Arquitectura de Computadores II
Clase #19

Facultad de Ingeniería
Universidad de la República

Instituto de Computación
Curso 2009

Arquitecturas de 64 bits

- Arquitecturas Intel
  - IA32
  - IA64
  - Mejora de performance: factores adicionales al procesador
  - AMD64-EMT64
- Hyper-threading
- Multicore
- Virtualización

80386

- Fundador de la familia IA32, 1985
  - Arquitectura extendida a 32 bits
    - Registros de 32 bits, espacio de direcciones de 32 bits
  - Nuevos modos de direccionamiento
  - Nuevas instrucciones
  - Soporte a paginado y gestión de memoria virtual (páginas de 4 kB)
- Modo real para ejecución de programas del 8086
- Frecuencias 20 - 40 MHz

Microarquitectura i386

- Primer procesador Intel en pipeline, 6 etapas
  - Bus Interface Unit
    - Accesa a memoria y E/S
  - Code Prefetch Unit
    - Coloca instrucciones en cola de 16 bytes
  - Instruction Decode Unit
    - Decodifica instrucciones a microcódigo
  - Execution Unit
    - Ejecuta microinstrucciones
  - Segment Unit
    - Traslación de direcciones, chequeos de protección
  - Paging Unit
    - Completa traslación de direcciones, incluye un TLB de 32 entradas, 4 vías

i486

- Introducido en 1989, 25 - 33 MHz
- Incluye el coprocesador de PF en el mismo chip
- 8 kB L1 cache on-chip
- Prefetch Unit
  - Cola de instrucciones de 32 bytes
- Instruction Decode and Execution units
  - Expandida a pipeline de 5 etapas (CPI = 1 para microinstrucciones)
- Soporte para L2 cache y multiprocesador

Pentium

- 1993, 60 - 200 MHz
  - Partición de L1 cache en 8 kB de datos y 8 kB de instrucciones
  - Segundo pipeline de ejecución (superescalar)
  - Branch predictor
  - Soporte de páginas de 4 MB
  - Modo especial para sistemas con dos procesadores
  - Data paths internos ampliados a 128 y 256 bits
- Pentium MMX
  - Modelo SIMD para ejecución paralela sobre enteros empaquetados en registros MMX de 64 bits
  - Registros de 80 bits de PF reutilizados para MMX
  - Instrucciones MMX adicionales

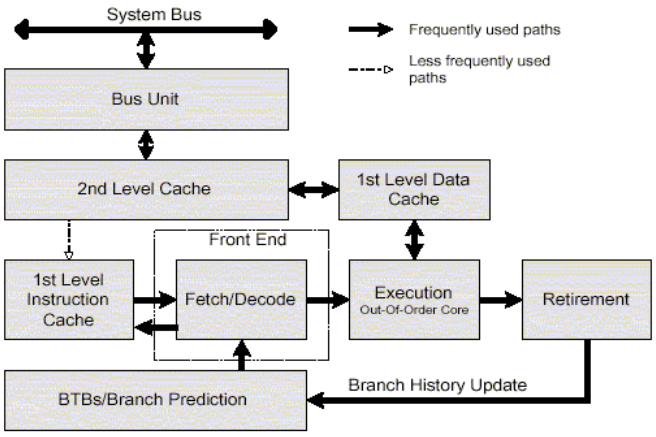
Familia P6 - 1995

- Nueva arquitectura superescalar
- Mejora de performance con el mismo proceso de manufactura
  - Mejor organización
- Procesadores de la familia P6
  - Pentium Pro, 1995, 200 MHz
  - Pentium II, 1997, 266 MHz
  - Pentium III, 1999, 500 MHz
  - Pentium III Xeon, Octubre 1999, 500 MHz

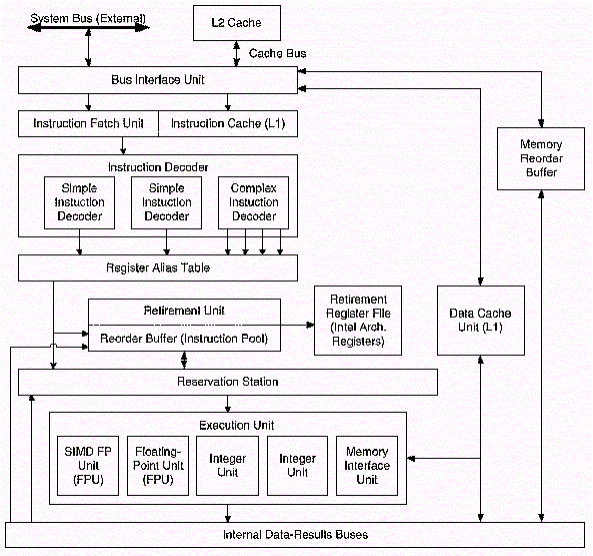
Microarquitectura P6

- Pipeline de 12 etapas dividido en 4 secciones
  - L1 & L2 caches
    - init. 8kB + 8kB with off chip L2 (256kB and 512 kB)
    - P III - 16kB + 16 kB with on chip 256kB L2
  - Front end (fetch/decode)
  - Core de ejecución out of order
    - Seis unidades de ejecución
    - Branch prediction
    - Análisis dinámico de dependencias entre instrucciones
    - Register renaming
  - Retire section
    - Búsqueda en pool de instrucciones por instrucciones completadas
    - Commit de resultados a memoria y registros de la arquitectura en orden de "issue".

Microarquitectura P6



Microarquitectura P6: detalles



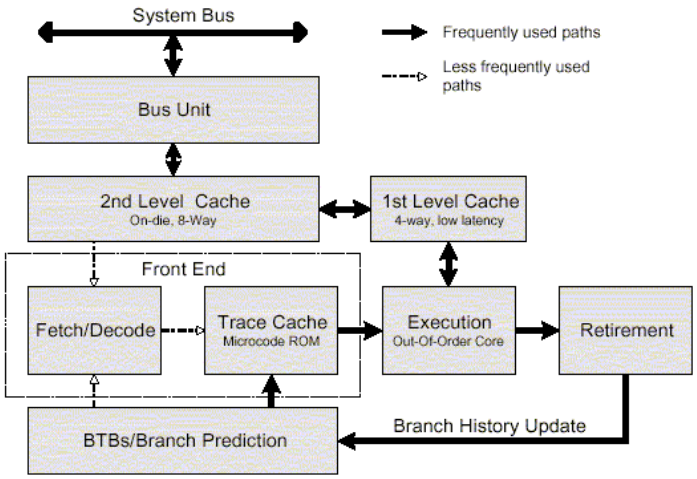
Arquitectura Pentium 4 NetBurst

- Lanzada en 2000
  - 1.5 GHz, fabricación 0.18μ, 42 millones de transistores (0.13μ y 2.4 GHz en 2002)
  - Pipeline de 20 etapas
  - Branch prediction mejorada (30% mejor que Pentium III)
  - Caches
    - Split L1 cache 20kB con bloque de 128 byte
    - L2 cache 256kB, 8-way unificada
    - Execution Trace Cache para microcódigo decodificado y "branch-predicted"
  - ALU entera
    - Al doble de frecuencia del procesador
  - Motor de ejecución out of order
    - > 100 instructions en proceso
    - Expande registros de hardware con "renaming"

