Intel® Processor Architecture

January 2013

Software & Services Group, Developer Products Division

Agenda

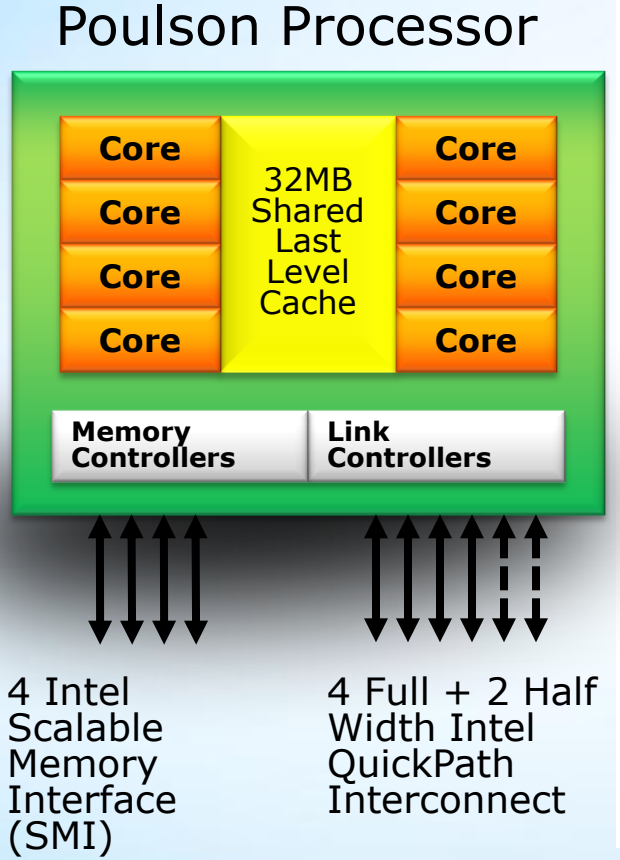
- Overview Intel® processor architecture
- Intel x86 ISA (instruction set architecture)
- Micro-architecture of processor core
- Uncore structure
- Additional processor features
  - Hyper-threading
  - Turbo mode
- Summary

Intel® Processor Segments Today

Architecture	Target Platforms	ISA	Specific Features
Intel® ATOM™ Architecture	phone, tablet, netbook, low-power server	x86 up to SSSE-3, 32 and 64 bit	optimized for low-power, in-order
Intel® Core™ Architecture	mainstream notebook, desktop, server	x86 up to Intel® AVX, 32 and 64bit	flexible feature set covering all needs
Intel® Itanium® Architecture	high end server	IA64, x86 by emulation	RAS, large address space
Intel® MIC Architecture	accelerator for HPC	x86 and Intel® MIC Instruction Set	+60 cores, optimized for Floating-Point performance

Itanium® 9500 (Poulson)

New Itanium Processor

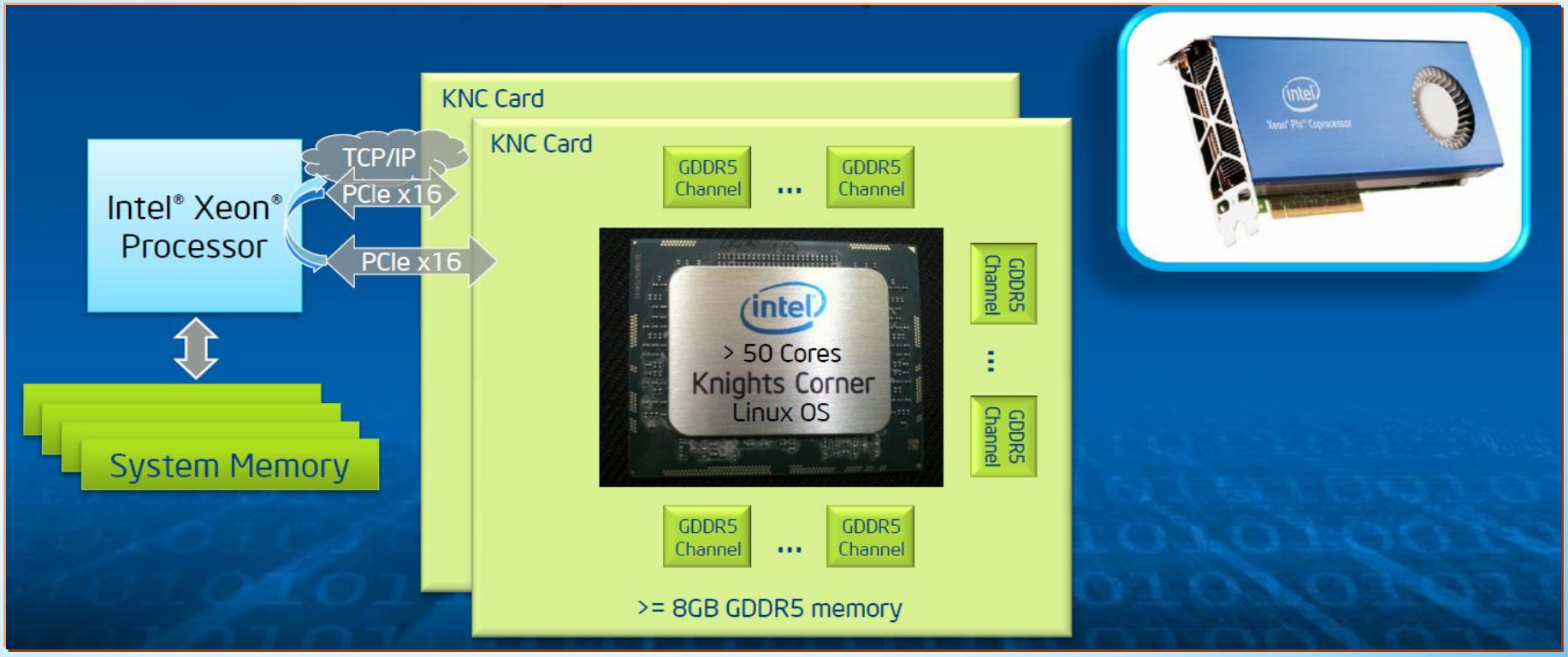


- *Compatible with Itanium® 9300 processors (Tukwila)*
- *New micro-architecture with 8 Cores*
- *54 MB on-die cache*
- *Improved RAS and power management capabilities*
- *Doubles execution width from 6 to 12 instructions/cycle*
- *32nm process technology*
- *Launched in November 2012*

Compatibility provides protection for today's Itanium® investment

Intel® XEON™ Phi

Former Code Name "*Knights Corner*"



Intel® XEON™ Phi - The first product implementation of the Intel® Many Integrated Core Architecture (Intel® MIC)

Agenda

- Overview Intel® processor architecture
- Intel x86 ISA (instruction set architecture)
- Micro-architecture of processor core
- Uncore structure
- Additional processor features
  - Hyper-threading
  - Turbo mode
- Summary

X86: From Smartphones to …
Motorola RAZR* i

- Launched September 2012
- RAZR i is the first smartphone that can achieve speeds of 2.0 GHz

Processor:

Intel® ATOM™ Z2460

X86: … to Supercomputers
LRZ SuperMUC System

- Installed summer 2012
  - Most powerful x86-architecture based computer
  - #6 on Top500 list
  - More than 150000 cores
- Processor:
  - Intel® Xeon® E5-2680 (“Sandy Bridge”)
  - Intel® Xeon® E7-4870 (“Westmere”)

Intel Tick-Tock Roadmap for Mainstream x86 Architecture since 2006

Intel® Core™ MicroArchitecture		Micro Architecture Codename "Nehalem"		2nd Generation Intel® Core™ Micro Architecture	3nd Generation Intel® Core™ Micro Architecture
Merom	**Penryn**	**Nehalem**	**Westmere**	**Sandy Bridge**	**Ivy Bridge**
NEW Micro architecture	NEW Process Technology	NEW Micro architecture	NEW Process Technology	NEW Micro architecture	NEW Process Technology
65nm	**45nm**	**45nm**	**32nm**	**32nm**	**22nm**
TOCK	***TICK***	***TOCK***	***TICK***	***TOCK***	***TICK***
2006	2007	2008	2009	2011	2012
SSSE-3	SSE4.1	SSE4.2	AES	AVX	7 new instructions

TICK + TOCK = SHRINK + INNOVATE

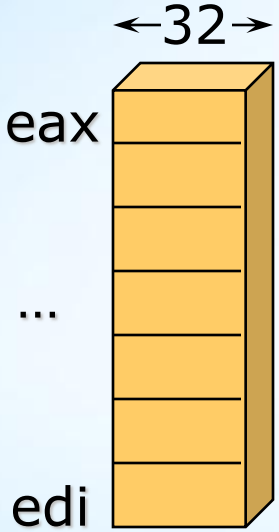
To be continued …

4nth Generation Intel® Core™ Micro Architecture	TBD	TBD	TBD	TBD	TBD
Haswell NEW Micro architecture **22nm**	**Broadwell** NEW Process Technology **14nm**	**TBD** NEW Micro architecture **14nm**	**TBD** NEW Process Technology **10nm**	**TBD** NEW Micro architecture **10nm**	**TBD** NEW Process Technology **7nm**
TICK	***TOCK***	***TICK***	***TOCK***	***TICK***	***TOCK***
2013 AVX-2	>= 2014	???	???	???	???

TICK + TOCK = SHRINK + INNOVATE

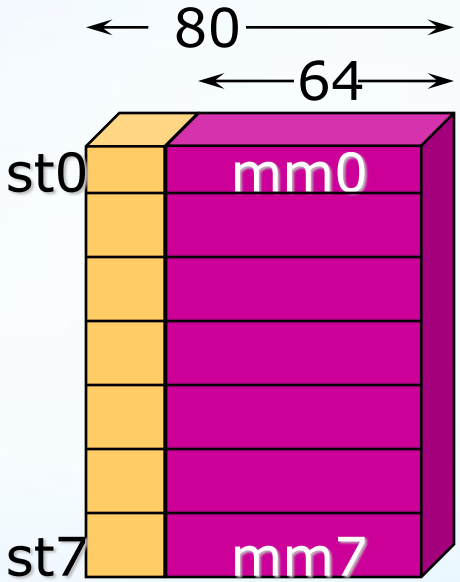
Registers State for Intel® Pentium® 3 Processor (1998)

IA32-INT Registers



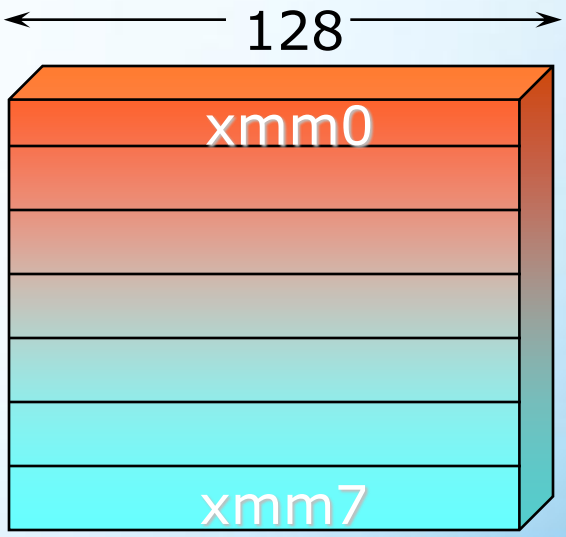
Fourteen 32-bit registers
Scalar data & addresses
Direct access to regs

MMX Technology / IA-FP Registers



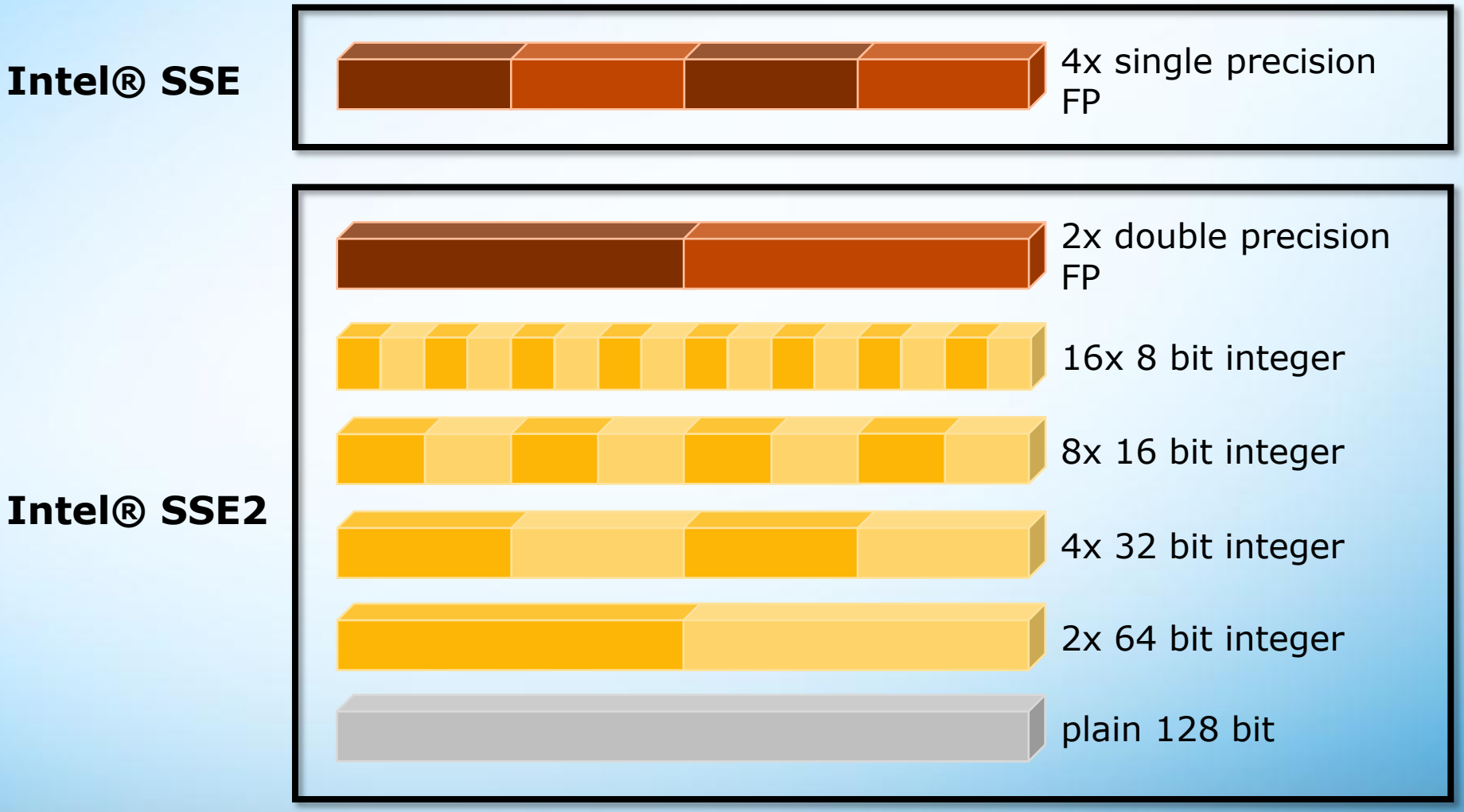
Eight 80/64-bit registers
Hold data only
Direct access to MM0..MM7
No MMX™ Technology / FP interoperability

