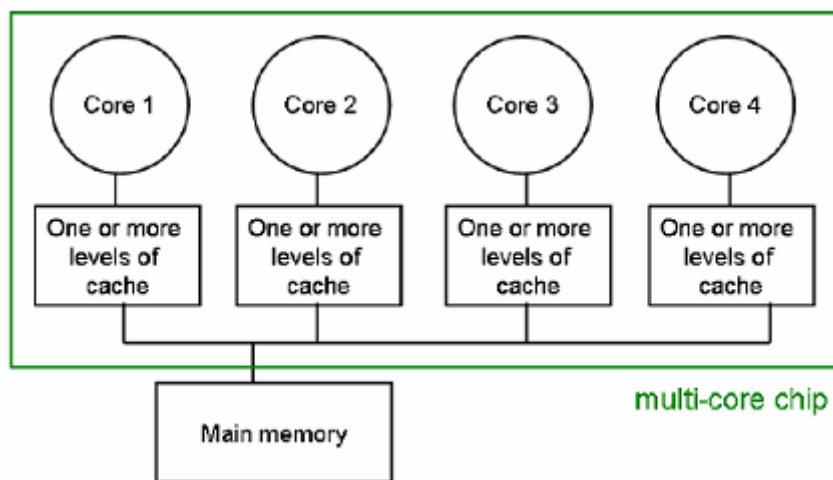


Universidad Católica
"Nuestra Señora de la Asunción"

Facultad de Ciencias y Tecnología

TRABAJO PRACTICO DE
Teoría y Aplicaciones de la Informática 2

Tecnología Multicore



Nicolás Escobar Jariton

Ingeniería Informática
10º Semestre

Prof. Ing. Juan E. de Urraza

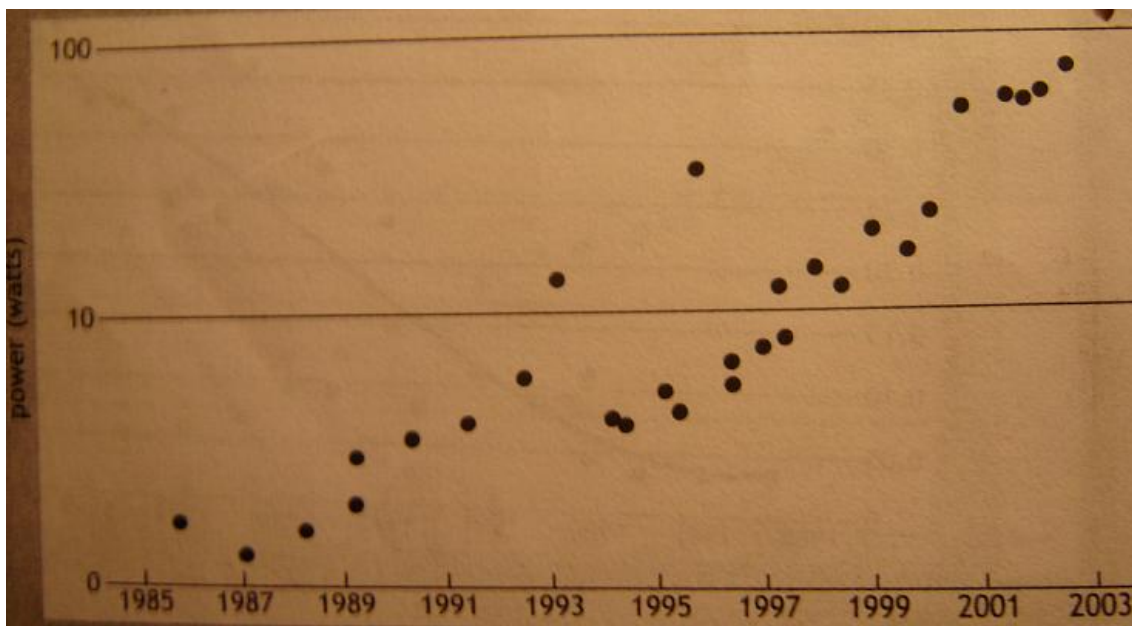
Setiembre 2006

Introducción

Con el paso del tiempo, la demanda de un mayor poder computacional va creciendo y por ello se van creando nuevas formas de cumplir con ésta. Tradicionalmente, los desarrolladores de sistemas se han enfocado en tres variables que influyen (en distintos niveles de importancia) en mejorar el rendimiento de un microprocesador: velocidad de reloj, velocidad del bus y el caché.

Mientras más aumenta la velocidad de reloj, más procesos pueden ser manejados por segundo. Sin embargo, es muy difícil aumentar la velocidad de reloj sin generar calor excesivo (debido a la cantidad de transistores que se necesitan). A medida que los procesadores fueron creciendo de manera superescalar, la energía consumida de un típico microprocesador pasó de 1 W a 100 W (Figura 1). Los diseñadores de microprocesadores siguieron utilizando más transistores en los núcleos para agregar más capacidad superescalar y de *pipelining*. El efecto global fue el incremento exponencial en la generación de calor debido a la energía necesaria para el procesamiento. Este problema se vio cuando Intel lanzó su *Pentium 4* con tecnología de 90 nanómetros y 125 millones de transistores. El consumo pasaba de 90 W a casi 120 W para frecuencias altas. La causa principal es que hay una corriente de fuga cuando el transistor está desactivado o activado que aumenta con la frecuencia. Con más transistores y por encima de los 3,5 GHz, el consumo de energía debido a estas perdidas se dispara calentándose muchísimo el *chip*.

Figura 1. Gráfico de consumo de energía de procesadores Intel. Potencia en watts por año.



Desafortunadamente, la tecnología de “enfriado” no escala de forma exponencial, tan fácil. Como resultado, los procesadores pasaron de no necesitar disipadores de calor en los 80s a utilizar disipadores de tamaño moderado en los 90s, y de tamaño gigantescos en la actualidad, a veces hasta con ventiladores dedicados que incrementan el flujo de aire que va al microprocesador. Si esto siguiera avanzando, la próxima generación de microprocesadores requerirá soluciones de “enfriado” exóticas (como el *water cooling*) que son imprácticas desde el punto de vista económico en la mayoría de los casos. Así mismo, para aprovechar el aumento de la velocidad de reloj del procesador también se deberían incrementar las velocidades de acceso a la memoria y a los dispositivos de entrada/salida.

Una alternativa interesante utilizada actualmente por varias organizaciones para el ahorro de energía es DBS (*Demand-Based Switching*). DBS permite al procesador reducir el consumo de energía (bajando la frecuencia y el voltaje) durante los periodos de baja demanda computacional. DBS está disponible en procesadores *single-core* actualmente, y su inclusión en procesadores *multicore* puede favorecer a la administración del consumo de energía, y eventualmente de la generación de calor.

Otra forma de mejorar el rendimiento del sistema es implementando mejores mecanismos de caché. Sin embargo, aunque se agreguen más cachés la mejora no es lineal (es decir, doblar la cantidad de cachés solo mejora un poco el rendimiento global del chip y aumenta el costo significativamente).