Arquitectura Pentium 4 NetBurst

- Instrucciones SIMD mejoradas
  - SSE2: Streaming SIMD Extensions 2
  - 76 nuevas instrucciones SSE2
  - Mejoras a la 68 instrucciones SSE enteras existentes
  - Instrucciones SIMD pueden operar sobre enteros, PF y enteros empaquetados (MMX)
  - El propósito es reemplazar la antigua unidad de PF x87
    - Nuevos programas pueden beneficiarse de mejoras en procesamiento de PF
    - Pero... Las viejas instrucciones de PF deben ser mantenidas por compatibilidad

Microarquitectura NetBurst



Microarquitectura NetBurst

- Front end pipeline: in order
  - Fetch y prefetch normal de instrucciones
  - Decodifica instrucciones IA32 en micro-operaciones
    - Se genera microcódigo para instrucciones complejas
  - Branch prediction
  - Entrega de micro-ops decodificadas al trace cache
  - En el trace cache
    - Se construyen secuencias de micro-ops
    - Las micro-ops con destino mas probable siguen, independientemente del secuenciamiento de instrucciones
    - El trace cache se convierte en la nueva fuente de instrucciones

Microarquitectura NetBurst

- Core out of order
  - Hasta 6 micro-ops despachadas por ciclo
  - Numerosos buffers para facilitar el flujo de micro-ops
- Retirement
  - Instrucciones deben ser retiradas en el orden del programa
  - Las excepciones son lanzadas en el acto del retire
  - Una micro-op es retirada cuando se completa y escribe resultados
  - Reorder buffer
    - Almacena operaciones completadas
    - Actualiza estado de la arquitectura en orden
    - Gestiona orden de las excepciones

Resumen IA32

- Abrumador: compatibilidad hacia atrás
  - Decisiva para éxito comercial, pero ...
  - Imposibilita un diseño "elegante"
  - Registros
    - Solamente 8 registros "no tan" generales
    - Algunos compiladores no usan instrucciones con operandos implícitos, y tratan los registos disponibles como generales
  - Otros problemas legados
    - Modo real de acceso a memoria
    - Todas las instrucciones que se agregan deben permanecer en el set de instrucciones – se expande siempre..
    - Arquitectura CISC con microcódigo – procesamiento adicional del flujo de instrucciones

Qué aplicaciones se benefician de las arquitecturas de 64 bits?

- Grandes bases de datos
- Software de modelado y simulación científica y de negocios
- Software intensivo de gráficos (CAD, juegos 3-D)
- Criptografía

IA-64

- Abandona la compatibilidad hacia atrás, pero...
  - El set de instrucciones IA32 está disponible
- Anunciado en Marzo 1997 como "Merced"
  - Renombrado Itanium, disponible comercialmente desde Junio de 2001
  - Primeras versiones a 733 y 800 MHz
- Arquitectura EPIC
  - Explicitly Parallel Instruction set Computer
  - Revisión de VLIW
  - *Static scheduling*
  - Basado en *predication* y *speculation*

Entornos Operativos de IA-64

System Environment	Application Environment	Usage
IA-32	IA-32 Instruction Set	IA-32 Protected Mode, Real Mode and Virtual 8086 Mode application and operating system environment. Compatible with IA-32 Pentium®, Pentium Pro, Pentium II, and Pentium III processors.
IA-64	IA-32 Protected Mode	IA-32 Protected Mode applications in the IA-64 system environment, if supported by OS.
	IA-32 Real Mode	IA-32 Real Mode applications in the IA-64 system environment, if supported by OS.
	IA-32 Virtual Mode	IA-32 Virtual 86 Mode applications in the IA-64 system environment, if supported by OS.
	IA-64 Instruction Set	IA-64 Applications on IA-64 operating systems.

Surgimiento de IA-64

- Pentium 4: el último en la línea x86?
- Intel & Hewlett-Packard (HP) desarrollaron una nueva arquitectura
  - 64 bits
  - NO es una extensión de x86
  - NO es una adaptación de la arquitectura RISC de 64 bits de HP
- Explota gran densidad de circuitería y alta velocidad
- Uso sistemático del paralelismo
- Evolución del superscalar

Motivación

- Instruction level parallelism
  - Implícito en instrucciones de máquina
  - NO determinado por el procesador en run time
- Palabras de instrucciones largas (LIW/VLIW)
- *Branch predication* (NO es lo mismo que *branch prediction*)
- *Speculative loading*

Intel & HP lo llamaron Explicit Parallel Instruction Computing (EPIC)

- IA-64: set de instrucciones que implementa EPIC
- Itanium: primer producto EPIC de Intel

Superscalar vs. IA-64

Superscalar	IA-64
RISC-line instructions, one per word	RISC-line instructions bundled into groups of three
Multiple parallel execution units	Multiple parallel execution units
Reorders and optimizes instruction stream at run time	Reorders and optimizes instruction stream at compile time
Branch prediction with speculative execution of one path	Speculative execution along both paths of a branch
Loads data from memory only when needed, and tries to find the data in the caches first	Speculatively loads data before its needed, and still tries to find data in the caches first

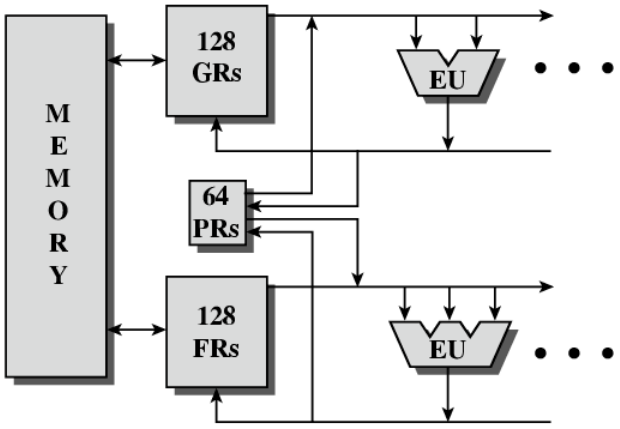
Por qué una nueva arquitectura?

- Incompatibilidad de hardware con x86
- Decenas de millones de transistores disponibles en el chip
- Se pueden construir cachés más grandes
- Más unidades de ejecución…
  - Aumentar superescalaridad
  - Gran complejidad
  - Más lógica para orquestar el funcionamiento
  - Se necesita mejorar *branch prediction…*
  - …y pipelines más largos
  - Mayor penalización por *misprediction*
  - Mayor cantidad de *renaming registers* requeridos

Paralelismo Explícito

- *Instruction parallelism* despachado en tiempo de compilación
  - Incluído en instrucción de máquina
- El procesador utiliza esta información para la ejecución en paralelo
- Requiere circuitería menos compleja
- El compilador tiene "todo el tiempo del mundo" para determinar posibles operaciones paralelizables
  - Visión global del programa

Organización General



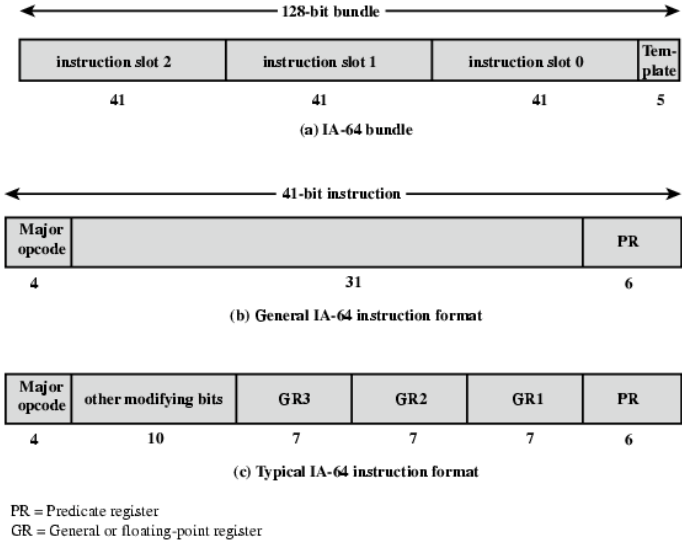
Carácterísticas clave

- Gran cantidad de registros
  - Formato de instrucción de IA-64 asume 256
    - 128 * 64 bits ints. enteras, lógicas y de propósito general
    - 128 * 82 bits ints. punto flotante y gráficos
  - 64 * 1 bit *predicated execution registers*
- Múltiples unidades de ejecución
  - 8 o más…
  - Dependiente de cantidad de transistores disponibles
  - La ejecución de instrucciones en paralelo depende del hardware disponible
    - Ej: 8 instrucciones paralelas pueden ser ordenadas en dos grupos sii hay 4 unidades de ejecución disponibles