SSE Registers



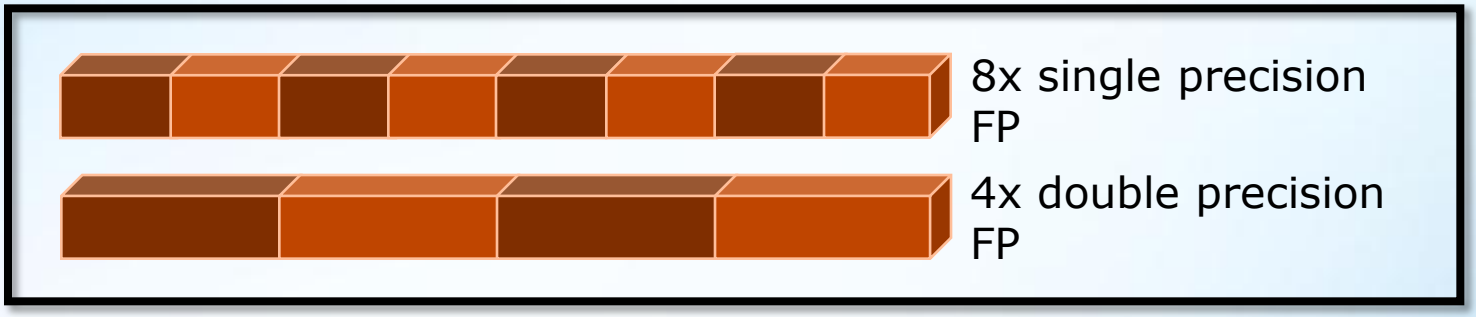
Eight 128-bit registers
Hold data only:
- 4 x single FP numbers
- 2 x double FP numbers
- 128-bit packed integers

SSE Vector Types

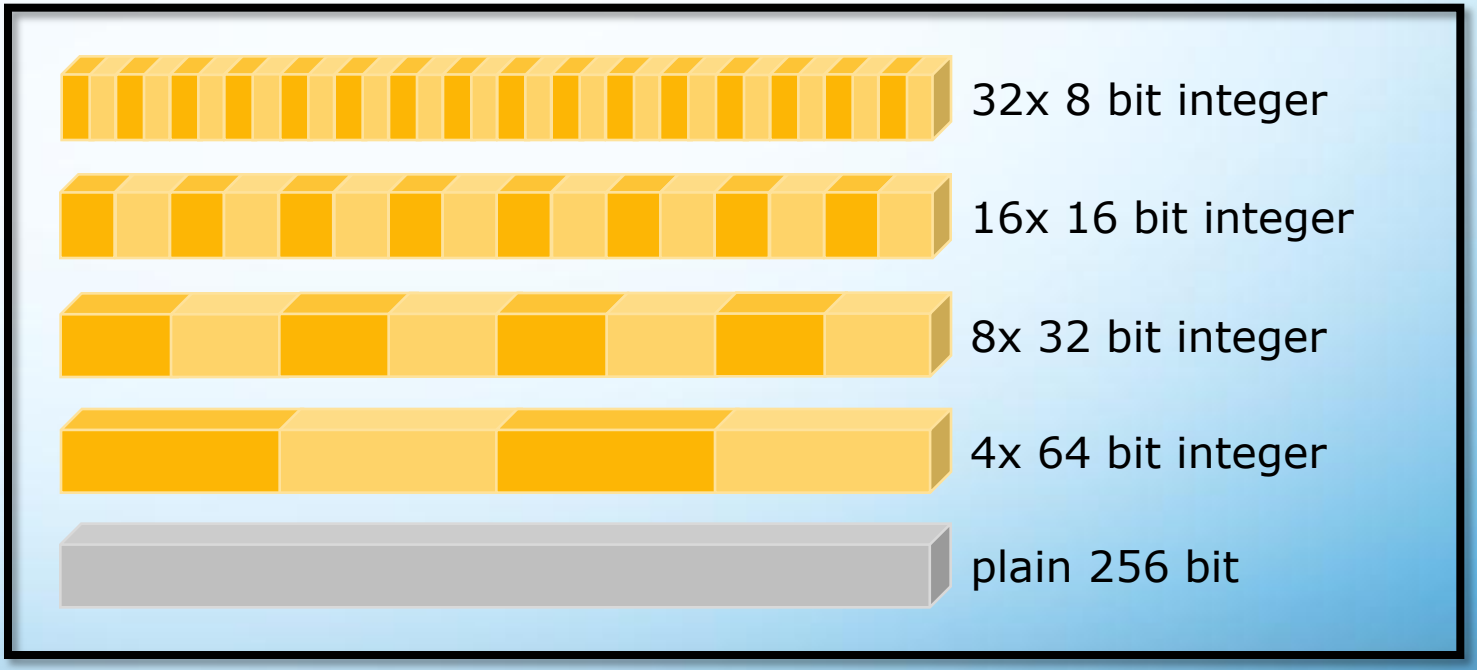


AVX Vector Types

Intel® AVX



Intel® AVX2 (Future)



X86 ISA: The Instruction Set

- The instruction set for the x86 architecture has been extended numerous times since the set supported by the 8086 processor
  - See http://en.wikipedia.org/wiki/X86_instruction_listings for an excellent overview
- Today, the "base" instructions set ("IA32 ISA") is the one supported by the first 32bit processor - 80386
- Multiple, "smaller" extensions added then before SSE (1998 / Intel® Pentium® 3) like
  - MMX(64 bit SIMD using the x87 FP registers)
  - Conditional move
  - Atomic exchange

New Instructions in Haswell (2013)

Group		Description	Count *
AVX2	SIMD Integer Instructions promoted to 256 bits	Adding vector integer operations to 256-bit	170 / 124
	Gather	Load elements from vector of indices vectorization enabler	
	Shuffling / Data Rearrangement	Blend, element shift and permute instructions	
FMA		Fused Multiply-Add operation forms (FMA-3)	96 / 60
Bit Manipulation and Cryptography		Improving performance of bit stream manipulation and decode, large integer arithmetic and hashes	15 / 15
TSX=RTM+HLE		Transactional Memory	4/4
Others		MOVBE: Load and Store of Big Endian forms INVPCID: Invalidate processor context ID	2 / 2

* Total instructions / different mnemonics

HSW Improvements for Threading
Sample Code Computing PI by Windows Threads

```
#include <windows.h>
#define NUM_THREADS 2
HANDLE thread_handles[NUM_THREADS];
CRITICAL_SECTION hUpdateMutex;
static long num_steps = 100000;
double step;
double global_sum = 0.0;

void Pi (void *arg)
{
   int i, start;
  double x, sum = 0.0;


  start = *(int *) arg;
  step = 1.0/(double) num_steps;

  for (i=start;i<= num_steps;
                        i=i+NUM_THREADS){
      x = (i-0.5)*step;
     sum = sum + 4.0/(1.0+x*x);
  }
  EnterCriticalSection(&hUpdateMutex);
  global_sum += sum;
  LeaveCriticalSection(&hUpdateMutex);
}
```

```
void main ()
{
  double pi; int i;
  DWORD threadID;
  int threadArg[NUM_THREADS];

  for(i=0; i<NUM_THREADS; i++)
            threadArg[i] = i+1;

  InitializeCriticalSection(&hUpdateMutex);

  for (i=0; i<NUM_THREADS; i++){
            thread_handles[i] = CreateThread(0, 0,
            (LPTHREAD_START_ROUTINE) Pi,
                        &threadArg[i], 0, &threadID);
  }

  WaitForMultipleObjects(NUM_THREADS,
            thread_handles, TRUE,INFINITE);

  pi = global_sum * step;

  printf(" pi is %f \n",pi);
}
```

Locks can be key bottleneck – even in case there is no conflict

Intel® Transactional Synchronization Extensions (Intel® TSX)

Intel® TSX = HLE + RTM

HLE (Hardware Lock Elision) is a hint inserted in front of a LOCK operation to indicate a region is a candidate for lock elision

- XACQUIRE (0xF2) and XRELEASE (0xF3) prefixes
- Don't actually acquire lock, but execute region speculatively
- Hardware buffers loads and stores, checkpoints registers
- Hardware attempts to commit atomically without locks
- If cannot do without locks, restart, execute non-speculatively

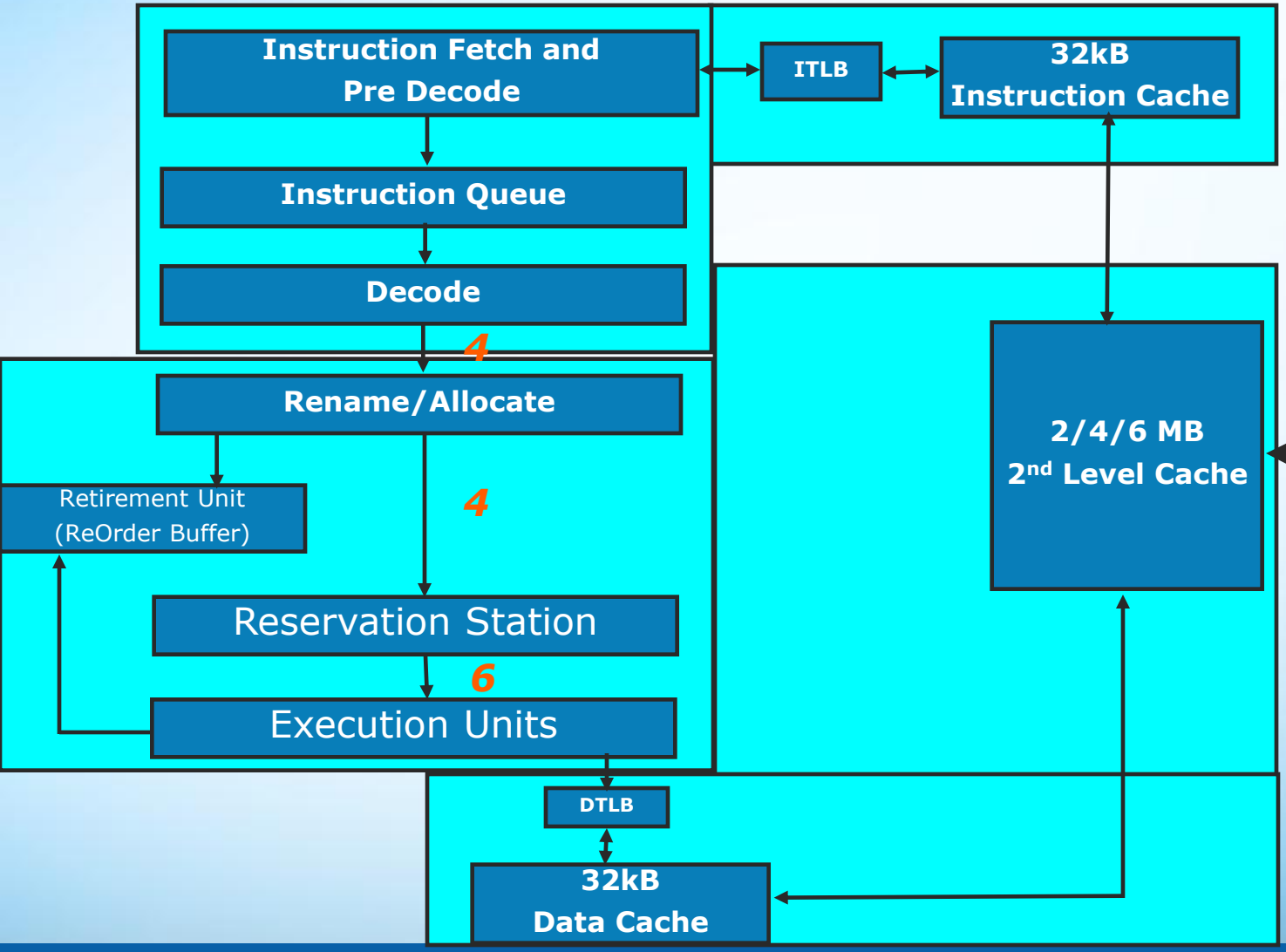
RTM (Restricted Transactional Memory) is three new instructions (XBEGIN, XEND, XABORT)

- Similar operation as HLE (except no locks, new ISA)
- If cannot commit atomically, go to handler indicated by XBEGIN
- Provides software additional capabilities over HLE