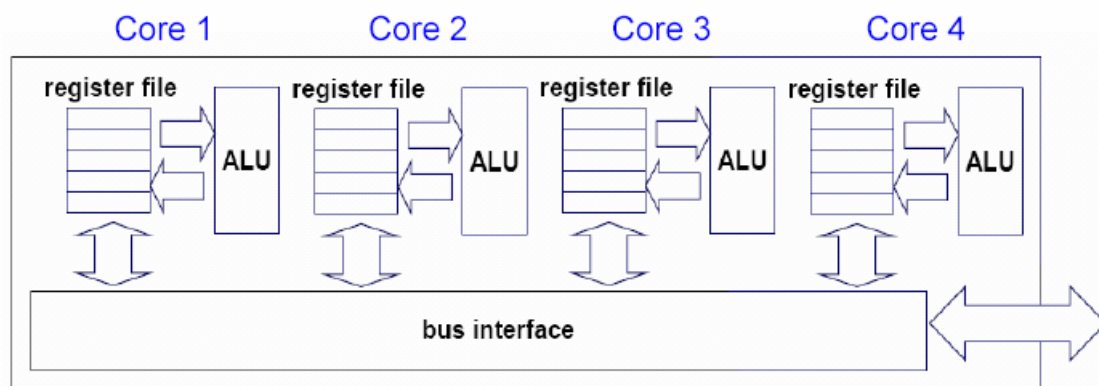
Entre las alternativas a estos problemas, está SMP (*Symmetric Multiprocessing*). En esta arquitectura dos o más procesadores idénticos se conectan a una memoria principal compartida. Los sistemas SMP permiten a cualquier procesador trabajar sobre cualquier tarea sin importar donde los datos para ésta tarea se localicen en memoria. Así, con el soporte adecuado del sistema operativo la arquitectura SMP puede mover fácilmente tareas entre procesadores de modo a balancear la carga de forma eficiente. El problema de estos sistemas (SMP y análogos) es que la memoria es mucho más lenta que los procesadores que acceden a ella, y hasta máquinas con un solo procesador tienden a pasar un tiempo considerable esperando que los datos lleguen de la memoria. El caso de SMP es peor, ya que sólo un procesador puede acceder a la memoria en un momento dado; haciendo posible que varios procesadores queden colgados. SMP es sólo un estilo de arquitecturas multiprocesador; otras incluyen NUMA, que dedican distintos bancos de memoria a procesadores diferentes. Esto permite que los procesadores se accedan a la memoria en paralelo, lo cuál mejora la eficiencia si los datos se mapean a procesos específicos. Sin embargo, el problema de

NUMA es que el costo de mover datos de un procesador a otro es muy elevado.

Aunque los multiprocesadores son una opción interesante, todavía acarrear ciertos problemas (a parte de los mencionados en los casos de SMP y NUMA) a medida que aumenta el número de procesadores. La cantidad de chips crece (uno por procesador) y por tanto, también lo hace el costo. Todo esto, llevó a los diseñadores de microprocesadores a idear una tecnología original que evite la generación de calor al incrementar el rendimiento y al mismo tiempo reduzca los costos de fabricación y por ende, de venta al usuario final. Es ahí donde nace *Multicore*.

Multicore: definiciones y funcionamiento

Un chip *multicore* o CMP (*chip multiprocessor*) es un circuito integrado, en el cual se encuentran dos o más núcleos de procesamiento. El núcleo o *core* de un microprocesador es el encargado de realizar todo el trabajo duro –contiene los cachés L1 y L2, la unidad aritmética, el mecanismo de predicción de *pipeline*, el controlador de interrupciones, etcétera. Esto, aparte de ofrecer un mayor rendimiento, ayuda a reducir el consumo de energía general y permite un mejor procesamiento de tareas simultáneas. Idealmente, un procesador *dual-core* es casi el doble de poderoso que un procesador *single-core*. Sin embargo, en la práctica, las ganancias de rendimiento llegan a un cincuenta por ciento (50%).



Multi-core CPU chip

Estos núcleos comparten ciertos recursos internos en el chip como memorias de nivel L1, así como todos los recursos externos al mismo (buses de datos, tarjetas de gráficos, etcétera). El uso de esta capacidad extra depende del sistema, y en servidores de alto desempeño pueden aplicarse para lograr procesamiento paralelo de datos, mientras que en sistemas portátiles pueden estar en reserva listos para usarse sólo cuando aplicaciones muy específicas lo requieran.

Una estructura *multicore* incluye módulos de caché. Estos pueden ser compartidos o independientes. Las implementaciones actuales de los procesadores *multicore* varían dependiendo del fabricante y del desarrollo del producto en el tiempo. Las variaciones incluyen módulos de caché compartidos o independientes, implementaciones de bus y capacidades de *threading* adicionales (como la tecnología *Hyper-Threading* de Intel).

Los primeros chip *multicore* orientados al mercado de los servidores implementaron dos o más procesadores superescalares convencionales en una sola pieza. La motivación inicial para ello fue reducir el volumen – múltiples procesadores podían entonces ocupar un espacio donde

anteriormente sólo uno ocupaba, por lo que el rendimiento global por unidad de volumen se incrementó. Así mismo, también ocurrieron ahorros de energía porque los procesadores en una sola pieza podían compartir una conexión al resto del sistema, reduciendo la infraestructura necesaria de alta velocidad. Algunos sistemas *multicore* (como los primeros anunciados por AMD e Intel) compartían únicamente una interfaz entre los núcleos de procesamiento (Figura 2) pero otros compartían uno o más niveles de caché *on-chip* (Figura 3) lo cual permitía comunicación inter-procesadores entre los núcleos del CMP sin necesidad de accesos *off-chip*.