IA-64 Execution Units

- I-Unit
  - Integer arithmetic
  - Shift and add
  - Logical
  - Compare
  - Integer multimedia ops
- M-Unit
  - Load and store
    - Between register and memory
  - Some integer ALU
- B-Unit
  - Branch instructions
- F-Unit
  - Floating point instructions

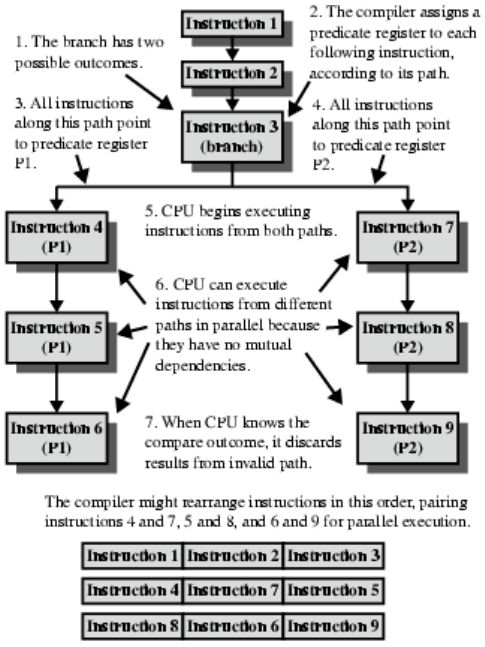
Formato de Instrucciones



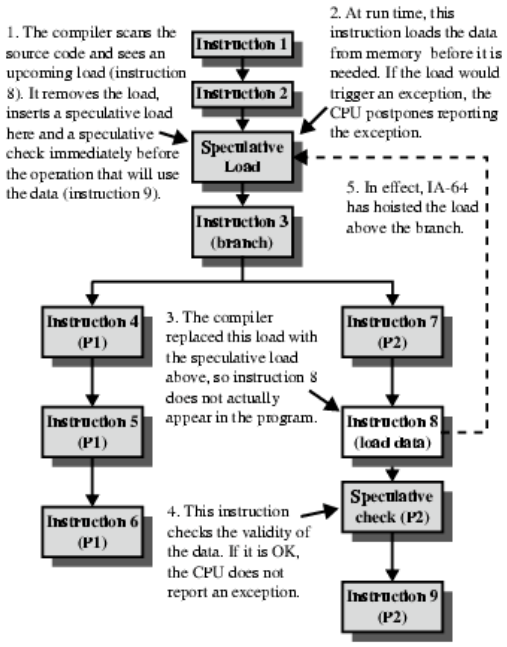
Formato de Instrucciones

- *Bundle* de 128 bits
  - Contiene 3 instrucciones (*syllables*) más un template
  - Fetch de uno o más bundles cada vez
  - El template contiene información acerca de las instrucciones que pueden ejecutarse en paralelo
    - No hay confinamiento a un solo bundle
    - Ej. un flujo de 8 instructiones puede ser ejecutado en paralelo
    - El compilador deberá reordenar instrucciones para formar bundles contiguos
    - Se pueden mezclar instrucciones dependientes e independientes en el mismo bundle
  - Instrucción de 41 bits
    - Más registros que RISC usual
    - *Predicated execution registers*

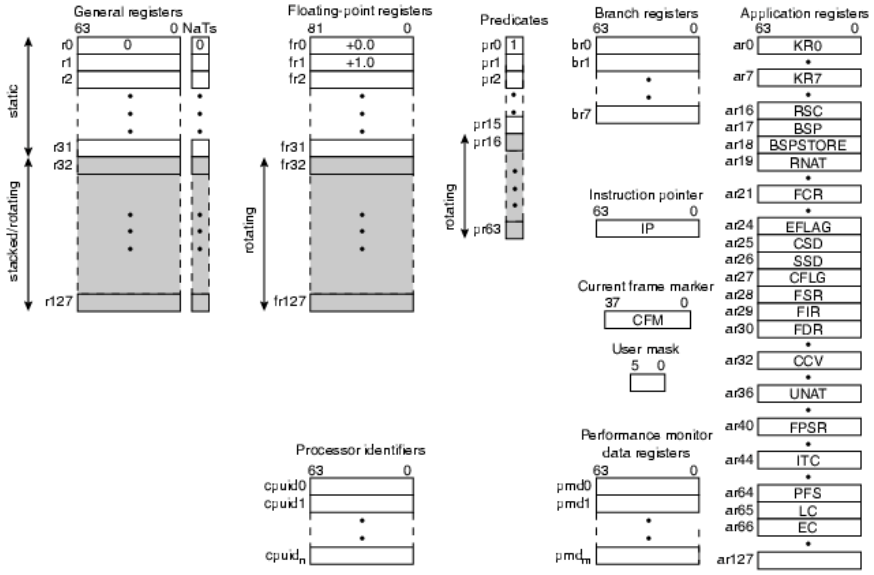
Predication



Speculative Loading



Registros de IA-64



Registros de IA-64

- General Registers
  - 128 registros de propósito general de 64 bits
  - r0-r31 estáticos
    - referencias interpretadas literalmente
  - r32-r127 pueden ser usados rotativamente para pipeline de software o stack de registros
    - Referencias virtuales
    - El hardware los puede renombrar dinámicamente
- Floating Point Registers
  - 128 registros de punto flotante de 82 bits
  - Formato IEEE 745 double extended
  - fr0-fr31 estáticos, fr32-fr127 rotativos
- Predicate registers
  - 64 registros de 1 bit usados como predicados
  - pr0 siempre 1 para permitir instrucciones NO predicadas
  - pr1-pr15 estáticos, pr16-pr63 pueden rotar

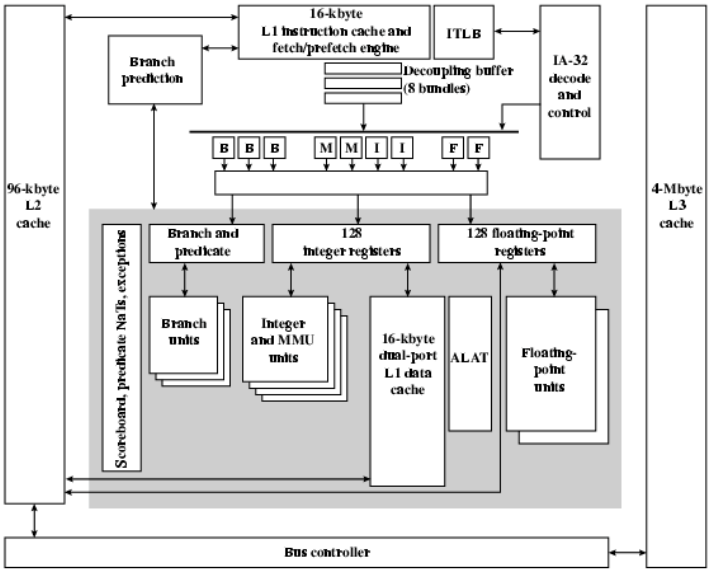
Registros de IA-64

- Branch registers
  - 8 registros de 64 bits
- Instruction pointer
  - Dirección del "bundle" en ejecución
- Current frame marker
  - Información de estado del marco de registros de stack actual
  - User mask
    - Conjunto de valores de bit
    - Traps de alineamiento, monitores de performance, monitores de uso de registros de PF
- Performance monitoring data registers
  - Soporte para hardware monitor de performance
- Application registers
  - Registros de uso específico