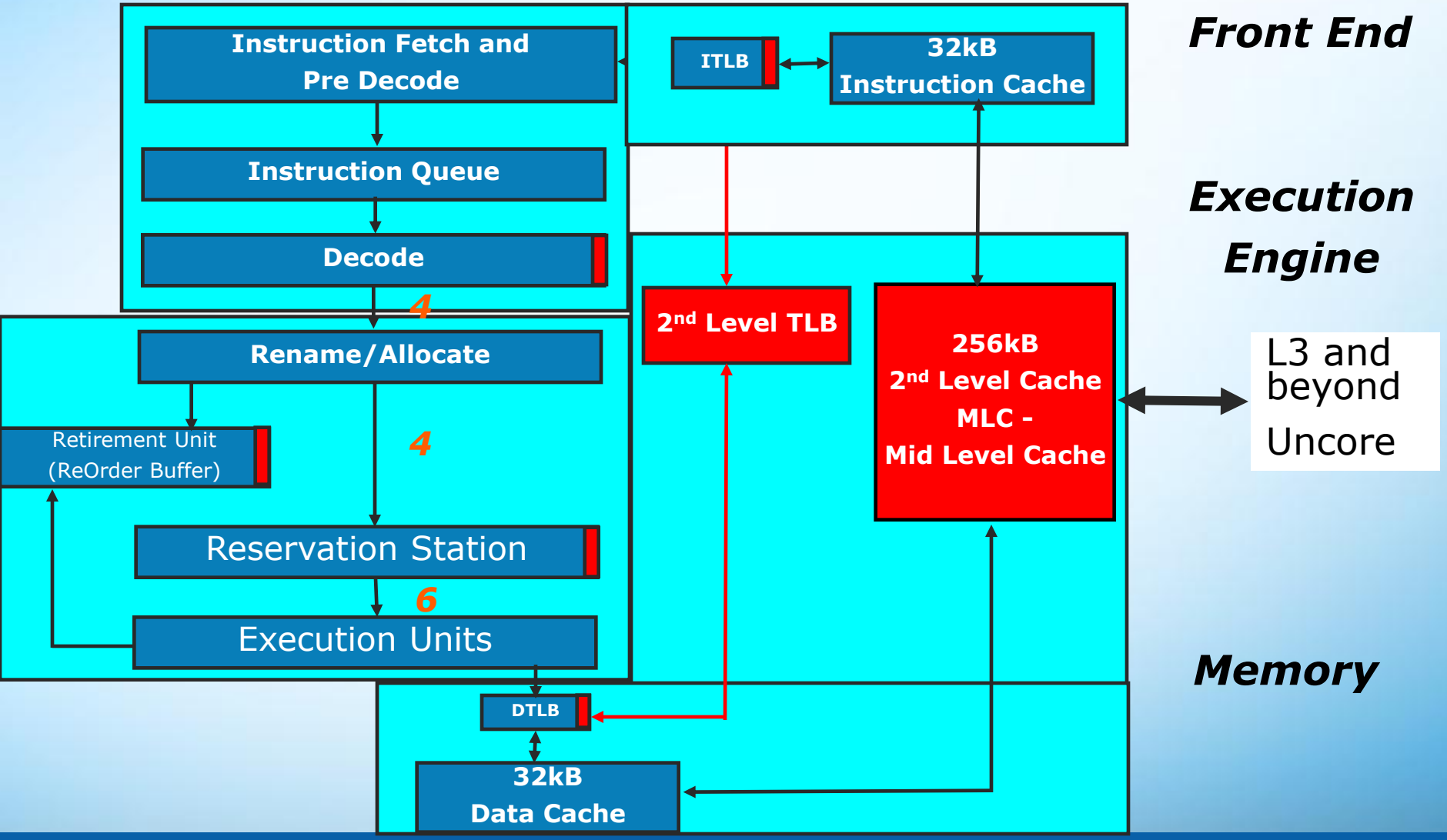
Agenda

- Overview Intel® processor architecture
- Intel x86 ISA (instruction set architecture)
- Micro-architecture of processor core
- Uncore structure
- Additional processor features
  - Hyper-threading
  - Turbo mode
- Summary

Core™ 2 Architecture (Merom)



NHM/SNB: Enhanced Processor Core



Peak FP Performance per Core & Cycle

	Single Precision	Double Precision	Comment
Nehalem	8	4	By SSE; MULT and ADD can start each cycle:
Sandy Bridge	16	8	AVX doubles all due to twice the vector length
Haswell	32	16	2 FMA instructions can start each cycle – doubling performance compared to SNB

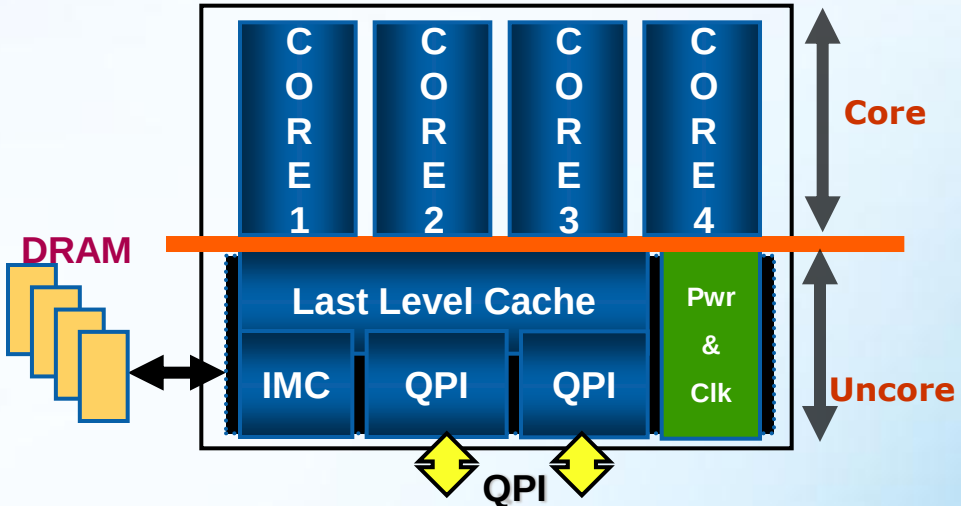
For a 2-socket, 16-core Haswell server system running at 3 GHz, this will sum up to 1.5 terra flops SP FP peak performance (0.77 for DP)

Agenda

- Overview Intel® processor architecture
- Intel x86 ISA (instruction set architecture)
- Micro-architecture of processor core
- Uncore structure
- Additional processor features
  - Hyper-threading
  - Turbo mode
- Summary

Common Core, Modular Uncore

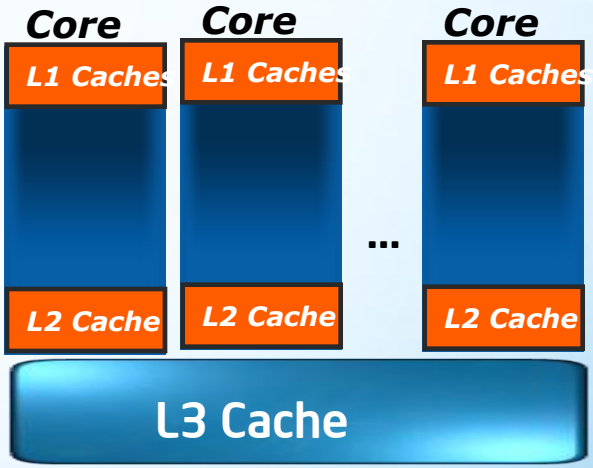
- Common "core"
  - Same core for server, desktop, mobile
  - Incremental improvements to u-arch of current Core architecture
  - Common target for SW optimization
  - Common feature set
- Segment differentiation in the "Uncore"
  - # of cores
  - # of QPI links
  - Size of L3 cache
  - # IMC channels
  - Frequency DDR3
  - Integrated graphics (GT)
  - …



	# of Cores	L3$ Size	Memory Controller	QPI Links	Graphic
Desktop i5 Desktop i3	2	4MB	2xDDR3	N/A	Yes
Desktop i7 NHM	4	8MB	3xDDR3	1 x 4.8	Yes
Desktop i7 SNB	6	8MB	3xDDR3	1 x 6.4	Yes
XEON E5-2600	2x8	20MB	4xDDR3	2 x 8.0	No
XEON E7-8870	4x10	30MB	3xDDR3	4 x 6.4	No

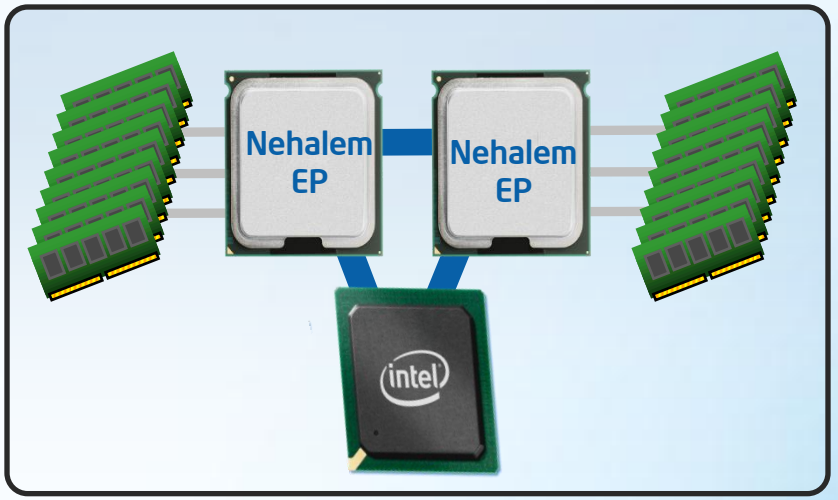
Level 3 Cache

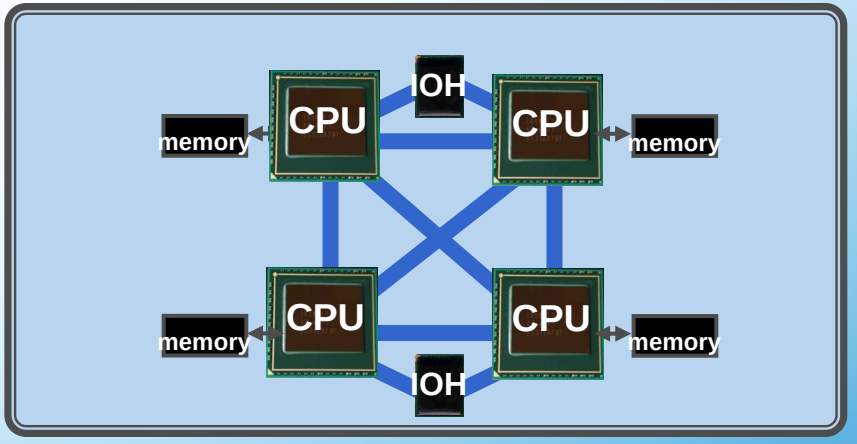
- New 3rd level cache
  - Also called LLC – Last Level Cache
- Shared across all cores of processor (socket)
- Size
  - NHM: 2MB/core (EX up to 3.0)
  - SNB: 2.5 MB/core (today)
- Latency:
  - NHM: >=35
  - SNB: 25-31
- Inclusive property
  - Cache line residing in L1/L2 ***must*** be present too in 3rd level cache



QuickPath Interconnect

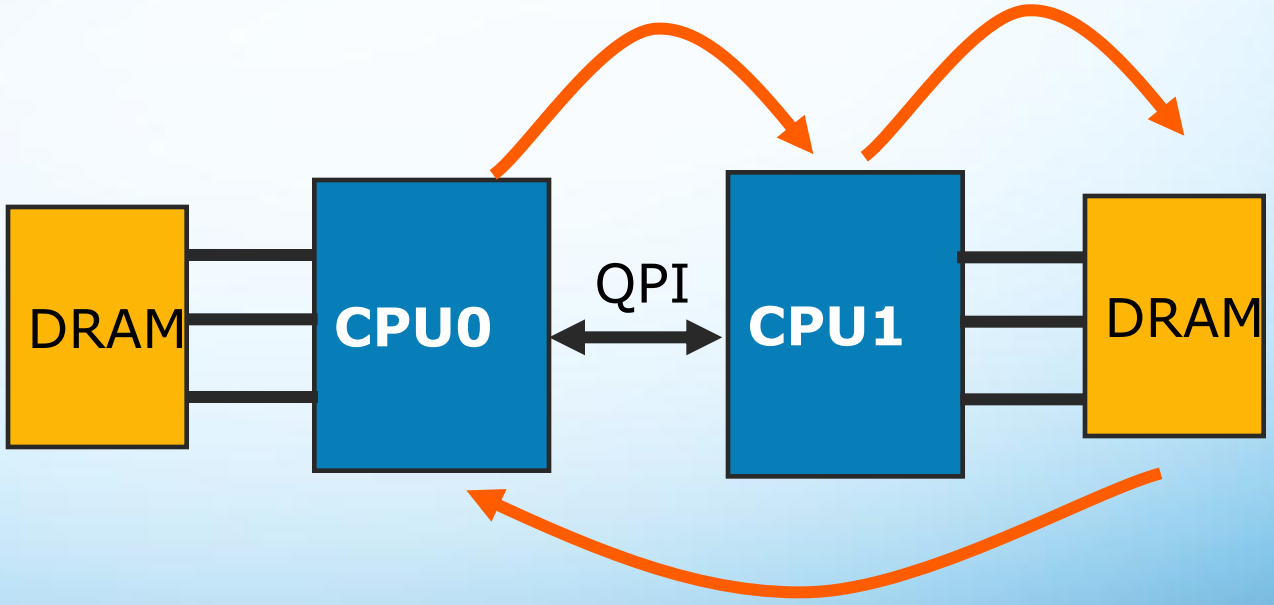
- Nehalem introduces new QuickPath Interconnect (QPI)
- ***High bandwidth***, ***low latency*** point to point interconnect
- 4.8/6.4/8.0 GT/sec
  - E.g. 6.4 GT/sec -> 12.8 GB/sec each direction
- Highly ***scalable*** for systems with varying # of sockets