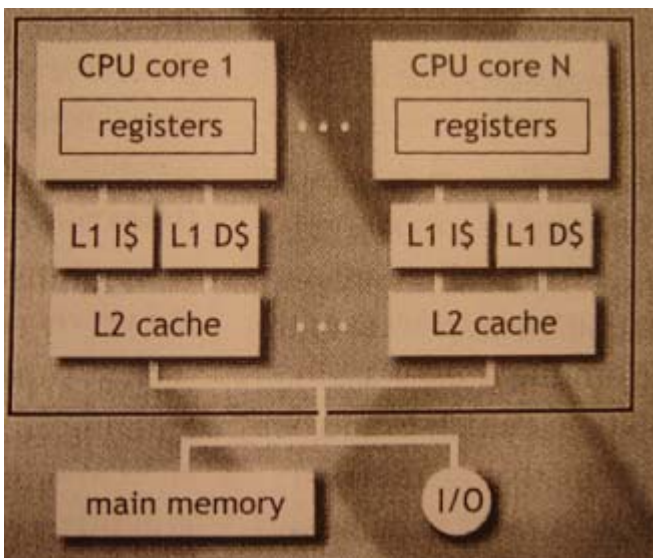


Figura 2. Arquitectura que sólo comparte la interfaz entre *cores*

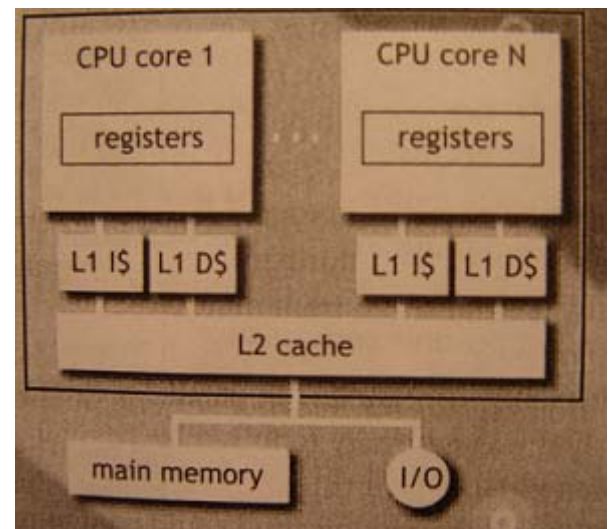


Figura 3. Arquitectura que comparte niveles de caché *on-chip*

Con *multicore*, es posible obtener esencialmente el mismo (o inclusive mejor) rendimiento que con *single-core* en balanceos de carga orientados a servidores con tan sólo la mitad de velocidad de reloj. Cada solicitud puede tomar hasta el doble de tiempo para ser procesada debido a la reducción del reloj. Sin embargo, con la mayoría de estas aplicaciones la reducción será mucho menor debido a que el tiempo de procesamiento ante las solicitudes está generalmente limitado por la memoria o el disco, antes que por el procesador. Por ello, como dos solicitudes pueden procesarse simultáneamente, el rendimiento global será igual o mejor (a no ser que los recursos de memoria y disco sean extremadamente escasos).

Aunque el rendimiento es igual, o quizás un poco mejor, esta tecnología es todavía ventajosa a nivel de sistema. La baja velocidad de reloj permite diseñar el sistema con suministro mucho más bajo de energía (una reducción lineal).

Con el “boom” del *multicore*, la tendencia es reemplazar los procesadores superescalares por varios núcleos de procesamiento pequeños. Cada

core procesará las solicitudes de forma más lenta que un procesador superescalar pero esta baja de latencia se compensa con el hecho que en el mismo chip se pueden ocupar una mayor cantidad de procesadores. Llevando esta idea más lejos, se puede conseguir más eficiencia con la inclusión de múltiples hebras en cada core.

Debido a que cada core tiende a pasar un buen tiempo esperando que se satisfagan solicitudes de memoria, tiene sentido asignar a cada núcleo de procesamiento varios *threads* incluyendo múltiples registros, uno por *thread*, en cada core. (Figura 4).

Así, aunque otros hilos estén esperando, el procesador puede todavía ejecutar instrucciones.

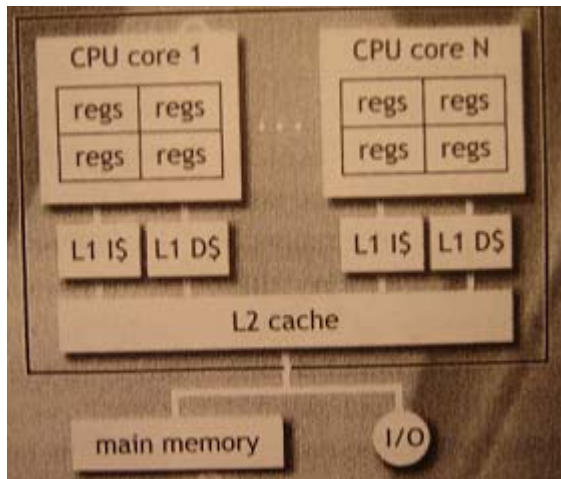


Figura 4. Multithreaded shared-cache chip multiprocessor

Un gran número de hebras también pueden permitir que los procesadores envíen más solicitudes de memoria en paralelo, incrementando así la utilización de los sistemas de alto *pipelining* de memoria de la actualidad. En general, los *threads* tendrán más latencia porque en ciertos momentos estarán todos activos y compitiendo por el uso del core.

Multiprocessing vs. Multicore

Aunque existe una controversia de si múltiples cores en un chip son equivalentes a múltiples procesadores (ver apartado de licenciamiento más abajo), en este trabajo práctico considero (basándome en informaciones técnicas de diversos artículos) que sí hay diferencia entre ambos.

Los sistemas multiprocesadores ubican sus procesadores en distintas piezas y el paso de datos entre ambos es lento porque el viaje se realiza *off-chip*. En el caso de *multicore*, los núcleos de procesamiento están en la misma pieza de silicio y gracias a su proximidad permiten aumentar la velocidad del flujo de información entre ellos.

Multicore: Más poder de procesamiento y menos consumo

¿Por qué usar un procesador de 64 bits y *multicore* produce menos calor que un procesador normal de rendimiento semejante? En efecto, la energía generada por el puro cómputo es similar, pero hay ahorros apreciables de energía por otros conceptos.

Por ejemplo, los 64 bits permiten ampliar la memoria RAM desde el actual máximo de 4 Gb a 16 Exabit. Pasarán años hasta que alguna computadora tenga una memoria RAM que se mida en Exabit pero magnitudes entre 10 y 100 Gb están cerca, lo que permitirá utilizar caches de datos de considerable tamaño que reducen la necesidad de acceder al disco duro para leer o escribir datos. Con ello se logra un fuerte ahorro de energía disipada por el motor de arrastre del disco duro, a la vez que mejora la velocidad de proceso. Por otra parte al elaborar datos más grandes, algunos procesos pueden tratar en un solo paso el doble de información que anteriormente, con lo que se ahorran pasos de proceso.

Con un único procesador, una aplicación exigente, pone a trabajar a tope al procesador generando gran cantidad de calor. Si existen diversos *cores*, solo trabajará al máximo el *core* que soporte esa aplicación, los otros *cores* irán más desahogados. Una nueva técnica ya utilizada en ordenadores portátiles, la *SpeedStep*, permite variar la frecuencia de trabajo desde unos 600 Mhz hasta su frecuencia nominal (es lo contrario del *overclocking*). De modo que al subir la temperatura, se reduce la velocidad del reloj, procurando mantener estable la temperatura sin que el rendimiento se resienta en exceso ya que todos los núcleos colaboran en hacer avanzar el trabajo a la vez

Más ventajas de Multicore

- La proximidad de múltiples *cores* en una misma pieza tienen la ventaja de que la coherencia de caché puede operar a una velocidad más alta de reloj que si las señales viajasen *off-chip* (fuera de chip entre microprocesadores).
- Un procesador con dos núcleos utiliza menos energía que dos procesadores de un núcleo emparejados, principalmente por la energía extra necesaria para enviar señales externas al chip y porque la geometría del silicio permite a los núcleos operar a voltajes más bajos; lo cual reduce la latencia. Así mismo, con un único procesador, una aplicación exigente pone a trabajar a tope al procesador, generando gran cantidad de calor.

Si existen diversos *cores*, se balanceará la carga de forma más eficiente (o el trabajo se dará a un *core* mientras el otro reposa –en el caso del *dual-core*) lo que en conjunto hace que se genere menor energía. Incluso, si una aplicación multi-hilo coloca periódicamente a pleno rendimiento a todos los núcleos a la vez, esto ocurrirá de forma intermitente, pudiendo bajar la temperatura en el intermedio.