Organización del Itanium

- Carácterísticas superescalares
  - Ancho x6, profundidad x10 pasos de pipeline
  - Prefetch dinámico
  - Branch prediction
  - *Scoreboard* de registros para optimizar indeterminaciones en tiempo de compilación
- Carácterísticas EPIC
  - Soporte en hardware para *predicated execution*
  - *Control & data speculation*
  - *Software pipelining*

Diagrama del Itanium



Transiciones del Set de Instrucciones

- Puede ejecutar instrucciones IA32 e IA64 en cualquier momento
  - Instrucciones especiales e interrupciones para habilitar la transición
    - Cambiar hacia IA32
    - Cambiar hacia IA64
    - Las interrupciones son manejadas por el set IA64
    - RFI Return From Interrupt puede retornar a cualquiera de los sets

Set Instrucciones IA-64

- Arquitectura load/store
- Comunicación compilador-procesador
  - Instrucciones load/store con *2-bit cache hint*
  - *Static prediction hint for branches*
- Especulación
  - Control y datos
    - El compilador puede introducir instrucciones antes de tiempo especulativamente
  - *Control speculation*
    - Instrucciones después de un branch ejecutadas antes del mismo
    - Chequeo de excepciones en lugar original
  - *Data speculation*
    - Load adelantado con instrucciones de chequeo para determinar posibles superposiciones con almacenamientos posteriores

Resumen IA-64

- Arquitectura RISC de 64 bits
- *Scheduling* de instrucciones realizado por el compilador
  - Static scheduling
  - Los mecanismos del hardware están expuestos, por eso el nombre Explicit Parallel Instruction Computing EPIC
- Para aprovechar el hardware disponible…
  - Las aplicaciones *legacy* deben ser recompiladas
  - El compilador es el elemento crucial en la producción de código eficiente

Mejora de Performance: factores adicionales a los 64 bits…

- Set de instrucciones nuevos y mejorados
- Más funcionalidades "on-chip"
- Caches más grandes
- Incremento de ancho de banda E/S y memoria
- Mejoras en la organización
- Las CPUs son cada vez más veloces…
  - …y el resto del sistema?
- La performance global del sistema depende de muchos más factores que la CPU
- Hypertransport, Hyper-Threading, Multicore, Virtualización?

Arquitectura AMD64

- Puente entre 32 y 64 bits
- Diseño de procesadores para servidores, workstation, y computadores personales
- Agrega capacidad de 64 bits a "cores" de 32 bits para servidores de 2 y 4 procesadores
- Protección de la inversión para usuarios de aplicaciones de 32 bits
- Actuales aplicaciones de 32 bits funcionan en los sistemas operativos de 32 bits y 64 bits
  - Windows XP, Windows 2000
  - Windows (with AMD64 compatibility)
  - Windows Server 2003
  - Linux x86, Linux x86_64
- Más allá de los 64 bits: mejoras en la conexión del procesador con la memoria y la E/S

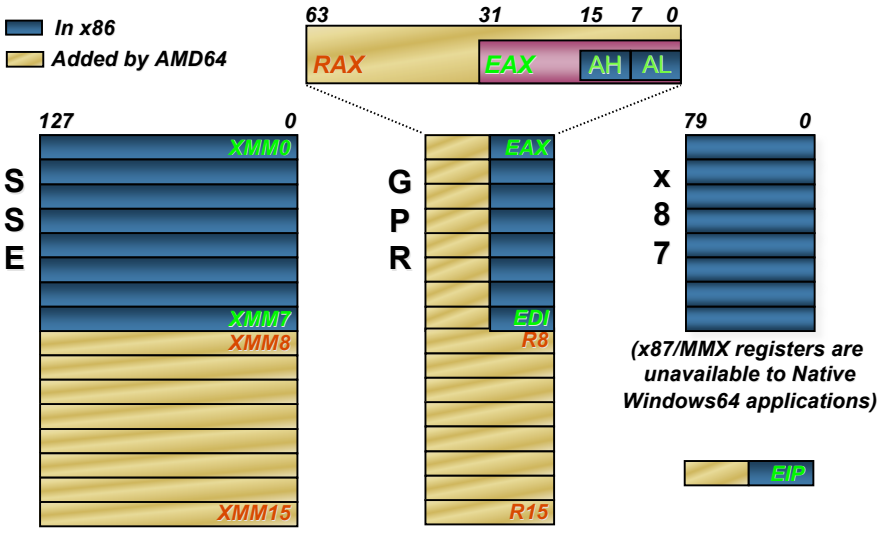
Arquitectura AMD64

- Extensión de arquitectura x86 a 64 bits
  - Soporte de 32 y 64 bits
- Nuevas instrucciones de 64 bits
- Registros de propósito general: 16 vs. 8
- Pipeline de 12 etapas
- Soporte para decodificar tres instrucciones por ciclo
- Ejecución fuera de orden
- Paralelismo explícito en hardware

Arquitectura AMD64

- BIOS: código estándar x86 de 32 bits
  - Transferencia a modo de 64 bits bajo el control del OS
- Legacy Mode
  - Los procesadores AMD64 corren aplicaciones de 32 bits con mejora de performance
  - Compatibilidad completa con sistemas de 32 bits
- Compatibility Mode bajo un OS de 64 bits
  - Aplicaciones de 32 bits corren bajo el control del OS de 64 bits
  - Compatibilidad x86 "at full speed"
  - No se necesita recompilar, no hay capa de emulación
  - OS provee la adaptación al nivel de llamadas al sistema
- 64-bit Mode bajo un OS de 64 bits
  - Algunas aplicaciones pueden ser reescritas/portadas: "full 64-bit"
  - Incluso aplicaciones que no necesitan direccionamiento de 64 bits pueden mejorar su performance si se recompilan en un entorno de 64 bits