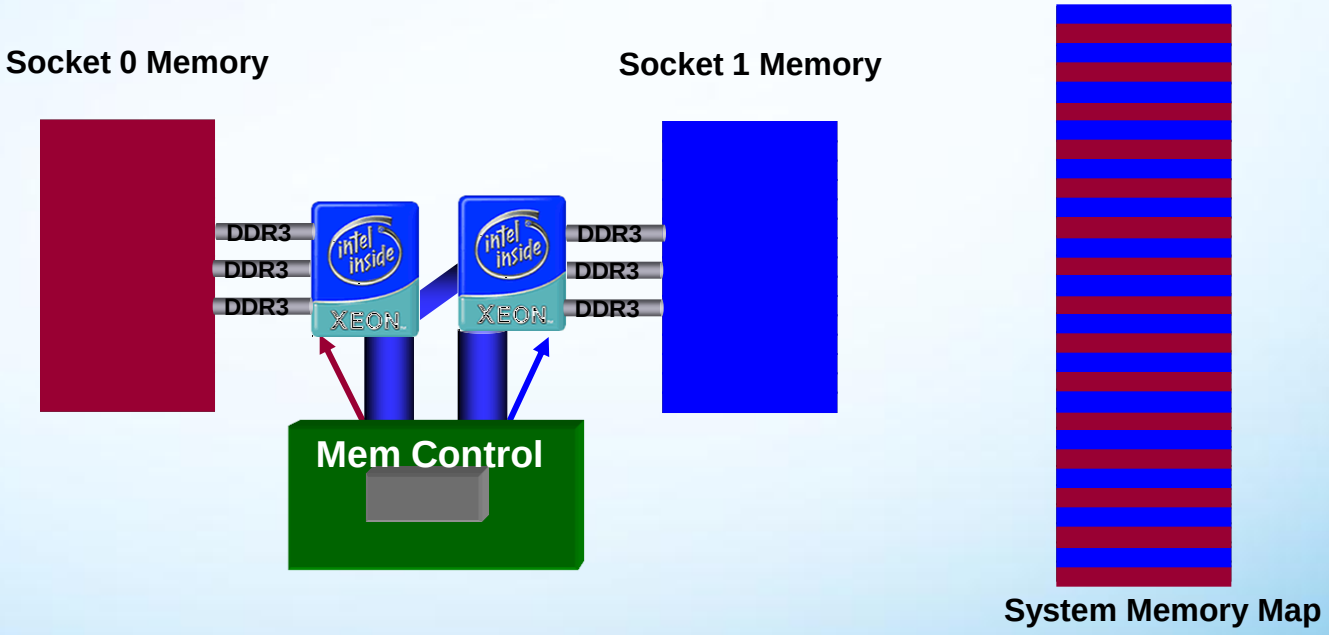
Remote Memory Access

- CPU0 requests cache line X, not present in any CPU0 cache
  - CPU0 requests data from CPU1; request sent over QPI to CPU1
  - CPU1's IMC makes request to its DRAM
  - CPU1 snoops internal caches
  - Data returned to CPU0 over QPI
- Remote memory latency a function of having a low latency interconnect
  - Typical numbers: Local access 60ns, remote access 90ns



Non-NUMA (UMA) Mode

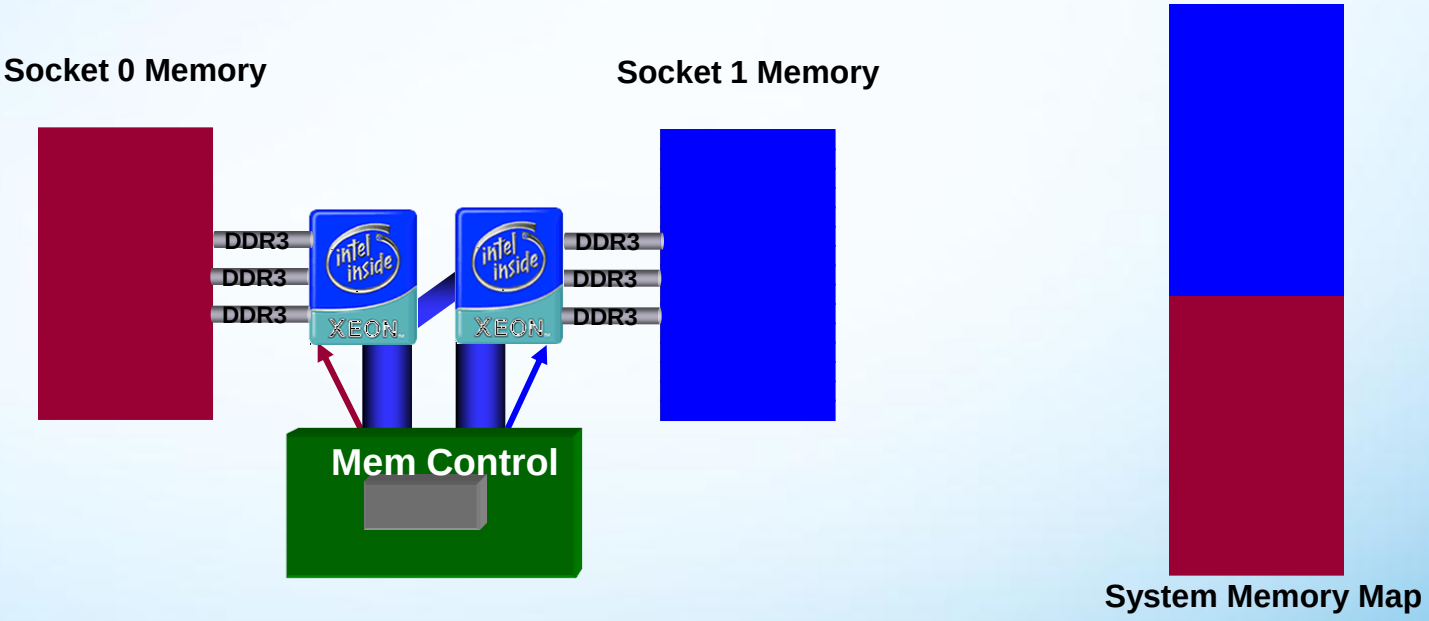
- Addresses interleaved across memory nodes by cache line
  - Some systems too support page size granularity
- Accesses may or may not have to cross QPI link



UMA lacks tuning for peak performance but in general delivers good performance without any additional tuning effort

NUMA Mode

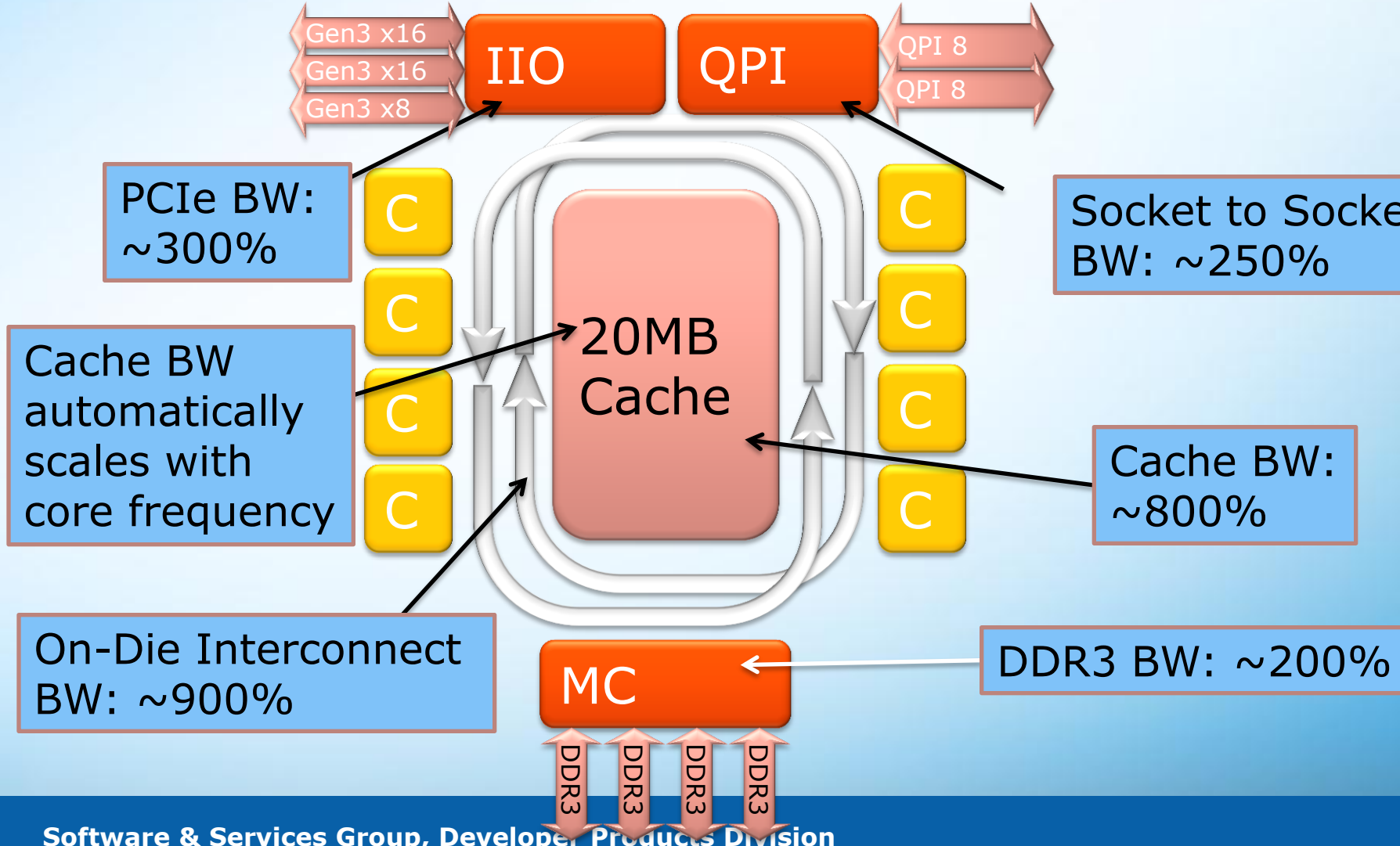
- Non-Uniform Memory Access (NUMA)
- Addresses not interleaved across memory nodes by cache line.
- Each CPU has direct access to contiguous block of memory.



Combined with thread affinity ("pinning") enables potential for peak performance but can degrade performance in case not taken care of

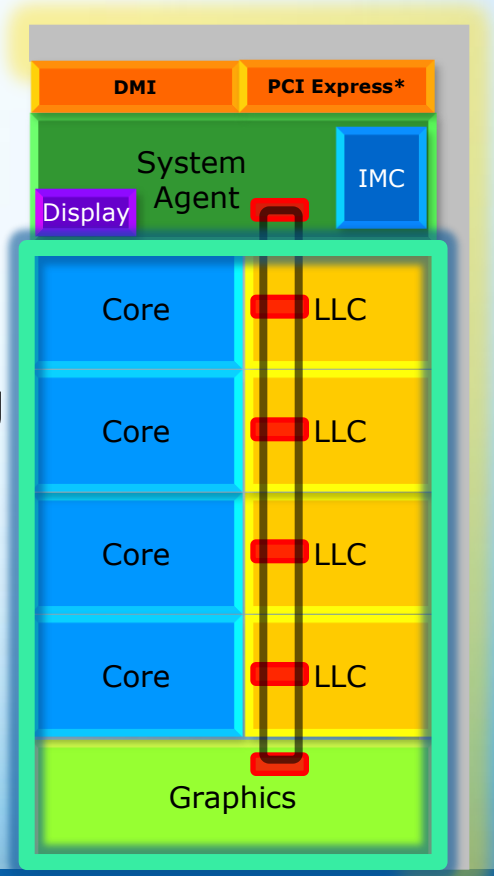
Uncore Architecture: Sandy Bridge

Significant Bandwidth Increases over Prior Generation



SNB: Scalable Ring On-die Interconnect

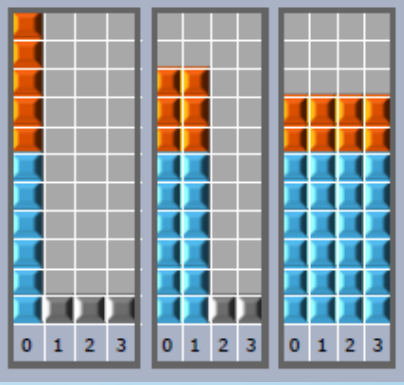
- **Ring-based** interconnect between Cores, Graphics, Last Level Cache (LLC) and System Agent domain
- Composed of **4 rings**
  - 32 Byte *Data* ring, *Request* ring, *Acknowledge* ring and *Snoop* ring
  - Fully pipelined at core frequency bandwidth, latency scale with cores
- Access on ring always picks the **shortest path** – minimize latency
- **Distributed arbitration**, sophisticated ring protocol to handle coherency, ordering, and core interface
- **Scalable to servers** with large number of processors