- CMP requiere sólo un esfuerzo modesto de ingeniería para cada generación de procesadores. Cada miembro de la familia de procesadores sólo requiere copias adicionales del *core* y hacer algunas modificaciones a la lógica de conexión entre los núcleos para acomodar los procesadores adicionales en cada generación (evitando así un completo rediseño). Más aún, el circuito típicamente sólo necesita ajustes menores de generación en generación, ya que externamente CMP no cambia en esencia a medida que aumenta la cantidad de *cores*.

La única diferencia real es que el circuito tendrá que lidiar con mayores requerimientos de ancho de banda de entrada/salida a medida que la cantidad de *cores* va creciendo. En los distintos procesos de generación de las piezas, los ahorros en costos de ingeniería pueden ser significantes, porque es relativamente fácil incluir nuevos núcleos en cada generación (según los expertos). Así mismo, el mismo esfuerzo de ingeniería puede ser amortizado entre la gran familia de procesadores relacionados. Simplemente variando los números y la frecuencia de reloj de los procesadores se puede utilizar en esencia el mismo hardware con distintos precios/rendimientos.

- La eficiencia del sistema mejora cuando el sistema ejecuta múltiples aplicaciones y se aprovechan las aplicaciones multi-hilo, como motores de bases de datos y software de ingeniería y gráficos. A diferencia de *single-core* que asigna periodos de tiempo a cada tarea, *multicore* puede correr tareas de forma simultánea.

Algunas desventajas de Multicore

- Se deben realizar ajustes a las aplicaciones existentes para que maximicen la utilización de los recursos computacionales provistos por los procesadores *multicore*. Así mismo, la habilidad de los procesadores *multicore* de incrementar el rendimiento depende del uso de múltiples hebras en las aplicaciones. Por ejemplo, la mayoría de los video juegos (en el 2006) correrán más rápido en un procesador *single-core* de 3GHz que en un *dual-core* de 2GHz,

porque son incapaces de utilizar de forma eficiente más de un núcleo a la vez.

- El hecho que dos núcleos de procesamiento compartan el mismo sistema de bus y el ancho de banda de memoria limita el real aprovechamiento de la tecnología. Si un sólo *core* tiene problemas con el ancho de banda de memoria, ir a la tecnología *dual-core* sólo mejorará en un rango de 30 a 70%.
- Este sistema tiene inconvenientes en el sistema de interconexión, que está limitado por la resistencia y capacitancia de los conectores, sean aluminio, cobre u otro metal de mejores propiedades o el uso de semiconductores diferentes al silicio como el germanio, indio o compuestos como *SiGe* o *InP*.

Aprovechamiento de la tecnología Multicore en el software

El software se beneficia de las arquitecturas *multicore* en los casos donde el código puede ser ejecutado en paralelo. Bajo la mayoría de los sistemas operativos más comunes esto requiere que el código se ejecute en distintos hilos. Cada aplicación que se ejecuta en un sistema corre en su propio *thread* y así múltiples aplicaciones se pueden beneficiar del *multicore*. Así mismo, cada aplicación puede tener varios *threads* de ejecución y la misma debe ser escrita específicamente para realizar esto. El sistema operativo también tiende a ejecutar muchos hilos como parte normal de su operación. La ejecución de máquinas virtuales se beneficiará también de la adopción de las arquitecturas *multicore* ya que cada máquina virtual corre independiente de las otras y puede ser ejecutada en paralelo.

La mayoría de los programas no están escritos para utilizar múltiples *threads* debido al desafío de hacerlo. La programación de código multi-hilo requiere, a veces, la coordinación compleja de los *threads* y puede introducir dificultades en el proceso de encontrar bugs debido a los datos que se comparten entre los distintos hilos. Así también, existe una falta de motivación para escribir aplicaciones multi-hilo porque muchas veces las ventajas que introducen en teoría no son realmente aprovechadas en la práctica.

Las técnicas de programación paralela pueden aprovechar al máximo las arquitecturas *multicore*. Algunos modelos de programación paralela existentes son OpenMP y MPI.

Entre los sistemas que utilizan plenamente las tecnologías multicore se citan: Maya, Blender3D, Quake 3 & 4, Allied Force, 3DS Max, Adobe Photoshop, Windows XP Professional, Windows 2003, Mac OS X, Linux,

GigaSpaces EAG y muchos sistemas operativos orientados al uso de servidores.

Así, la concurrencia adquiere un rol central en una aplicación verdaderamente paralela. Los pasos básicos para diseñar una aplicación paralela son:

Partición: esta etapa busca exponer las oportunidades para la ejecución paralela. Así, se enfoca en definir un número grande de pequeñas tareas para dar paso a una descomposición de grano fino del problema.

Comunicación: las tareas generadas por una partición se intentan ejecutar de manera concurrente pero no pueden, en general, ejecutarse independientemente. La computación a ser llevada a cabo por una tarea típicamente requiere datos asociados con otras tareas. Por ello, los datos deben ser transferidos entre tareas para permitir la computación. Este flujo de información se define en la fase de comunicación del diseño.

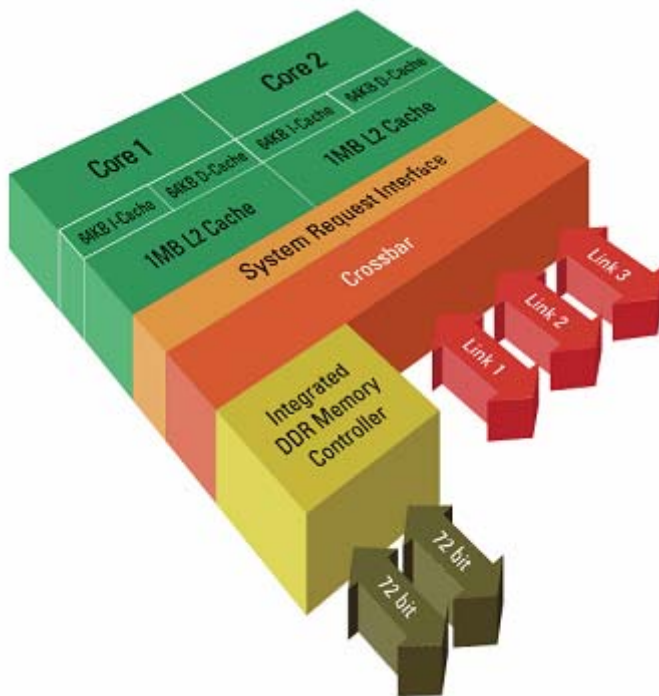
Aglomeración: en esta etapa, la planificación se mueve de lo abstracto a lo concreto. Se re-visitan las decisiones hechas en las fases de partición y comunicación con el objetivo de obtener un algoritmo que se ejecutará eficientemente en alguna clase de computadora paralela. En particular, se considera si es útil o no, combinar o aglomerar las tareas identificadas en la fase de partición para así proveer un menor número de tareas, cada una de un tamaño más grande. También se determina si es factible replicar datos o computaciones.

Mapeo: es la última etapa del proceso de diseño de un algoritmo paralelo y se especifica que proceso llevará a cabo cada tarea.

Ejemplos comerciales

Tecnología Multicore de AMD

Los procesadores *multicore* de AMD se caracterizan por su simpleza. AMD desechó la arquitectura FSB (*front-side bus*) y en vez, utilizó un diseño más innovador llamado *Direct Connect Architecture*, que ayuda a reducir los cuellos de botella de la arquitectura de sistemas a nivel de procesador.