AMD64: Modelo de Programación

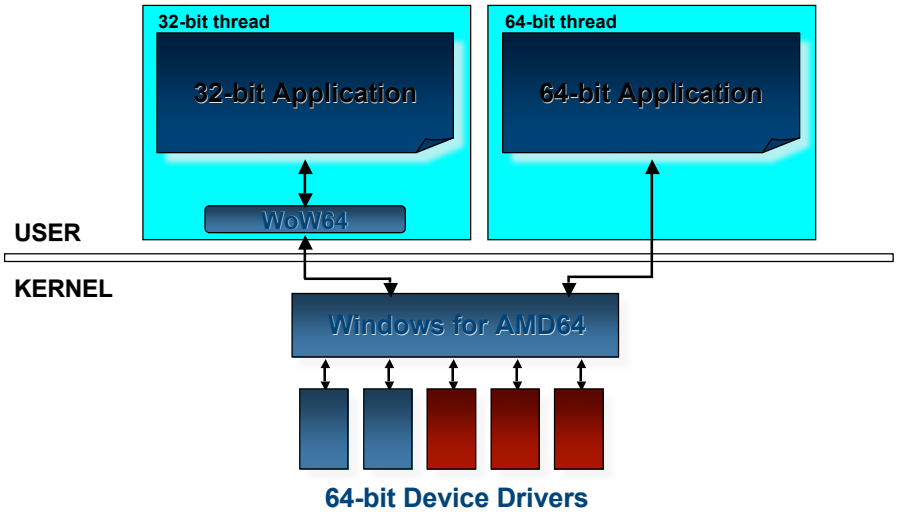


AMD64: modos de operación

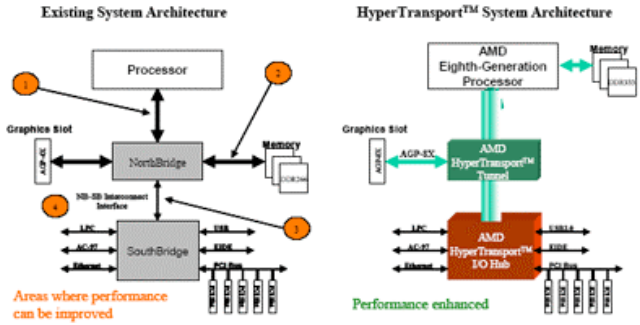
Mode		Operating System Required	Application Recompile Required	Defaults[1]			
				Address Size (bits)	Operand Size (bits)	Register Extensions[2]	GPR Width (bits)
Long Mode[3]	64-Bit Mode	New 64-bit OS	yes	64	32	yes	64
	Compatibility Mode		no	32		no	32
				16			
Legacy Mode[4]		Legacy 32-bit or 16-bit OS	no	32	32	no	32
				16	16		

1. Defaults can be overridden in most modes using an instruction prefix or system control bit.
2. Register extensions includes eight new GPRs and eight new XMM registers (also called SSE registers).
3. Long mode supports only x86 protected mode. It does not support x86 real mode or virtual-8086 mode. Also, it does not support task switching.
4. Legacy mode supports x86 real mode, virtual-8086 mode, and protected mode.

Ejemplo: Windows on Windows 64



Mejoras al chipset



HyperTransport

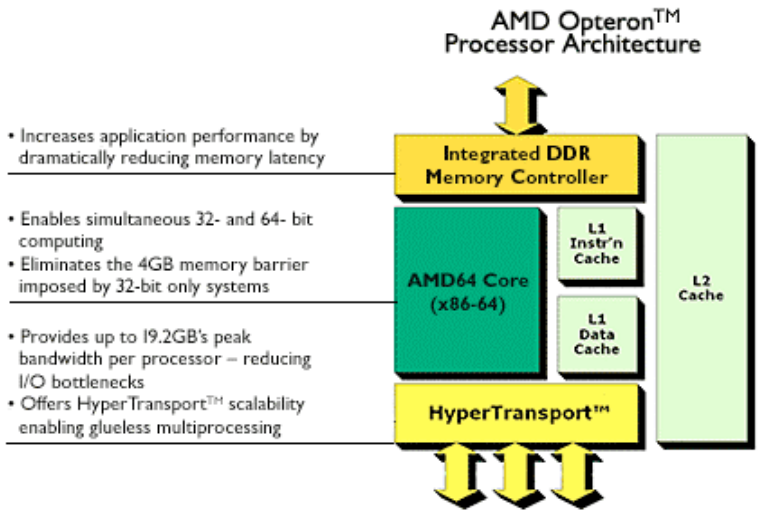
- Reemplazo del bus tradicional
- Enlaces punto a punto
- Mejora ancho de banda
- Permite superar cuellos de botella, y mejora la performance del sistema
- Hasta 1600 MHz, 9.6 GB/s por link