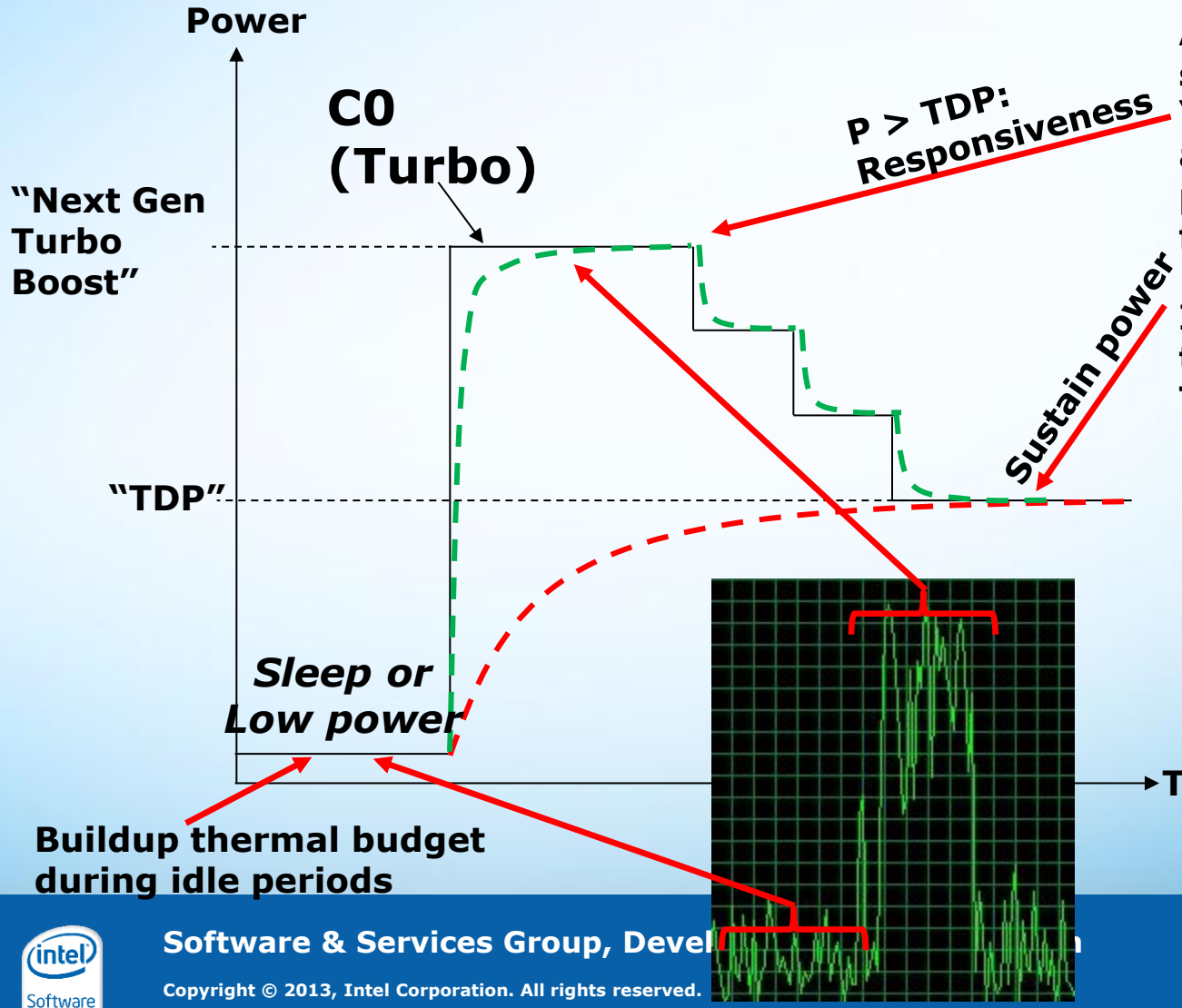
Agenda

- Overview Intel® processor architecture
- Intel x86 ISA (instruction set architecture)
- Micro-architecture of processor core
- Uncore structure
- Additional processor features
  - Hyper-threading
  - Turbo mode
- Summary

Intel® Turbo Boost Improvements

Client	Merom/ Penryn (Mobile only)	Nehalem/Westmere		Sandy Bridge
		Clarksfield Lynnfield/Clarkdale	Arrandale	
Key New Capabilities	• 1 turbo bin when other core is asleep	• Turbo controlled within power limit • Multi-core turbo • More turbo if cores are asleep	• Graphics Dynamic Frequency • Driver controlled power sharing between IA and Graphics (Mobile)	• HW controlled power sharing between IA cores and Graphics • Dynamic Turbo provides high responsiveness • More Turbo headroom from Improved power monitoring and control
Turbo Behavior Illustrative only. Does not represent actual number of turbo bins.		Quad Core Die 	Dual Core Die 	

Dynamic Adaption in Sandy Bridge



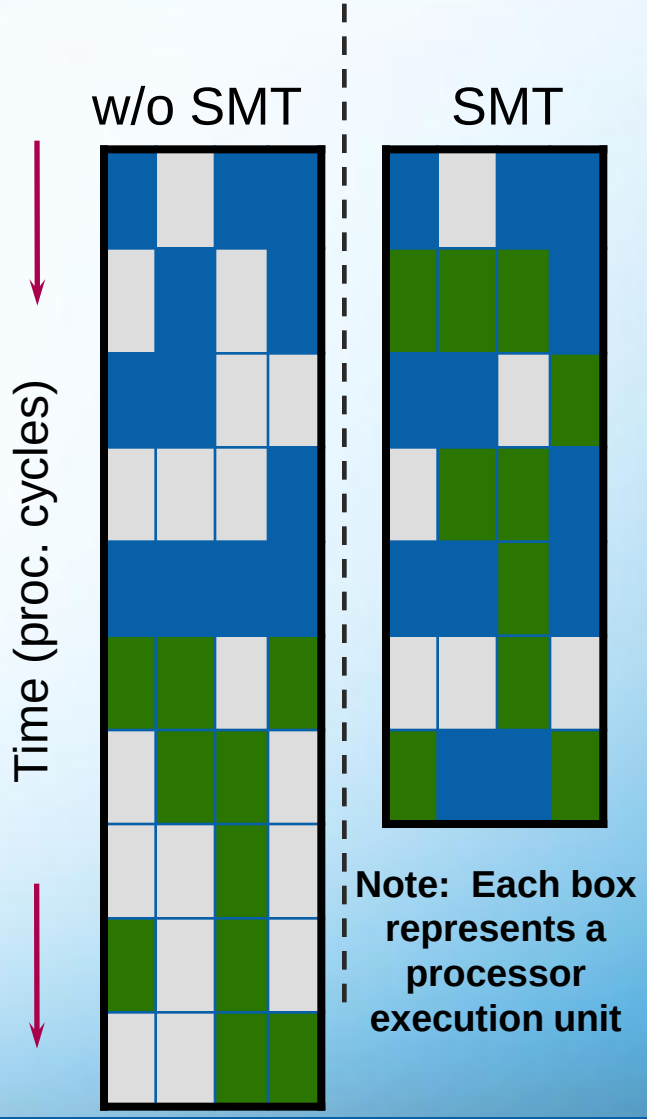
After idle periods, the system accumulates "energy budget" and can accommodate high power/performance for a few seconds

In Steady State conditions the power stabilizes on TDP

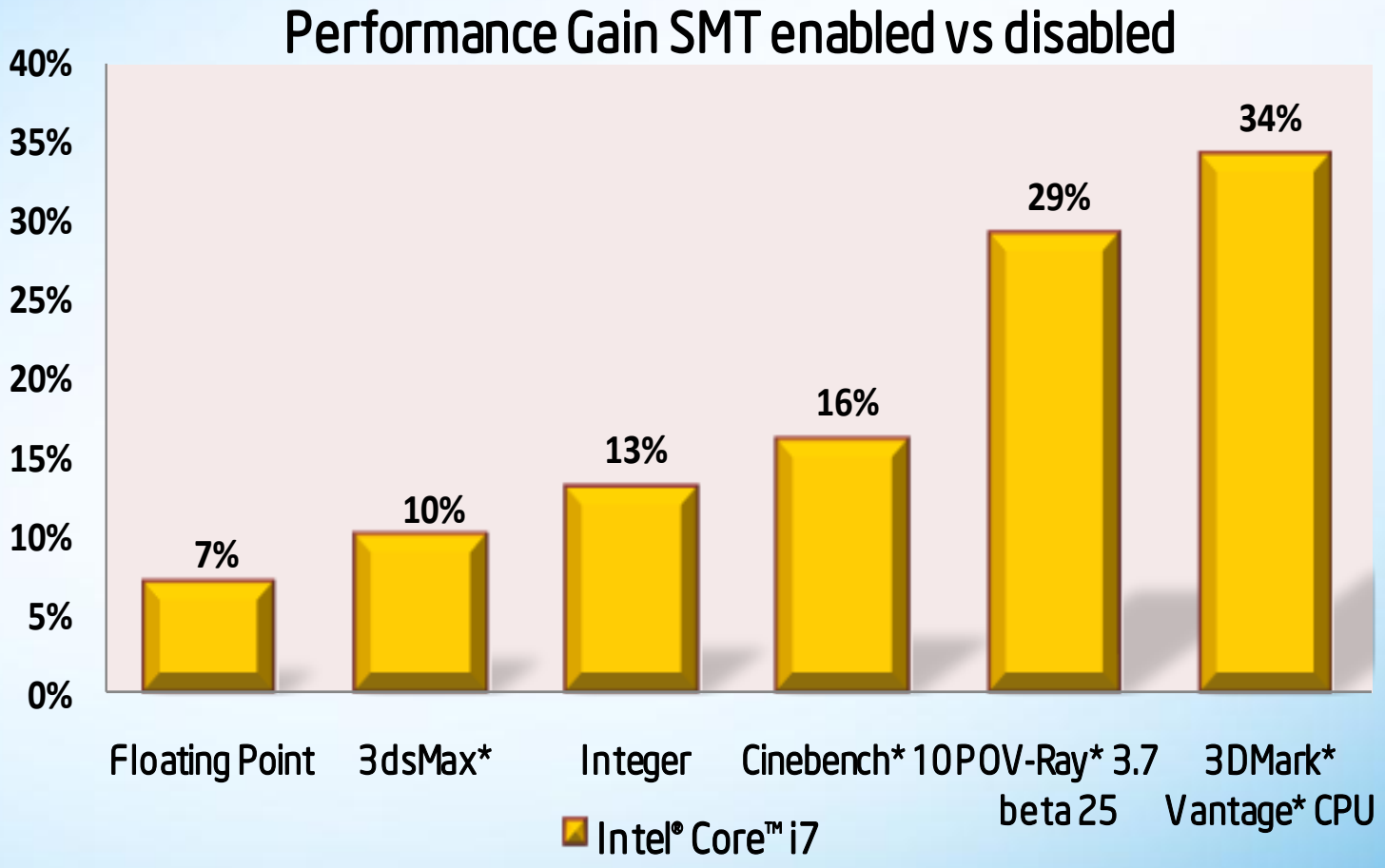
Use accumulated energy budget to enhance user experience

Simultaneous Multi-Threading (SMT)
"Intel Hyper-Threading – HT"

- Run 2 threads at the very same time per core
- Available on Nehalem (and successors) as well as Intel® ATOM Architecture
- Take advantage of 4-wide execution engine
  - Keep it fed with multiple threads
  - Hide latency of a single thread
- Most ***power efficient*** performance feature
  - Very low die area cost
  - Can provide significant performance benefit depending on application
  - Much more efficient than adding an entire core
- Nehalem advantages
  - Larger caches
  - Massive memory BW



SMT Performance Chart NHM



Floating Point is based on SPECfp_rate_base2006* estimate
Integer is based on SPECint_rate_base2006* estimate

SPEC, SPECint, SPECfp, and SPECrate are trademarks of the Standard Performance Evaluation Corporation.
For more information on SPEC benchmarks, see: http://www.spec.org

Source: Intel. Configuration: pre-production Intel® Core™ i7 processor with 3 channel DDR3 memory. *Performance tests and ratings are measured using specific computer systems and / or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, visit http://www.intel.com/performance/*

Agenda

- Overview Intel® processor architecture
- Intel x86 ISA (instruction set architecture)
- Micro-architecture of processor core
- Uncore structure
- Additional processor features
  - Hyper-threading
  - Turbo mode
- Summary

Memory Bandwidth and Performance
Sample Estimations

Platform	Memory Bandwidth	GFLOPs (DP) per core	FLOPS per DP Data Move to get Peak
NHM: 32GB/Socket 4 cores	8.00GB/core (3ch x 1333 x 8bytes)/4	12 4 x 3GHz	**12.0**
WSM: 32GB/Socket 6 cores	5.33GB/core (3ch x 1333 x 8bytes)/6	9.6 4 x 2.4GHz	**14.4**
SNB: 51GB/Socket 8 cores, SSE	6.40GB/core (4ch x 1600 x 8bytes)/8	9.6 4 x 2.4GHz	**12.0**
SNB: 51GB/Socket 8 cores, AVX	6.40GB/core (4ch x 1600 x 8bytes)/8	19.2 8 x 2.4GHz	**24.0**
Itanium 2 "Montecito" Dual core	5.40 GB/core (0.677Ghz x 16bytes)/2	6.4 4 x 1.6 Ghz	**9.5**

Tuning for memory bandwidth remains key challenge !

References

- Intel Software Development and Optimization manual
- Session from Intel Developer Forum on processor architecture – www.intel.com/idf
- Michael E. Thomadakis, Texas University, "The Architecture of the Nehalem Processor …"
- Agner, "The microarchitecture of Intel, AMD and VIA CPUs …"
- Wikipedia
  - x86
  - x86 assembly language

Legal Disclaimer

INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS". NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, reference www.intel.com/software/products.

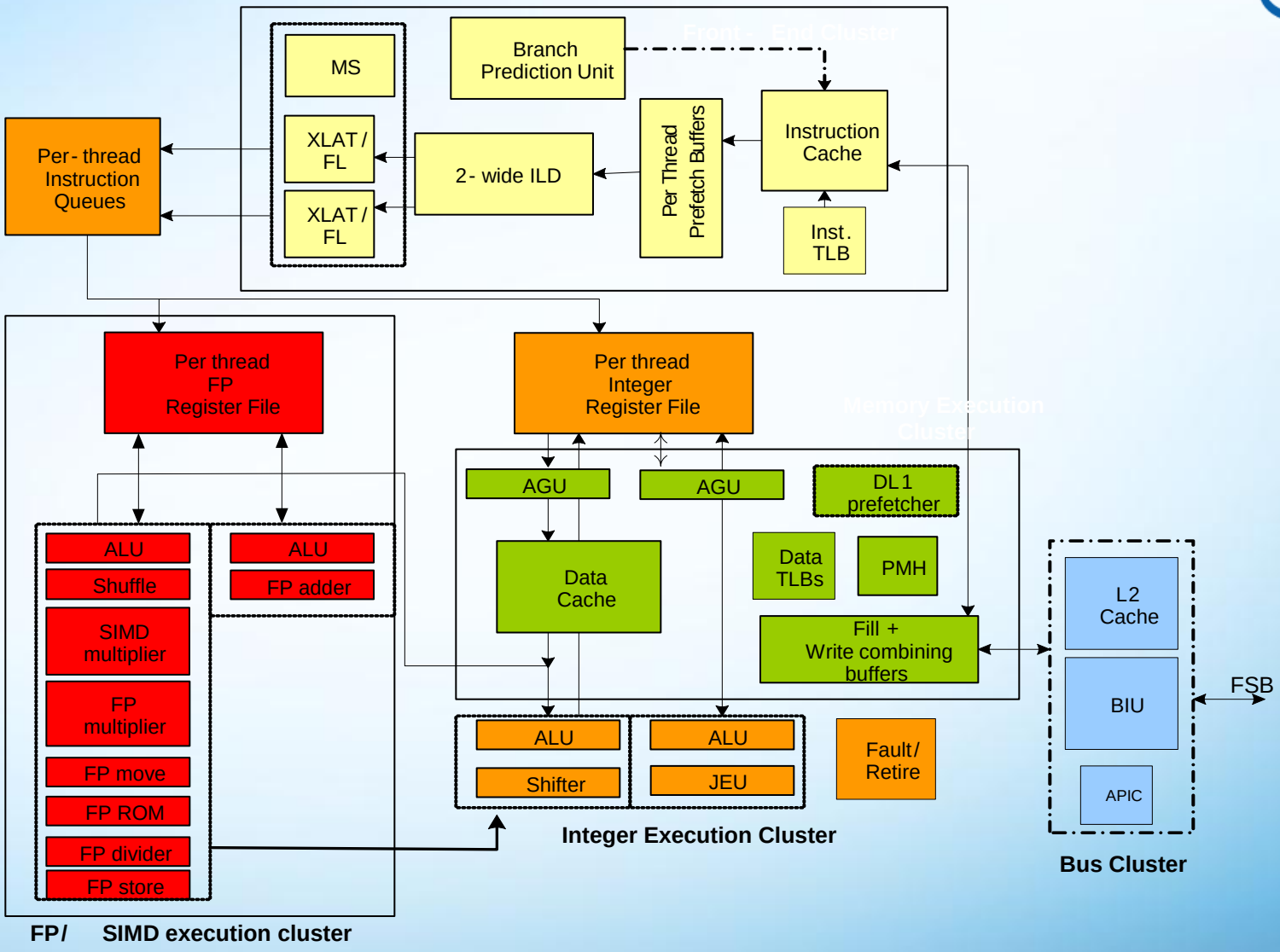
Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.
*Other names and brands may be claimed as the property of others.