Componentes básicos

AMD64 CORE: permite la computación simultánea de 32 y 64 bits y elimina la barrera de los 4GB de memoria impuestos por la arquitectura de sólo 32 bits.

L1 Instruction and Data Cache: caches internos de baja latencia

L2 Cache: cachés de segundo nivel con baja latencia

System Request Interface: maneja y prioriza el acceso de los *cores* al *crossbar switch* para solicitudes de memoria y de datos de entrada/salida.

Crossbar: conecta directamente los *cores* con las interfaces de memoria y de entrada/salida permitiendo que cada *core* ejecuta programas y tenga acceso a los datos.

Integrated DDR Memory Controller: reduce la latencia de los accesos de memoria sobre los tradicionales controladores basados en el *front-side bus* (FSB).

Tecnología HyperTransport: provee un enlace punto a punto de alta velocidad para conectar los procesadores. Esta tecnología también conecta directamente los procesadores a las interfaces de entrada/salida a una velocidad de 3.2GB/s por enlace, permitiendo un ancho de banda de hasta 24GB/s por procesador.

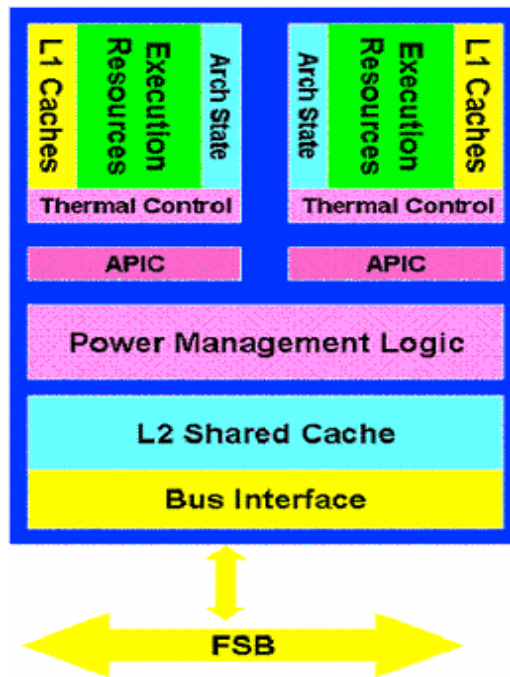
Interfaz de memoria: poca latencia, alto ancho de banda, 128-bit DDR con interfaz SDRAM que provee hasta 6.4GB/s de ancho de banda de memoria @ DDR-400.

Quadcore de AMD

Los primeros modelos *quad-core* (cuatro núcleos en un CMP) de AMD probablemente estarán disponibles a mediados del 2007 o inicios del año 2008 y utilizarán memoria DDR2 mientras que en siguientes generaciones llegará el soporte para la DDR3. Los nuevos núcleos se llamarán Deerhound, Greyhound, Zamora y Cadiz. A partir del Greyhound ya estará implementado el bus *HyperTransport 3.0* y este modelo y los siguientes podrán montar memoria DDR2 o DDR3 indistintamente, pero no a la vez.

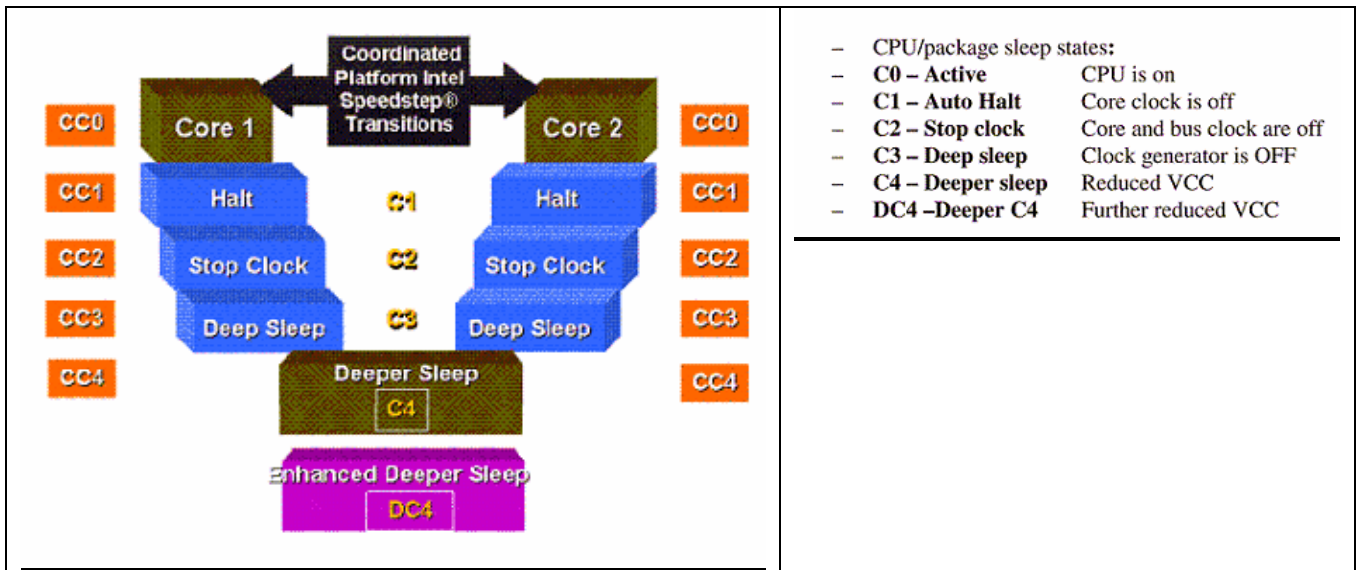
Tecnología Multicore de Intel

El procesador *Intel Core Duo* implementa la microarquitectura *shared cache-based CMP* para así maximizar el rendimiento de las aplicaciones. La siguiente figura describe la estructura general de la implementación.



- Cada núcleo incluye una unidad APIC (*Advanced Programmable Interrupt Controller*) independiente, que es presentada al sistema operativo como un procesador lógico separado.
- Desde el punto de vista del software, es totalmente compatible con los procesadores Pentium 4 con la tecnología *Hyper-Threading* y los sistemas basados en DP (*Dual Processor*). Se puede realizar optimizaciones para mejorar el rendimiento del *share-based cache organization*.
- Cada núcleo posee una unidad independiente de control termal.
- El sistema combina un control de estado de energía por núcleo

La forma tradicional de controlar la energía y el calentamiento de los dispositivos es a través de una interfaz software/hardware. Uno de los esquemas más comunes para llevar a cabo esto es llamado ACPI, donde el sistema define diferentes niveles de *sleep modes*, y cada uno de los estados representa una forma más eficiente de ahorrar energía. El procesador *Intel Core Duo* presenta dos vistas distintas del estado de energía del sistema; internamente se manejan los estados de cada núcleo independientemente y externamente se ve al sistema como un solo estado de energía sincronizado. La figura de abajo presenta un bosquejo de este esquema.