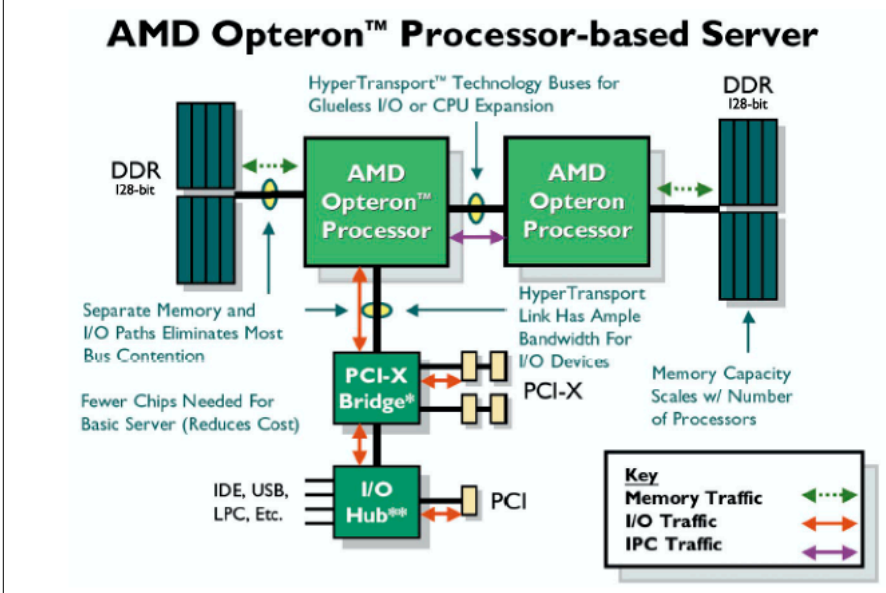
HyperTransport™ CONSORTIUM

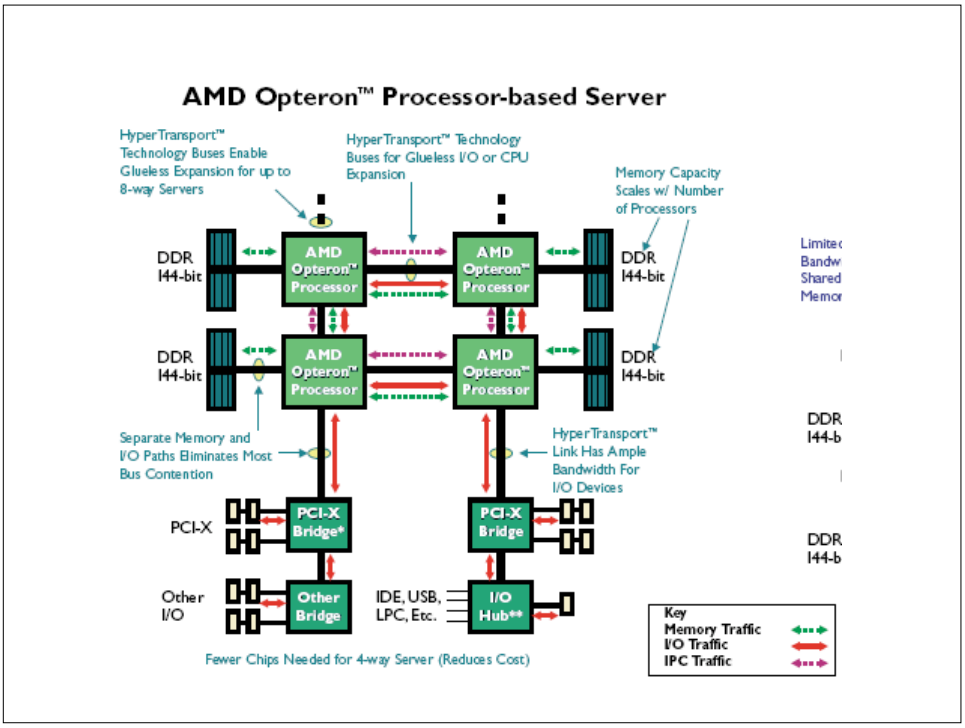
Controlador de memoria DDR integrado

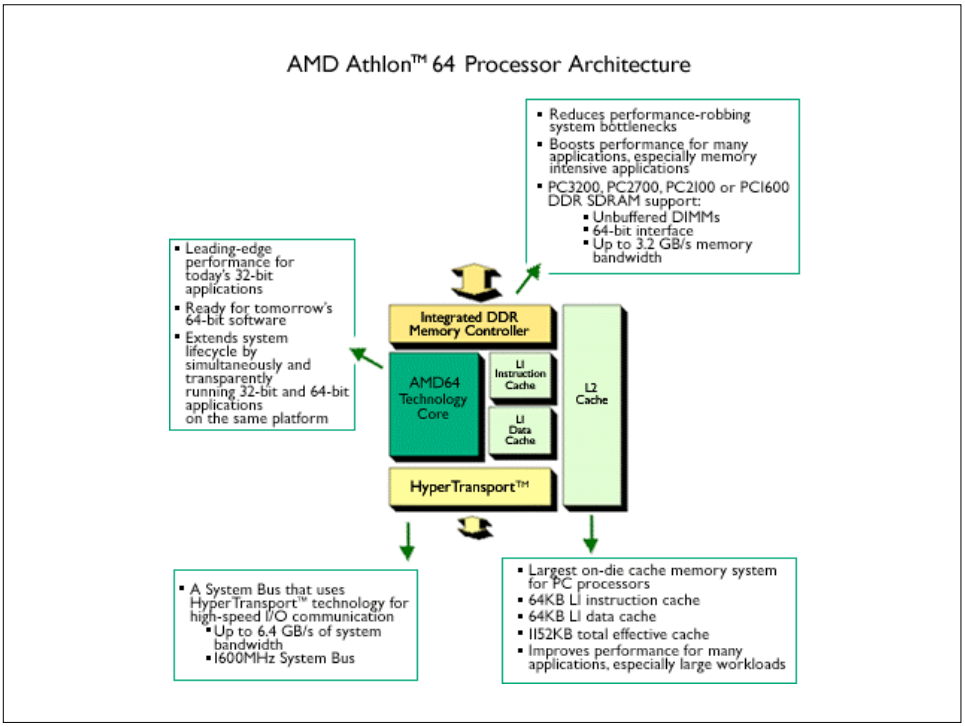
- Control directo desde la CPU
- Conexión punto a punto entre el procesador y la memoria, no hay "hubs" intermedios
- Mejor ancho de banda de memoria, menos cuellos de botella

AMD Opteron




AMD Opteron™ Processor-based Server
HyperTransport™ Technology Buses for Glueless I/O or CPU Expansion
DDR 128-bit
DDR 128-bit
AMD Opteron™ Processor
AMD Opteron Processor
Separate Memory and I/O Paths Eliminates Most Bus Contention
HyperTransport Link Has Ample Bandwidth For I/O Devices
PCI-X Bridge*
PCI-X
Fewer Chips Needed For Basic Server (Reduces Cost)
Memory Capacity Scales w/ Number of Processors
IDE, USB, LPC, Etc.
I/O Hub**
PCI
Key
Memory Traffic
I/O Traffic
IPC Traffic

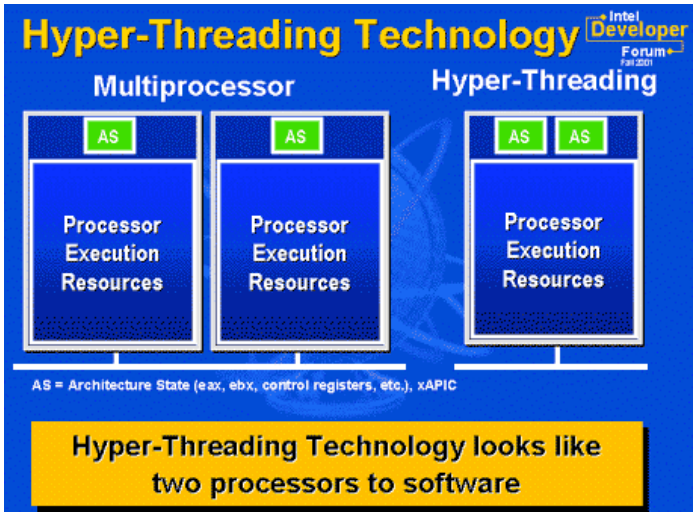

AMD Opteron™ Processor-based Server
HyperTransport™ Technology Buses Enable Glueless Expansion for up to 8-way Servers
HyperTransport™ Technology Buses for Glueless I/O or CPU Expansion
Memory Capacity Scales w/ Number of Processors
DDR 144-bit
AMD Opteron™ Processor
Separate Memory and I/O Paths Eliminates Most Bus Contention
HyperTransport™ Link Has Ample Bandwidth For I/O Devices
PCI-X
PCI-X Bridge*
Other I/O
Other Bridge
IDE, USB, LPC, Etc.
I/O Hub**
Fewer Chips Needed for 4-way Server (Reduces Cost)
Key
Memory Traffic
I/O Traffic
IPC Traffic



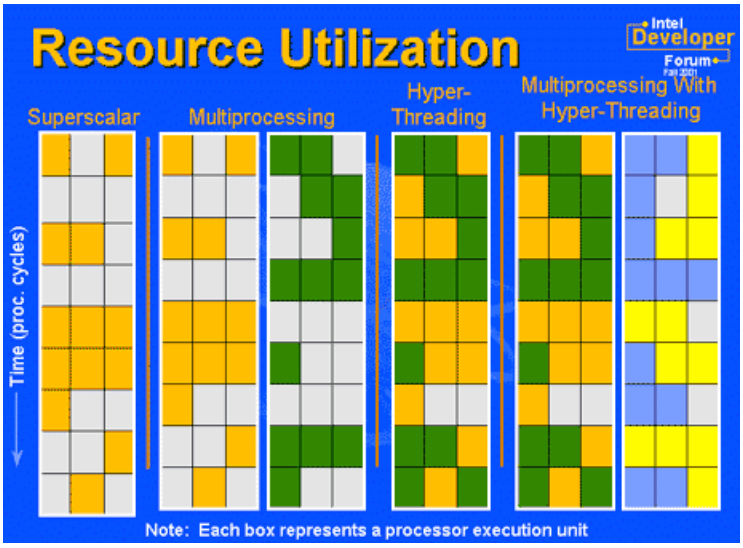
Hyper-Threading

- Capacidad de procesar dos threads simultáneamente
- Interfaz de dos procesadores, aunque físicamente hay uno solo
- Potencial incremento de la eficiencia de la CPU
  - El software debe preocuparse de usar esta potencialidad
- Procesadores
  - Pentium 4 HT
  - Xeon HT
- Críticas
  - Mayor consumo?
  - Chache trashing?