Optimization Notice

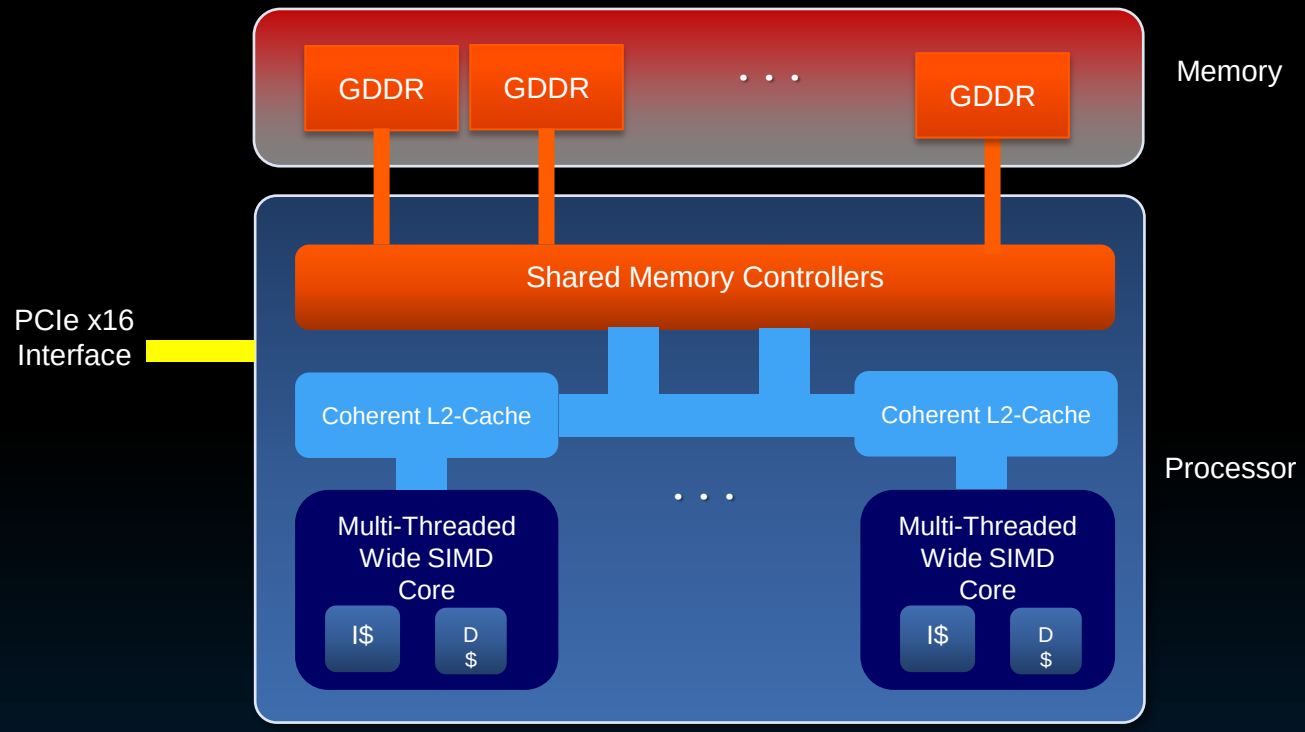
Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2®, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

Intel® ATOM Processor: Block Diagram

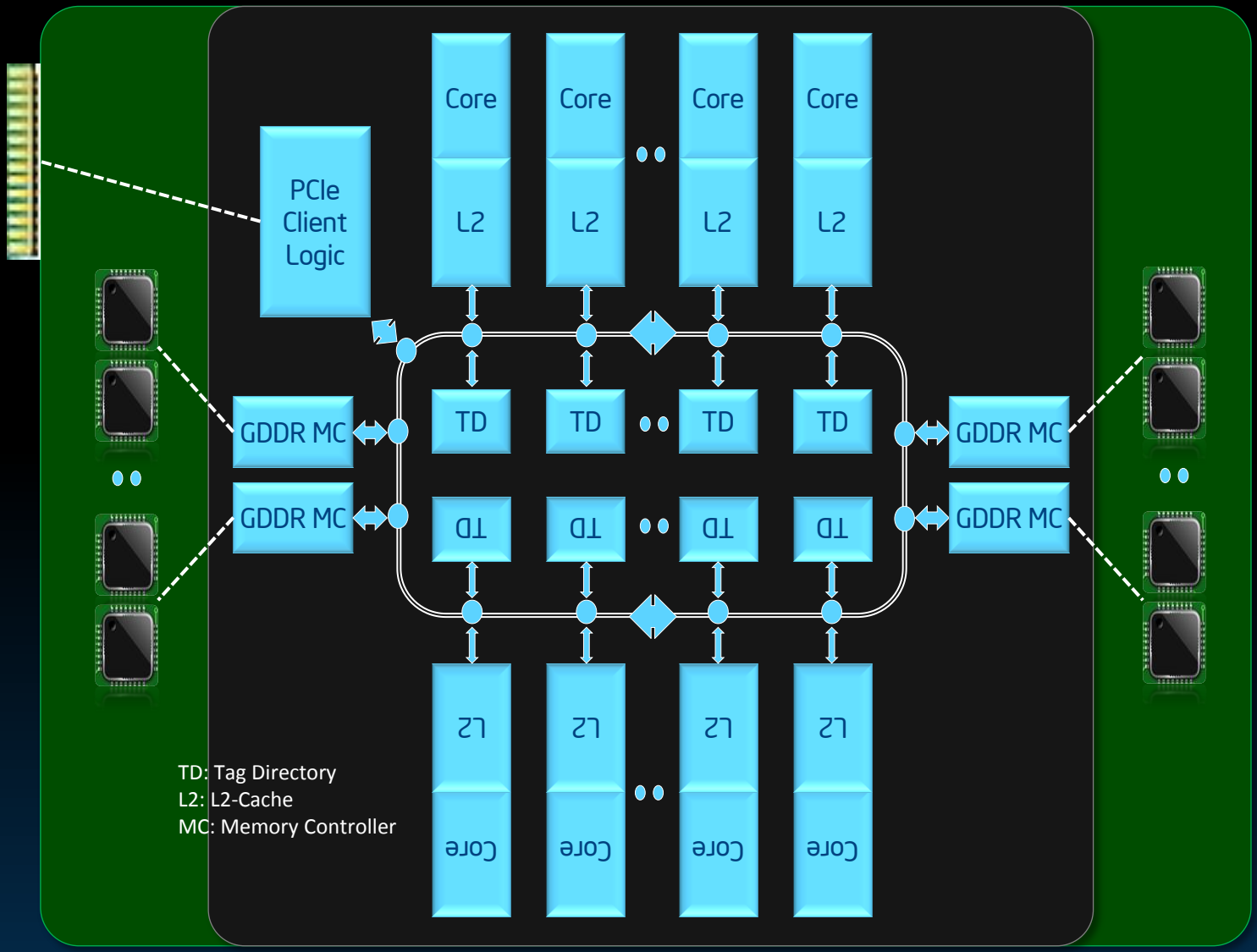


Intel® Xeon Phi™ Overview



Standard IA Shared Memory Programming

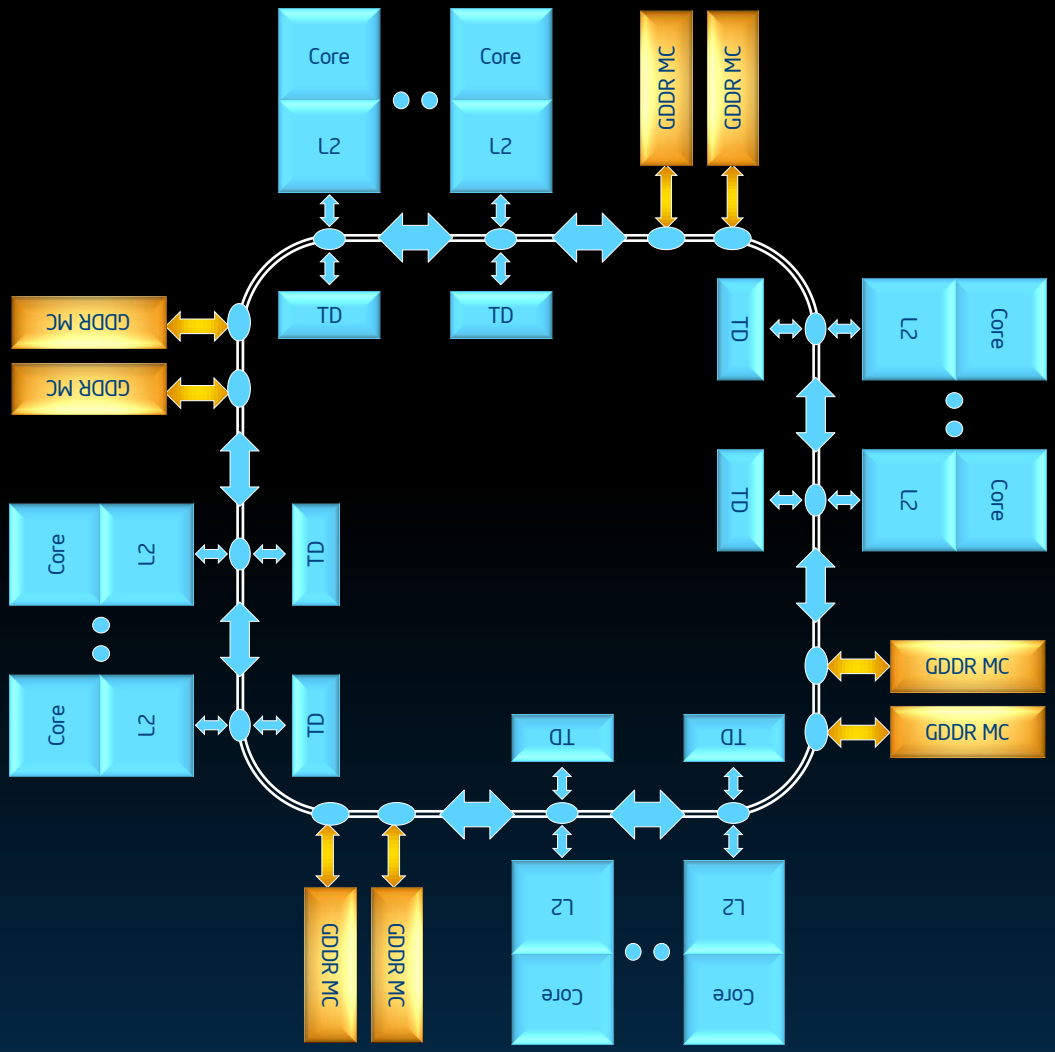
Intel® Xeon Phi™ Microarchitecture Overview



*Other brands and names are the property of their respective owners

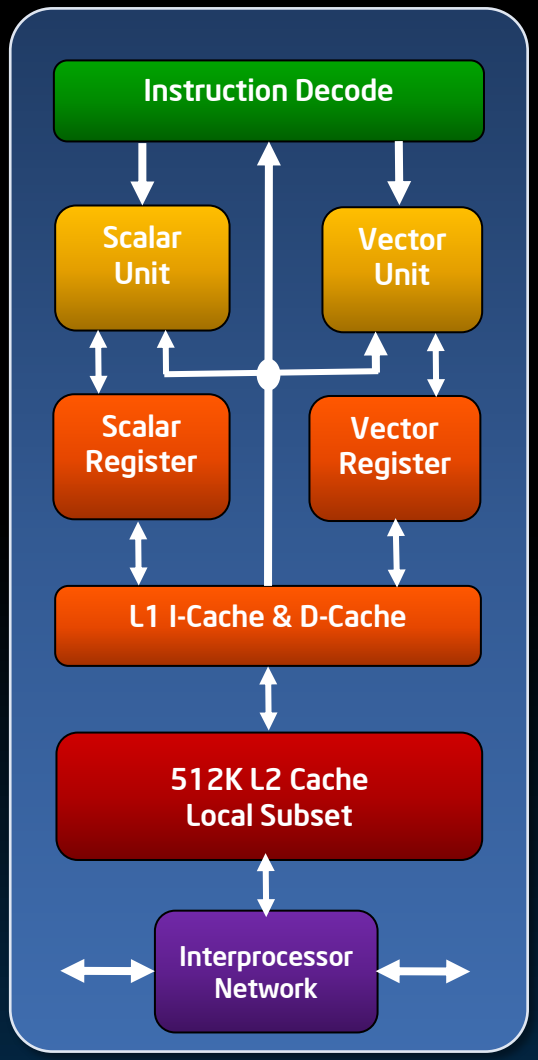
For illustration only

Interleaved Memory Access



For illustration only

Intel® Xeon Phi™ Core



Intel® Xeon Phi™ co-processor core:

- Scalar pipeline derived from the dual-issue Pentium processor
- Short execution pipeline
- Fully coherent cache structure
- Significant modern enhancements
  - such as multi-threading, 64-bit extensions, and sophisticated pre-fetching.
- 4 execution threads per core
- Separate register sets per thread
- 32KB instruction cache and 32KB data cache for each core.