Quadcore de Intel

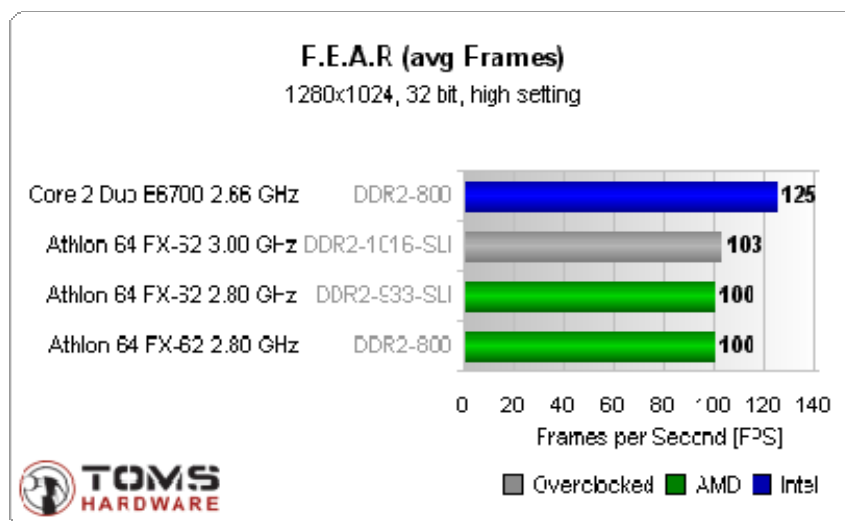
El procesador *quad-core* de Intel denominado Clovertown, empezará a distribuirse a fines del año 2006 e ingresará al mercado probablemente a inicios del 2007. Clovertown se lanzará para servidores con procesadores duales, lo que significa que estos servidores contarán realmente con 8 procesadores (dos procesadores con cuatro núcleos cada uno).

Duelo AMD vs. Intel

Luego de un tiempo en donde AMD era el ganador imbatible de los duelos entre estos dos contendientes, Intel presenta su nuevo modelo; el *Core 2 Duo* que dio vuelta la contienda. No sólo, los procesadores *Core 2 Duo* son más rápidos que los actuales *Athlon 64 X2*, sino que también consumen menos energía.

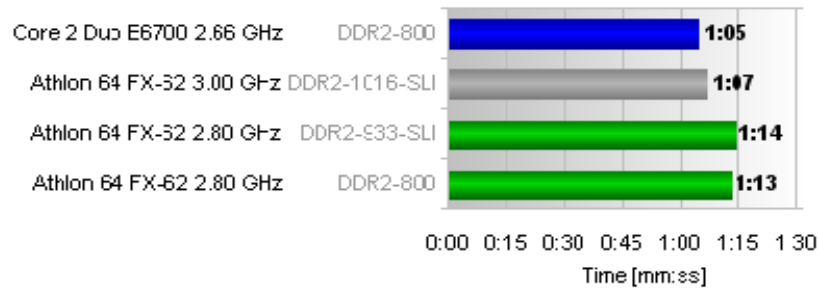
El *Core 2 Duo* se volverá el líder indiscutible en la relación rendimiento/energía. Intel no habla de cambios en la arquitectura de su procesador; sino de un completo rediseño. Los ingenieros tomaron elementos del actual *Pentium D Netburst* y agregaron ingredientes que hicieron que los procesadores *Pentium M* y *Core Duo* sean exitosos. Es ahí donde nació la arquitectura del *Core 2*.

Existen una gran variedad de *benchmarks* que muestran como este nuevo modelo de Intel le lleva la delantera a AMD (y otros que dicen lo contrario pero la tendencia prueba que Intel realmente lo hace). Más abajo exponemos algunos de ellos.