Hyper-Threading

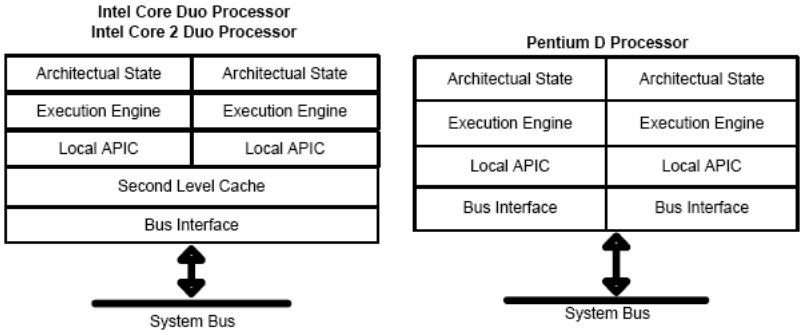


Hyper-Threading

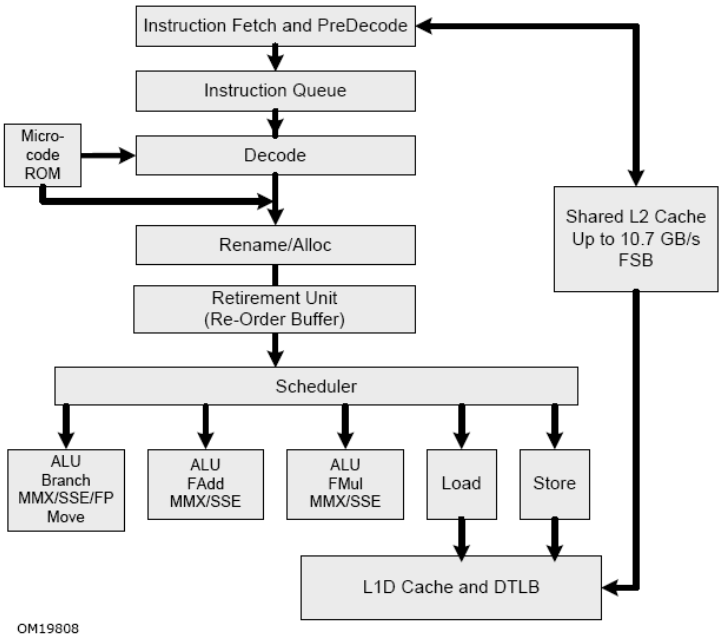


Procesadores Multi-Core

- Intel Pentium M, Pentium D, Core, Core 2, Quad…
  - Evolución de microarquitectura Netburst a Core
  - Netburst superada
- AMD Athlon 64 X2 Dual-Core



Microarquitectura Core



Máquina Virtual

- Una máquina virtual (MV) es un duplicado de una máquina real, eficiente y aislado

Características

- Duplicado: La MV se debería comportar de forma idéntica a la máquina real, excepto por:
  - La existencia de menos recursos disponibles (incluso diferentes entre ejecuciones)
  - Diferencias de temporización al tratar con dispositivos
- Aislado: Se pueden ejecutar varias MV sin interferencias
- Eficiente: La MV debería ejecutarse a una velocidad cercana a la del HW real
  - Requiere que la mayoría de las instrucciones se ejecuten directamente por el HW

Tipos de máquinas virtuales

- Máquinas virtuales por proceso
  - Ejemplos: Java, .NET Framework
  - Máquina virtual instanciada para un proceso
  - Cuando termina el proceso, termina la instancia de máquina virtual
- Máquina virtual por sistema
  - Virtualización ISA (Instruction Set Architecture)
  - Ofrecen un entorno de ejecución completo

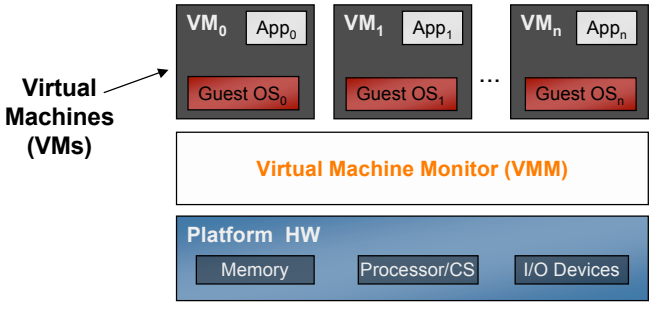
Algo de historia

- Idea bastante utilizada hasta los 70-80 en mainframes
  - Cae en desuso con el paso a computadores más pequeños
- Renace en esta década
  - Seguridad
  - Vista uniforme de Hardware
  - Encapsulación
    - Replicación, checkpointing y reinicio, depuración, ...
- Esto se parece mucho a lo que hace el SO…

Monitor de Máquina Virtual (MMV)

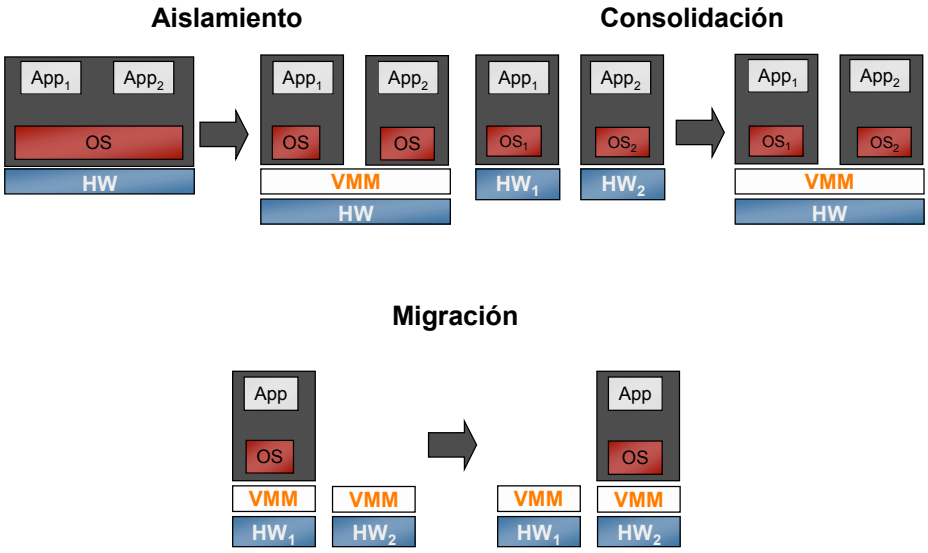
- Programa que corre sobre el hardware real para implementar la máquina virtual
- Control de recursos y planificación de huéspedes
- Implicancias:
  - MMV necesita ejecutarse en modo supervisor
  - Software huésped en modo usuario
  - Instrucciones privilegiadas de huéspedes implican traps
  - MMV interpreta/emula instrucciones privilegiadas

Virtual Machine Monitors (VMMs)



- VMM es una capa de software del sistema
  - Permite que múltiples VMs compartan la platforma de hardware
  - Las aplicaciones corren sin modificaciones

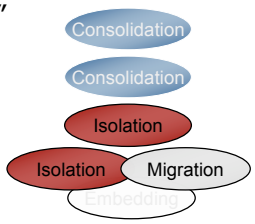
Virtualización: para que sirve?