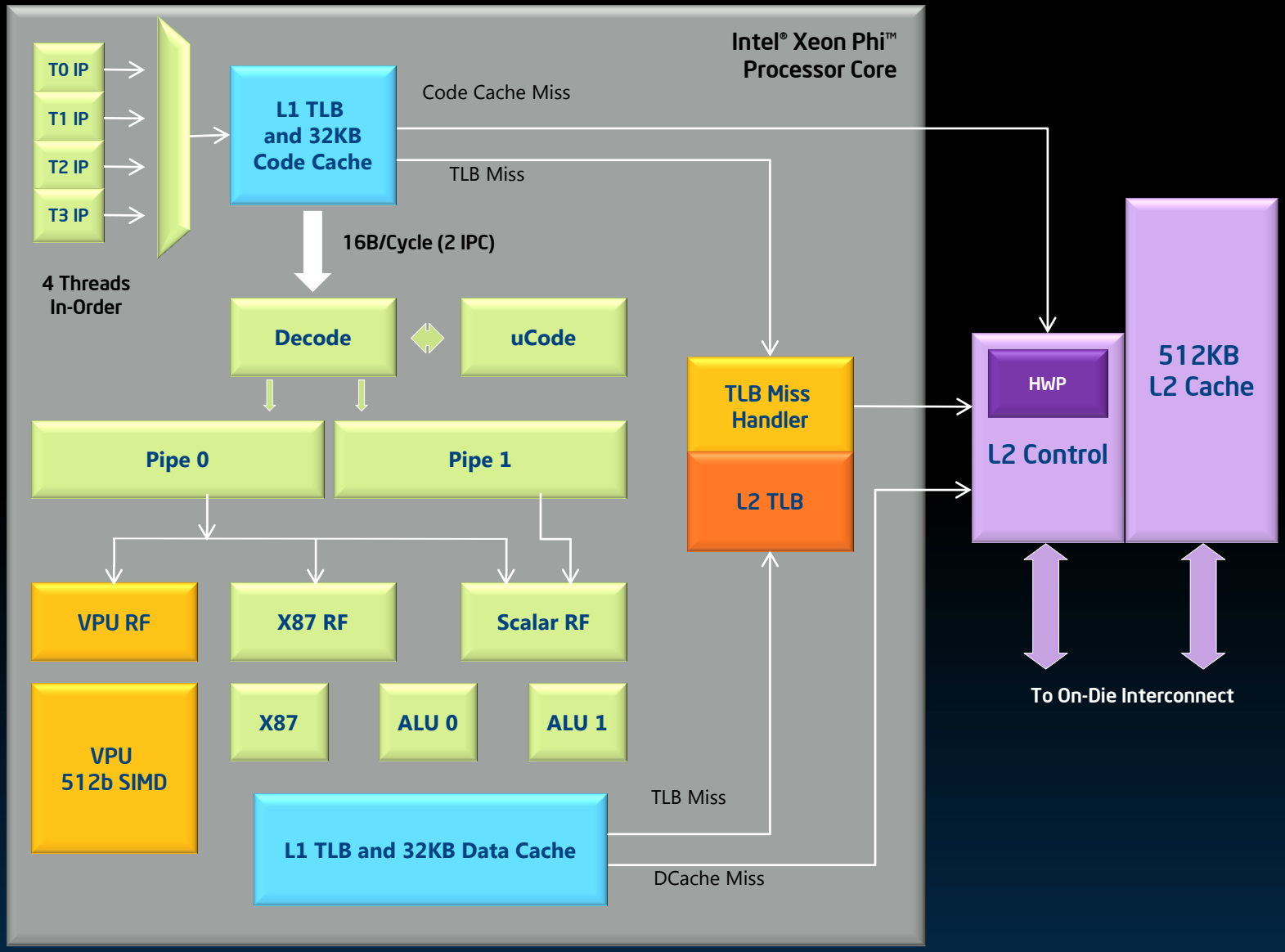
Enhanced instructions set with:

- Over 100 new instructions
- Wide vector processing operations , incl. gather/scatter and masking
- Some specialized scalar instructions
- 3-operand, 16-wide vector processing unit (VPU)
- VPU executes integer, SP-float, and DP-float instructions
- Supports IEEE 754 2008 for floating point arithmetic

Interprocessor Network

1024 bits wide, bi-directional (512 bits in each direction)

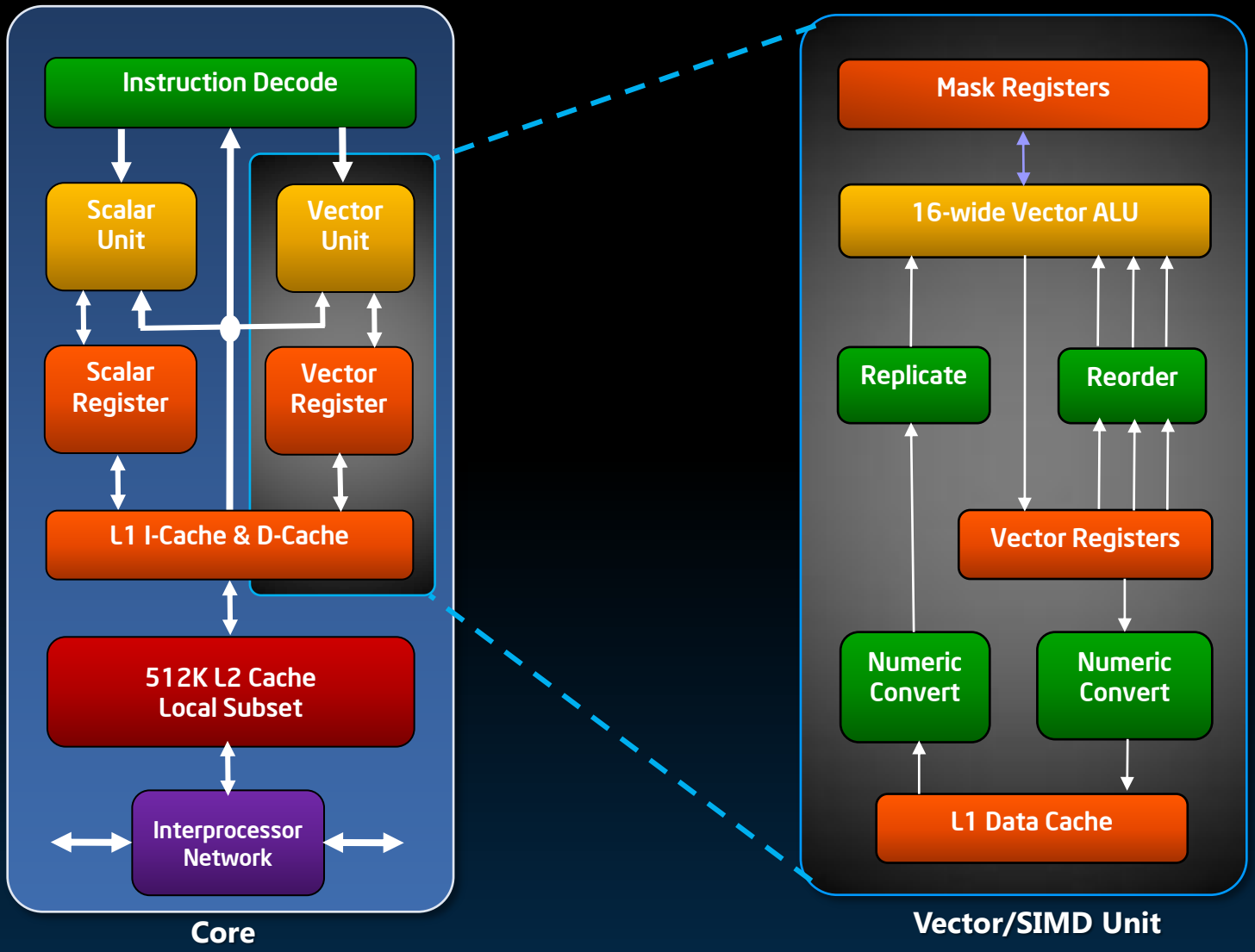
Future options subject to change without notice.