I-Tunes 6.04.2

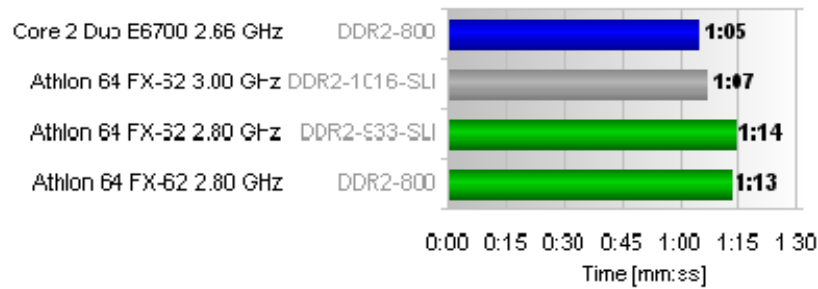
Wave File 701 MB



Overclocked AMD Intel

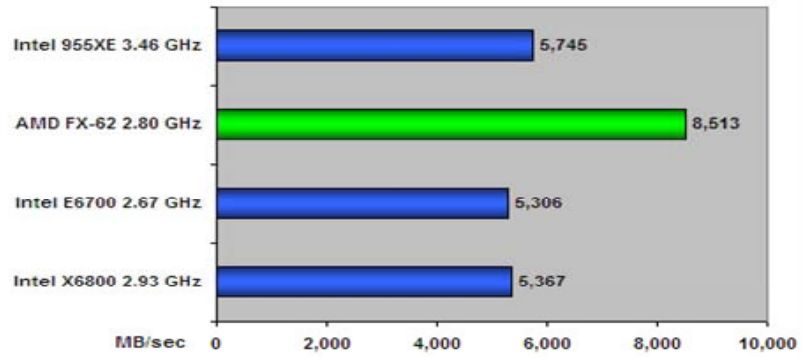
Windows Media Encoder 9

Windows Media Video 9 Advanced Profile



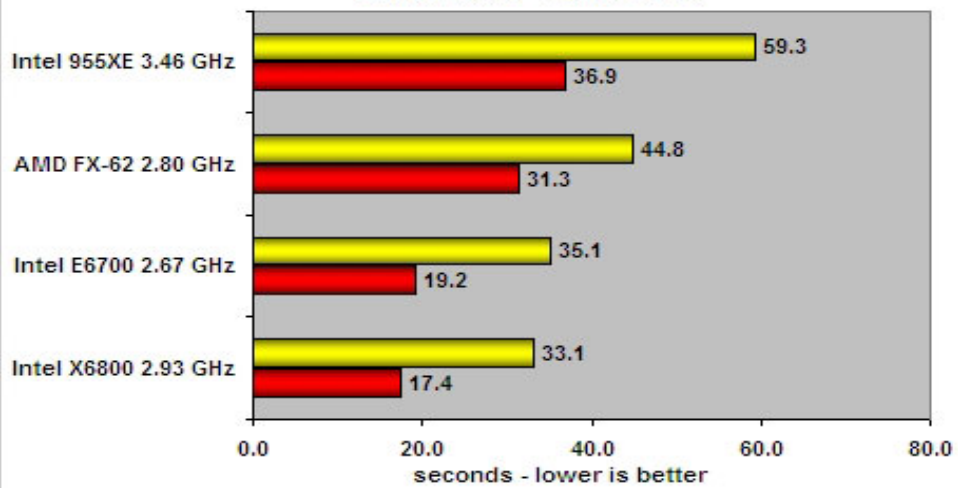
Overclocked AMD Intel

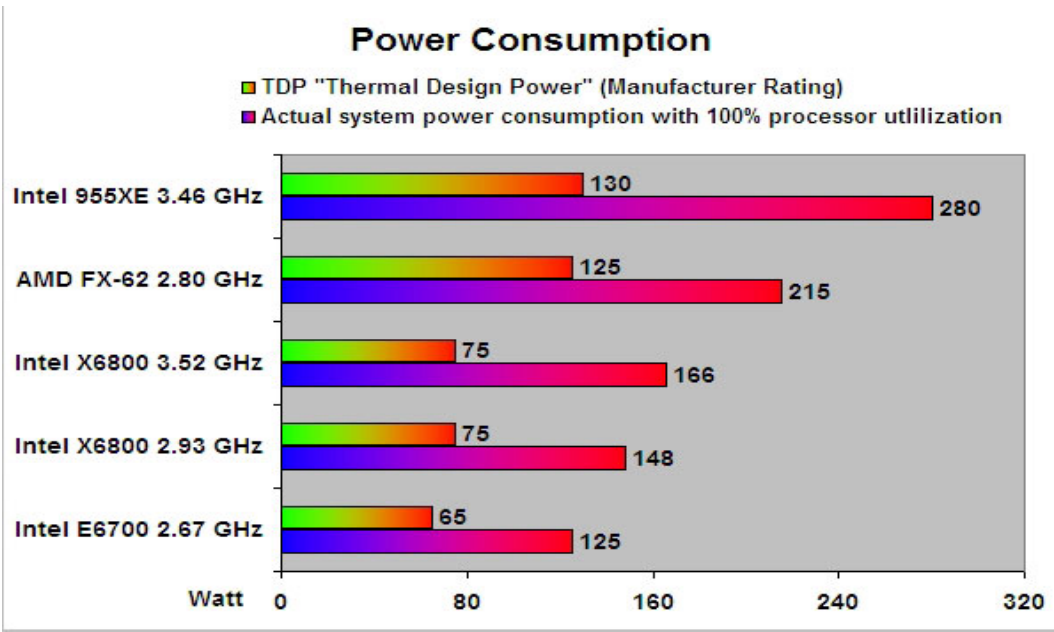
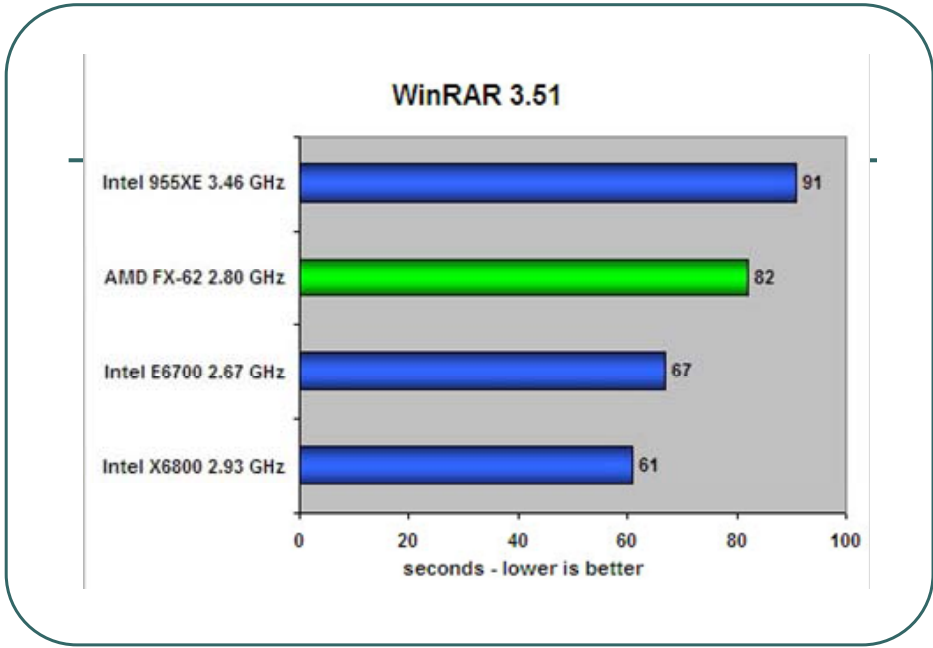
ScienceMark 2.0 Memory Bandwidth



Calculations

■ Super Pi 1 M ■ Pifast 10 M





Aún así, elegir el mejor procesador no es fácil. Existen distintas líneas de productos y modelos en el mercado, y el sólo hecho que Intel supere a AMD en la relación rendimiento/energía no significa que el Core 2 es la mejor opción por defecto. Existen productos superiores a precios más elevados.

Aunque Intel lleve la delantera momentánea, la línea de Athlon64 de AMD provee mucho poder de cómputo para las tareas del día a día y a buen precio. Así mismo, este modelo también puede ser considerado eficiente en términos de energía: los procesadores AMD tienen un controlador de memoria (como habíamos mostrado más arriba, entre los componentes básicos de la tecnología *multicore* de AMD) directamente integrado en el procesador, mientras que Intel es más conservador en ese sentido y coloca este elemento fuera del chip, donde consume más energía y agrega mayor latencia.

Otros procesadores multicore

1. CELL

El procesador CELL está basado en la arquitectura Power de IBM y ha sido desarrollado en colaboración con el grupo Sony y Toshiba. Con arquitectura multinúcleo (9 cores) y capacidad de comunicación de alta velocidad, está diseñado para ejecutar tareas de cálculo intensivo y aplicaciones multimedia de banda ancha, utilizadas en los contenidos digitales, como películas, videojuegos y otras formas de entretenimiento. Teóricamente CELL sería hasta 35 veces más rápido que un procesador doble núcleo.

2. Plataforma de doble núcleo de VIA

El procesador VIA Eden –N core funciona con una frecuencia de 1Ghz por núcleo, un FSB de 133 Mhz y una caché L2 de 64Kb. Viene soldado a una placa base ya que es de tipo NanoBGA (algo más pequeño que una moneda).

Teóricamente este procesador tiene ciertas limitaciones: los dos núcleos no se comunican entre sí, con lo cual todo pasa primero por el chipset, lo que aumenta la latencia en los procesos. Por lo contrario, los 14W de este procesador son tres veces más bajos que el consumo del Core Duo de Intel.

Las pruebas de rendimiento son bastante malas, y las funcionalidades también. La reproducción de DivX va a saltos y la reproducción de video de alta definición es imposible, comparando este nuevo C3 con un Core Duo de Intel, consiguieron una relación del rendimiento de 1:3 a favor del procesador Intel en todas las pruebas.

3. Sun Microsystems

- UltraSPARC IV
- UltraSPARC IV+
- UltraSPARC T1 eight cores, 32 threads

4. PowerPC G5 con doble núcleo independiente de 64 bits

5. ARM MPCore

Contenedor multicore para núcleos ARM9 y ARM11, desarrollados para aplicaciones embebidas de alta performance y entretenimientos. El ARM9E se usa en el Nintendo DS y telefonía móvil.