Virtualización: Modelos de Uso

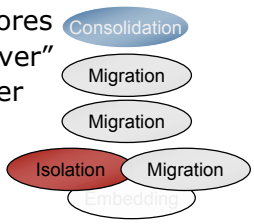
CLIENTE

- Soporte de SW "legado"
- Entrenamiento
- Partición de actividades
- Gestionabilidad
- …

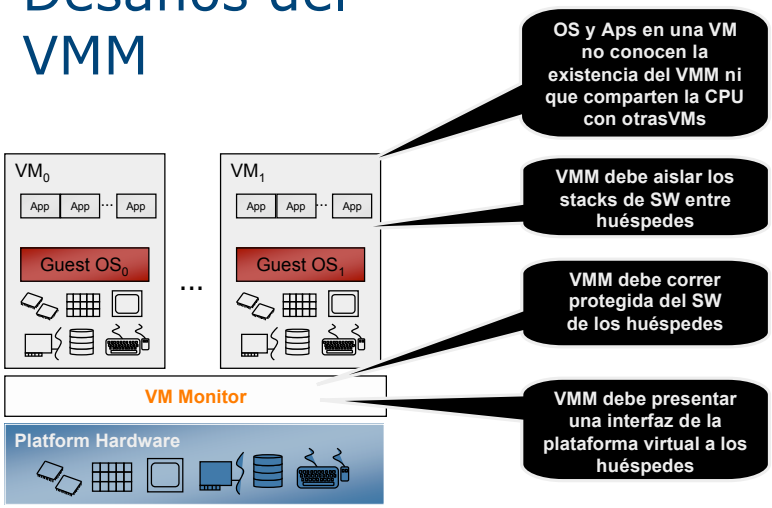


SERVIDOR

- Consolidación de Servidores
- Infraestructura de "failover"
- Flexibilidad de Datacenter
- Gestionabilidad
- …



Desafíos del VMM

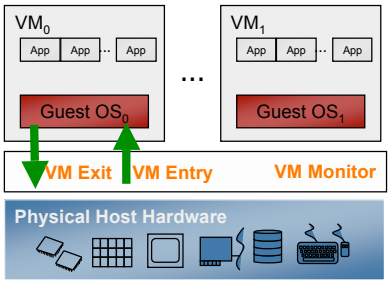


Modos de operación

- VMX root
  - Totalmente privilegiado
  - Pensado para VMM
- VMX non-root
  - No privilegiado
  - Pensado para SW huésped

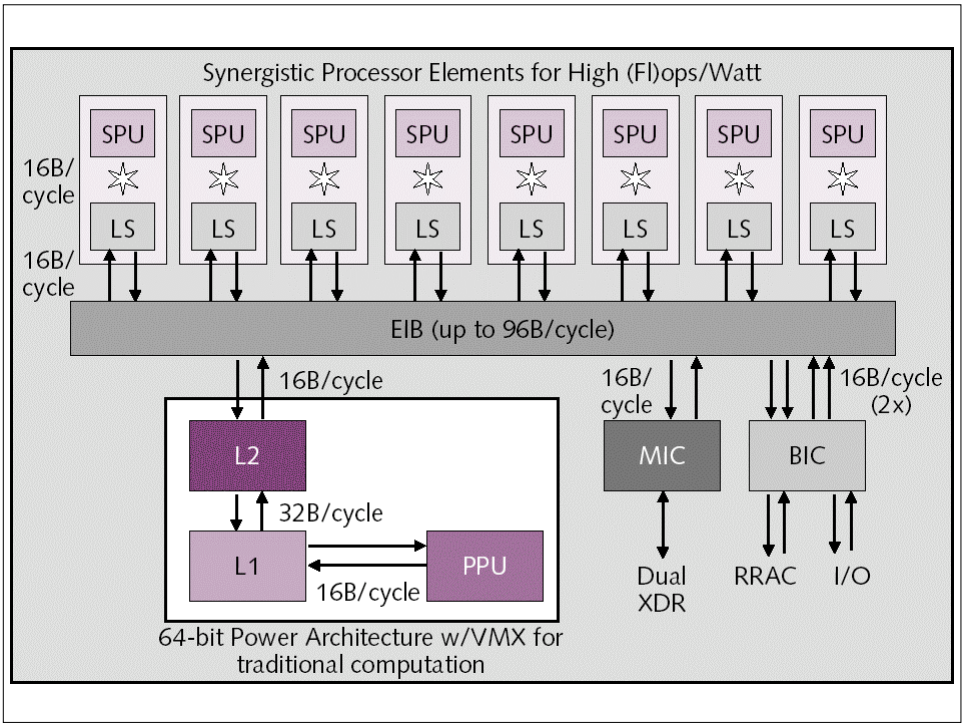
Entrada y Salida de VMs

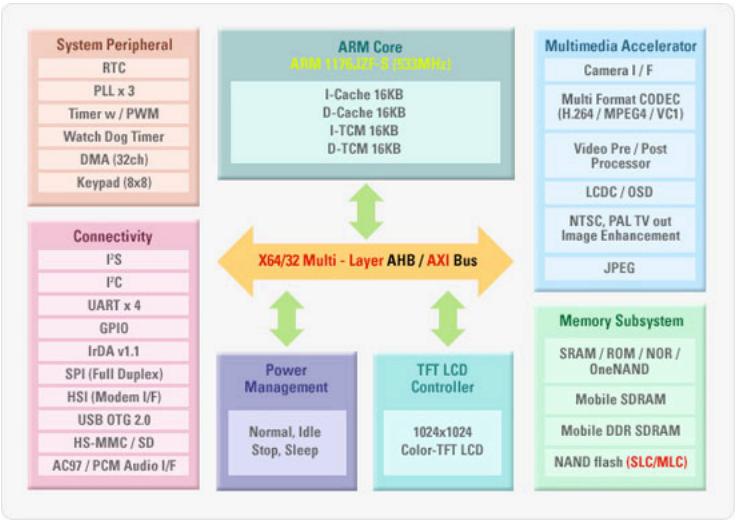
- VM Entry
  - Transición de VMM a Huésped
  - Entra en modo non-root
  - Carga el estado del huesped
  - VMLAUNCH instrucción usada en entrada incial
    VMRESUME instrucción usada en llamadas siguientes
- VM Exit
  - VMEXIT instrucción usada para pasar a VMM
  - Entra en modo root
  - Salva el estado del huesped
  - Carga el estado de VMM



El futuro llegó…hace rato

- Cell processor
  - PS3
- ARM processor
  - iPhone
- Arquitectura Multi-Core No-Homogénea
- Evolución de Multi-Core, EPIC (o VLIW) y Superescalares


Synergistic Processor Elements for High (Fl)ops/Watt
SPU
LS
16B/ cycle
16B/ cycle
EIB (up to 96B/cycle)
16B/cycle
L2
32B/cycle
L1
PPU
16B/cycle
64-bit Power Architecture w/VMX for traditional computation
16B/ cycle
MIC
Dual XDR
16B/cycle (2x)
BIC
RRAC
I/O


System Peripheral
RTC
PLL x 3
Timer w / PWM
Watch Dog Timer
DMA (32ch)
Keypad (8x8)
ARM Core
I-Cache 16KB
D-Cache 16KB
I-TCM 16KB
D-TCM 16KB
Multimedia Accelerator
Camera I / F
Multi Format CODEC (H.264 / MPEG4 / VC1)
Video Pre / Post Processor
LCDC / OSD
NTSC, PAL TV out Image Enhancement
JPEG
Connectivity
I²S
I²C
UART x 4
GPIO
IrDA v1.1
SPI (Full Duplex)
HSI (Modem I/F)
USB OTG 2.0
HS-MMC / SD
AC97 / PCM Audio I/F
X64/32 Multi - Layer AHB / AXI Bus
Power Management
Normal, Idle Stop, Sleep
TFT LCD Controller
1024x1024 Color-TFT LCD
Memory Subsystem
SRAM / ROM / NOR / OneNAND
Mobile SDRAM
Mobile DDR SDRAM
NAND flash (SLC/MLC)