*Other brands and names are the property of their respective owners

For illustration only

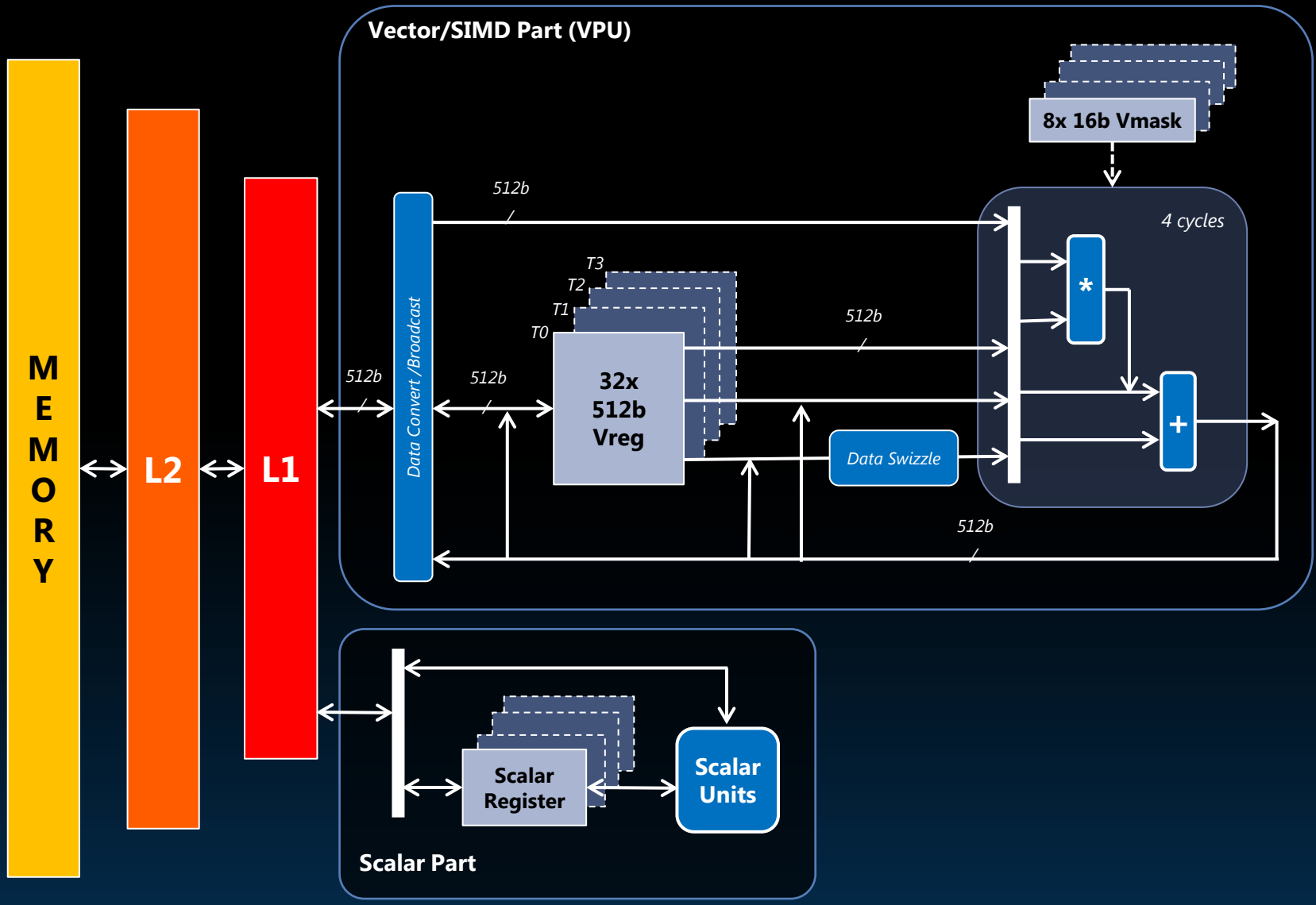
Vector/SIMD High Computational Density



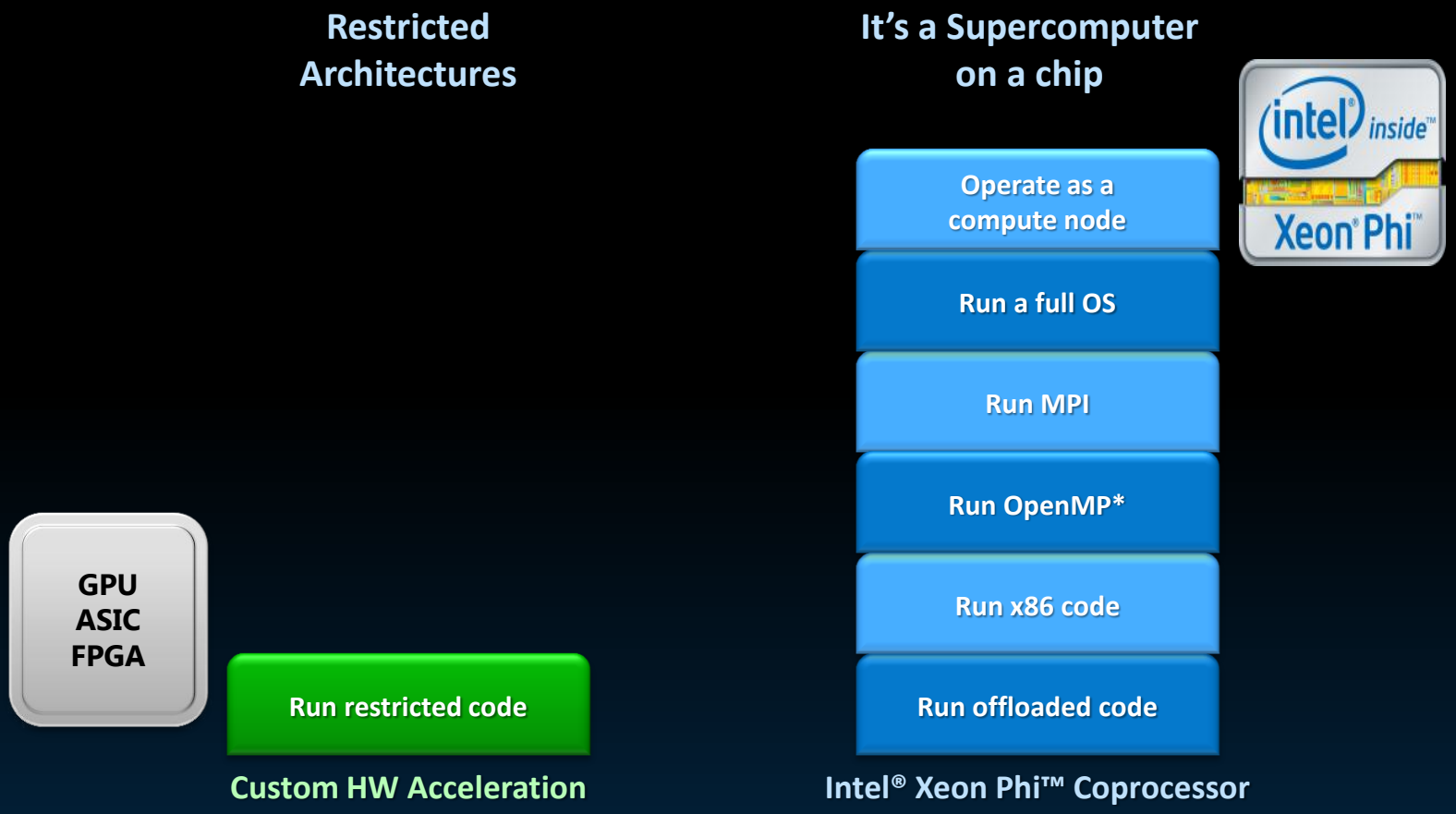
Future options subject to change without notice.

*Other brands and names are the property of their respective owners

VPU Block Diagram



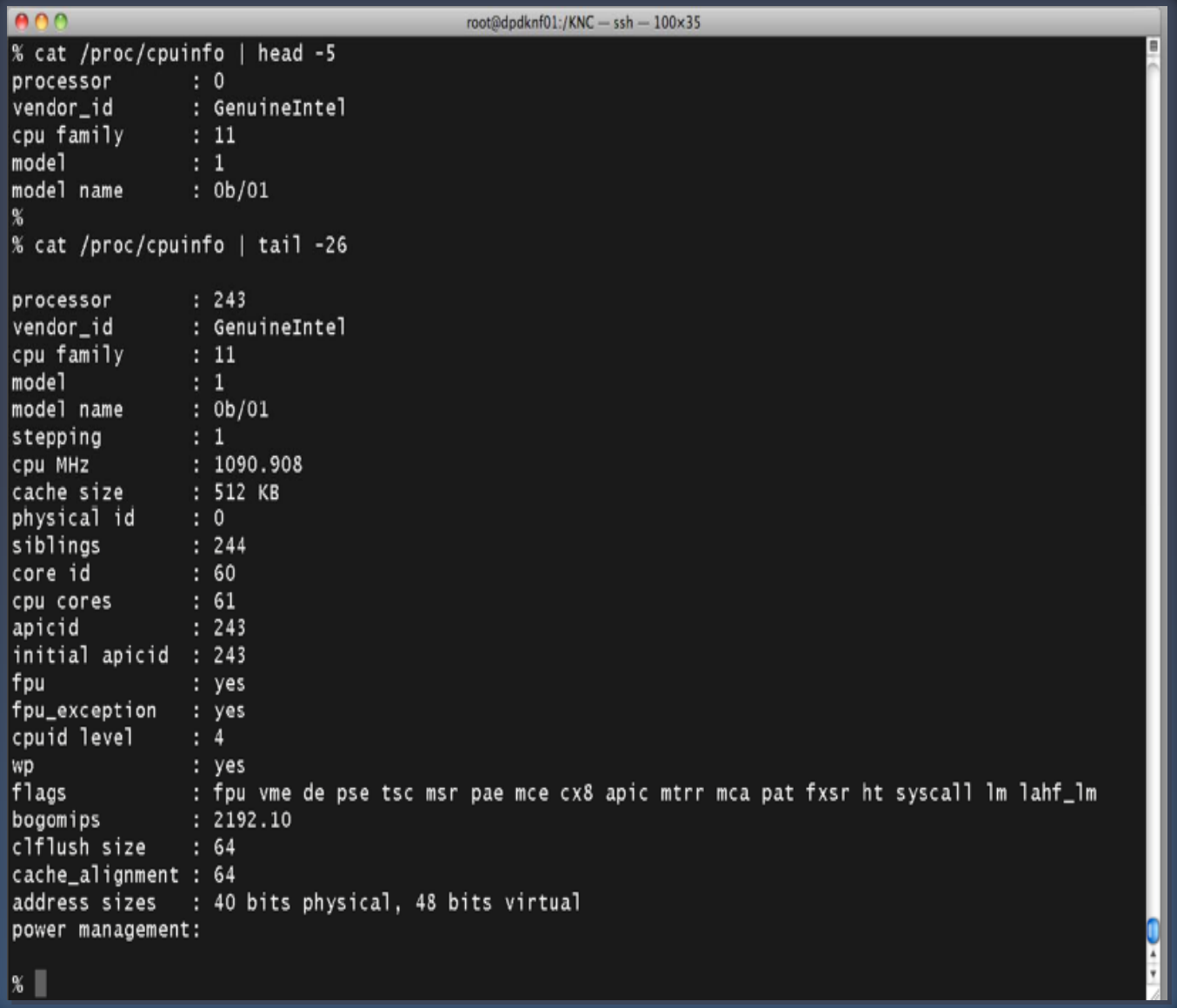
Intel® Xeon Phi™ Coprocessor: Prorgaming Model:



Restrictive architectures limit the ability for applications to use arbitrary nested parallelism, functions calls and threading models

*Other brands and names are the property of their respective owners

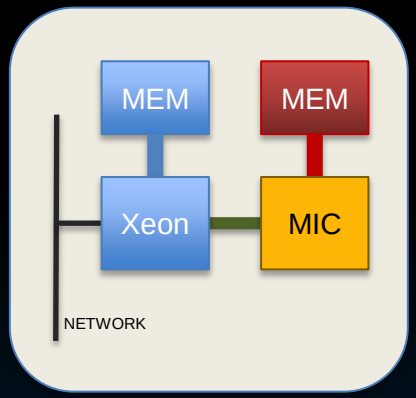
Well, it is an SMP-on-a-chip running Linux*



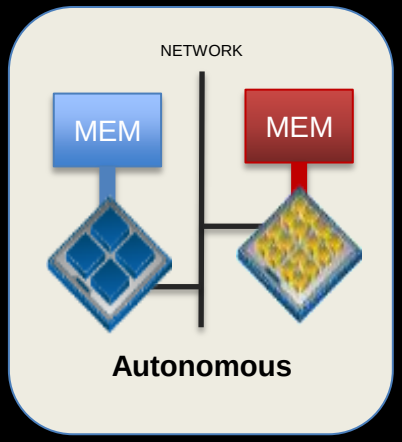
*Other brands and names are the property of their respective owners

Intel® Xeon Phi™ Environment

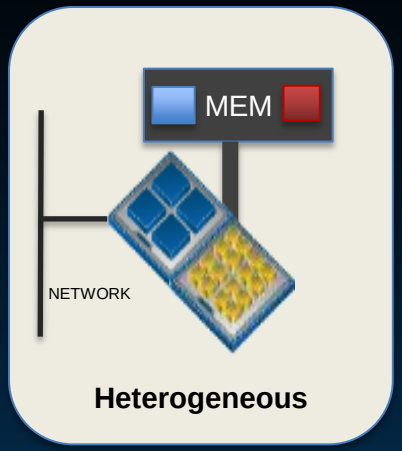
Physical View



Logical Views



NATIVE
Linux
IP
SSH
FTP
NFS
. . .

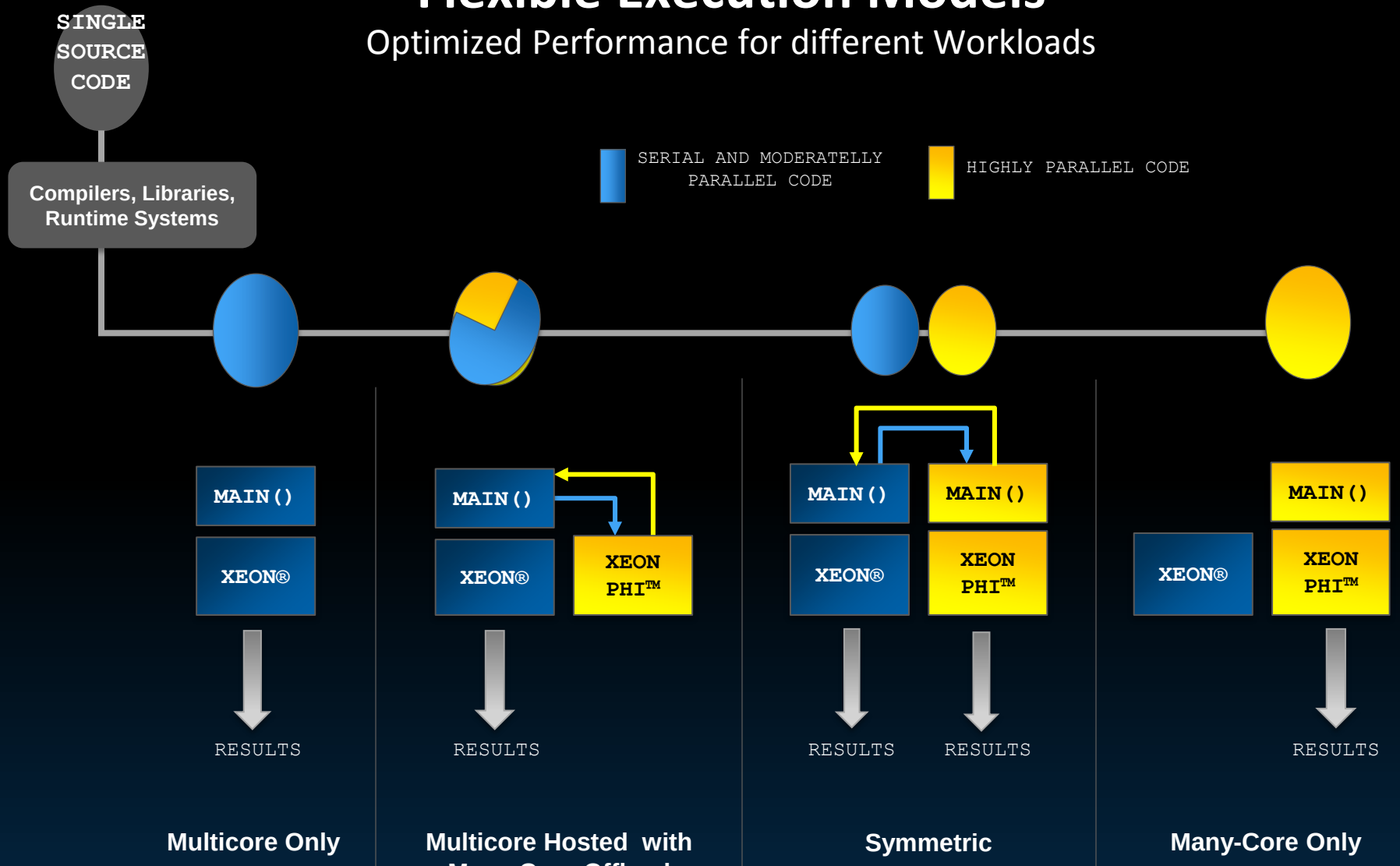


OFFLOAD

*Other brands and names are the property of their respective owners

Flexible Execution Models

Optimized Performance for different Workloads

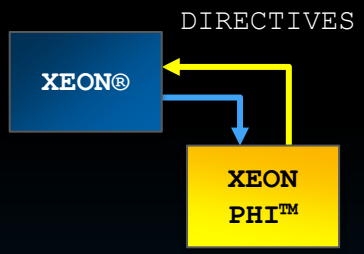


*Other brands and names are the property of their respective owners

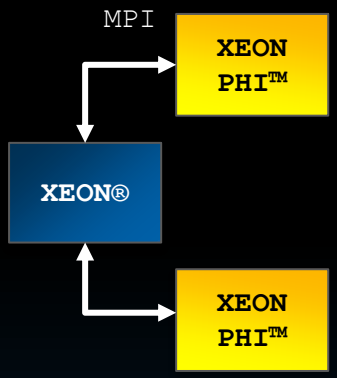
Flexible Execution Models

Optimized Performance for different Usage Models

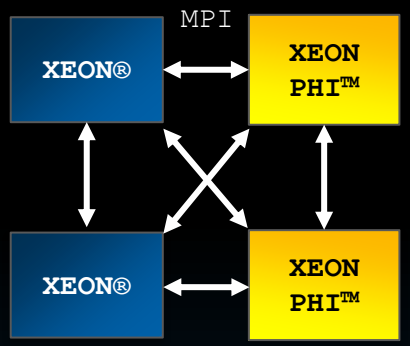
NATIVE ONLY



OFFLOAD



CO-WORKER



SYMMETRIC

*Other brands and names are the property of their respective owners