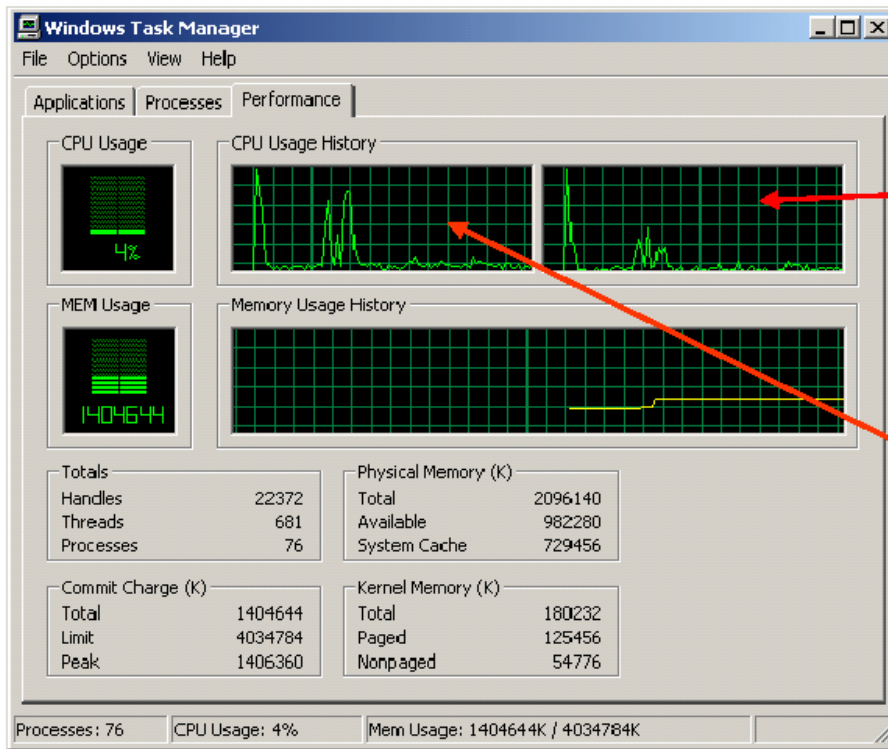
6. Microsoft's Xbox 360

Utiliza un procesador PowerPC de tres núcleos.

Licenciamiento de Multicore

Otro tema debatible, es el licenciamiento del software que corre sobre tecnologías *multicore*. Típicamente las aplicaciones empresariales están licenciadas por procesador. En el pasado, una CPU era un procesador (y la mayoría de las computadoras tenían un sólo procesador) y no existía ambigüedad. Ahora existe la posibilidad de contar los núcleos o *cores* como procesadores y cobrar al cliente dos licencias cuando utilizan una CPU dual-core. Sin embargo, la tendencia parece contar los chips *dual-core* como un único procesador. Oracle toma los modelos *dual-core* de AMD e Intel como de un único procesador pero tiene otros números para distintos tipos. IBM, HP y Microsoft toman los módulos *multi-chip* como múltiples procesadores. Por tanto, parece que la industria lentamente va tomando a la pieza como un procesador, sin importar la cantidad de núcleos que ésta contenga. En octubre de 2004, Microsoft anunció que trataría los chips *multicore* como un único procesador, en vez de tratarlo como N procesadores para los temas del licenciamiento de su software.

Imagen del administrador de tareas de Windows con dos núcleos



core 2

core 1

Conclusión: Multicore, una transición inevitable

Como resultado de estas tendencias, estamos en un punto en donde la tecnología *multicore* está entrando fuertemente al mercado de los computadores. El impacto más grande de la misma se da en el área de performance debido a que *multicore* mejora la relación rendimiento/energía. Así mismo, la tecnología de múltiples núcleos trajo un impacto importante en el área de computaciones de crítica latencia.

Aunque sea necesario utilizar múltiples hebras paralelas de ejecución en la mayoría de las aplicaciones de latencia crítica para realmente aprovechar las ventajas del *multicore*, los CMP hacen que este proceso sea más fácil que con los multiprocesadores *multichip*, debido a la baja latencia en la comunicación entre núcleos.

Por todo esto, la transición a la tecnología *multicore* es inevitable. Los esfuerzos anteriores de acelerar las arquitecturas de un procesador con técnicas que no modifican el modelo básico de computación de Von Neumann (como el *pipelining* y la ejecución superescalar) encuentran duras resistencias técnicas. Como resultado, la industria de los microprocesadores lidera el camino hacia la tecnología *multicore*; sin embargo, el beneficio total de estas arquitecturas no será aprovechado completamente hasta que la industria del software implemente en mayor grado la programación paralela. El arte de la programación orientada a multiprocesadores (hasta ahora sólo dominada por una minoría de programadores) es más complejo que la programación en máquinas de un solo núcleo y requiere un entendimiento de nuevos principios computacionales, algoritmos y herramientas de programación.

La expansión de *cores* o núcleos de procesamiento será el tema dominante de los fabricantes de procesadores en los próximos años. Al final de la década, los chips con decenas de *cores* serán posibles. Así mismo, en diez años es teóricamente posible que los CMP incluyan cientos de núcleos.

Bibliografía

Textos

“Planning Considerations for Multicore Processor Technology”. Dell Power Solutions. John Fruehe. Mayo 2005.

“Chip multiprocessors’s promise of huge performance gains is now a reality”. QUEUE. Edición Setiembre 2005.

“Multi-core technology: trends and design challenges” Markus Levy. Presidente del The Multicore Association. Febrero 2006.

“Achieving Software Acceleration in the Multi-core Processing Era”. CPU Technology Inc. 2005.

“Multi-core architectures”. Jernej Barbic. Mayo 2006

Artículos e informaciones de Internet

“Multi-core”

<http://en.wikipedia.org/wiki/Multicore>

“SMP. Symmetric Multiprocessing”

http://en.wikipedia.org/wiki/Symmetric_multiprocessing

“Tendencias para el año 2006, multicore”

<http://www.laflecha.net/articulos/blackhats/multicore/>

“Microsoft Won't Charge More for Multicore Licenses”

<http://it.slashdot.org/article.pl?sid=04/10/19/2257248>

“Ushering a New PC Era: The Dual-Core Computer”

<http://www.devx.com/amd/Article/26686>

“Introduction to Intel® Core™ Duo Processor Architecture”

http://www.intel.com/technology/itj/2006/volume10issue02/art01_Intro_to_Core_Duo/p05_power_control.htm

“Driving in the Fast Lane: What Multi-Core Computing Means for Programmers”

<http://www.devx.com/amd/Article/28976>

“Game Over? Core 2 Duo Knocks Out Athlon 64”

http://www.tomshardware.com/2006/07/14/core2_duo_knocks_out_athlon_64/index.html

“Intel's new 'Core' could gore AMD”

http://www.theregister.co.uk/2006/03/07/intel_core_idf/

“Intel shows off its quad core”

http://news.com.com/Intel_shows_off_its_quad_core/2100-1006_3-6038148.html

“First Benchmarks: Conroe vs. FX-62”

http://www.tomshardware.com/2006/06/05/first_benchmarks_conroe_vs_fx-62/page7.html

“Double the cores, double the heat?”

http://news.zdnet.com/2100-9584_22-6101569.html

“AMD Multi-Core: Introducing Multi-Core Technology”

<http://multicore.amd.com/en/Technology/>

“Inside Dual Core: The Deep Dive”

<http://www.devx.com/amd/Article/28524>

“Chips multicore, el salto en velocidad y rendimiento El Niagara de Sun contará con hasta 8 procesadores”

<http://www.idg.es/computerworld/impart.asp?id=163469>

“Procesadores multicore”

<http://tecnicos2.madpage.com/modules.php?name=News&file=article&sid=3>