we seriously doubt that a hardcore gamer would run an intensive task in the background and risk such low framerates. Hyperthreading can enable you to perform relatively light tasks in background (like playing MP3s) while runnning games or other CPU intensive tasks, however.

And while it makes your pc a bit more responsive in some situations, we doubt that CPU performance is really the reason today why PCs become non responsive. In our experience, in 95% of the "pc is not responsive" situation the real culprit is not the CPU but one of the IDE devices (especially if drivers are not working correctly in DMA mode) or one of the legacy ports (parallel, Serial...). Typical non responsive situations include getting data from a digital camera via the serial port, searching for a file on the CDROM which is hooked up on the same IDE channel as the harddisk on which the user is currentely working on etc. Of course, I am not saying anything new to the typical Ace's reader, but many people are surprised that the multi GHz beasts of today can still be "slow to respond."

In our humble opinion, the people who will gain the most from Hyperthreading are those who like to run some typical multithreaded applications on their desktop, not the multi-tasking people. If you like to compile, Animate, Encode MPEG4 or render on the same desktop system on which you play games, Hyperthreading as a lot to offer. Dual Xeon systems are in many cases too expensive, and Dual Athlons offer rather mediocre performance in gaming as the AMD760MP chipset has limited bandwidth and a relatively high latency. With Hyperthreading you get the fast gaming and single threaded performance of a typical desktop CPU, and at the same time, you get a Dual CPU system that is as fast as a lower clocked dual system. Right now is the pricetage of the Pentium 4 3.06 GHz (about 700$) is too high for most enthusiasts, but that may change in the future.

*compare for example the 3DSMax performance of a Dual Athlon 2000+ with the 3DSMax performance we measured today: both render the 3 frames of the architecture scene in about 900 seconds.