



## インテル Nehalem プロセッサに関するメモ

Nehalem は、インテルが開発する最新のマイクロプロセッサであり、インテルの「Tick-Tock モデル」(製造プロセス・ルールの微細化とプロセッサの基本構造であるマイクロアーキテクチャの刷新を交互に進めるスケジュール)では、マイクロアーキテクチャの刷新を行う”Tock”になります。



Nehalem は、Penryn に対して新しいマイクロアーキテクチャになりますが、その製造プロセスは、現在のプロセッサと同じ 45nm プロセスになります。来年には、この Nehalem の 32nm バージョンを”Tick”として製品化することになり、翌年には、再度マイクロアーキテクチャの刷新を行うこととなります。

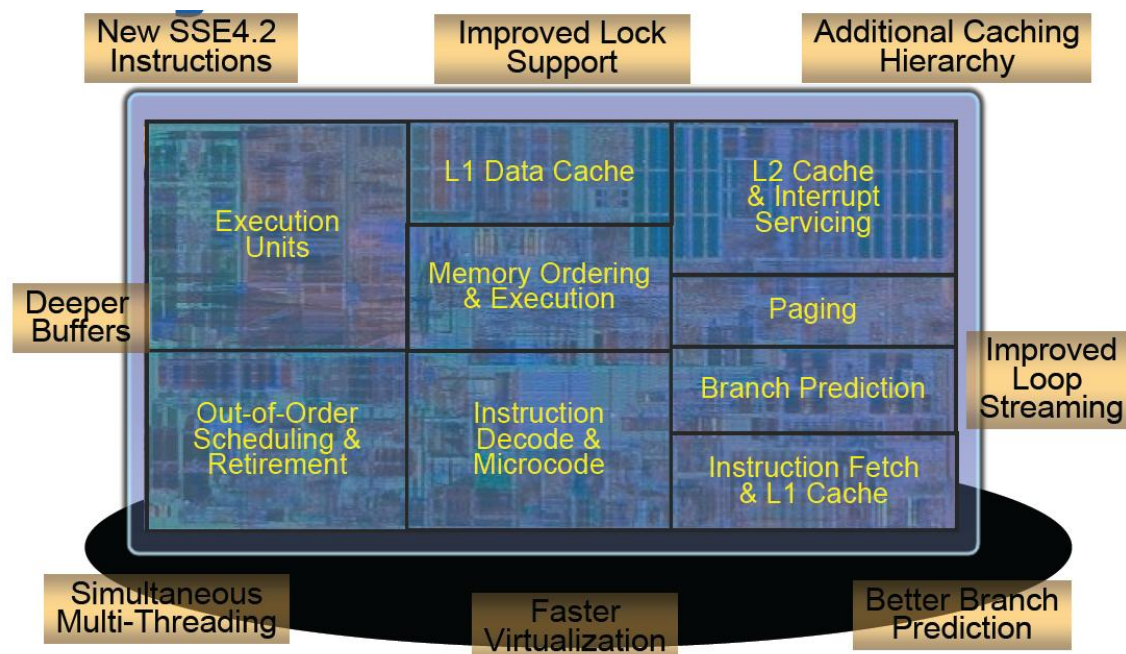
Merom のマイクロアーキテクチャが、インテルの 2 つのマイクロプロセッサ開発チーム

のうちのイスラエルの開発チーム(PentiumMなどの開発)によって行われたのに対して、今回の Nehalem は、インテルのオレゴンチーム(Pentium4などの開発)チームによって行われたことは有名です。今回、オレゴンチームが設計していますが、従来の Pentium4 などの設計とは全く違って、消費電力(TDP)と性能のバランスを最優先に設計されたプロセッサとなっています。

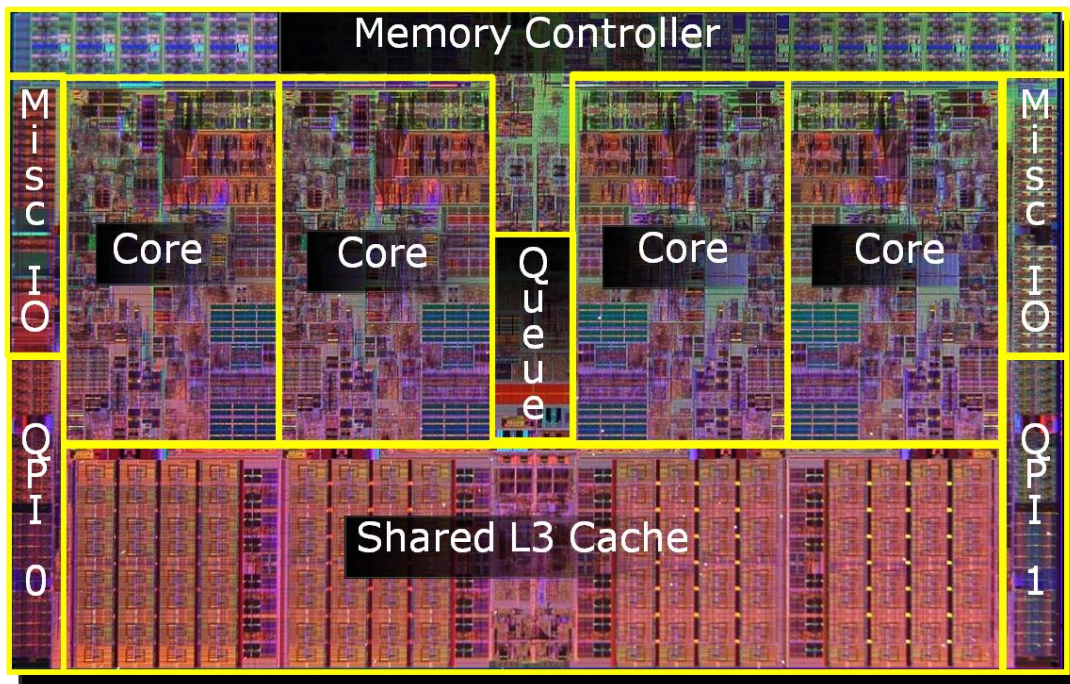
インテルは、Nehalem のブランドとして、Intel Core i7 マイクロプロセッサを発表しています。このレポートでは、Nehalem のままで説明します。

### Nehalem 概要

最初に Nehalem のダイアグラムを示します。



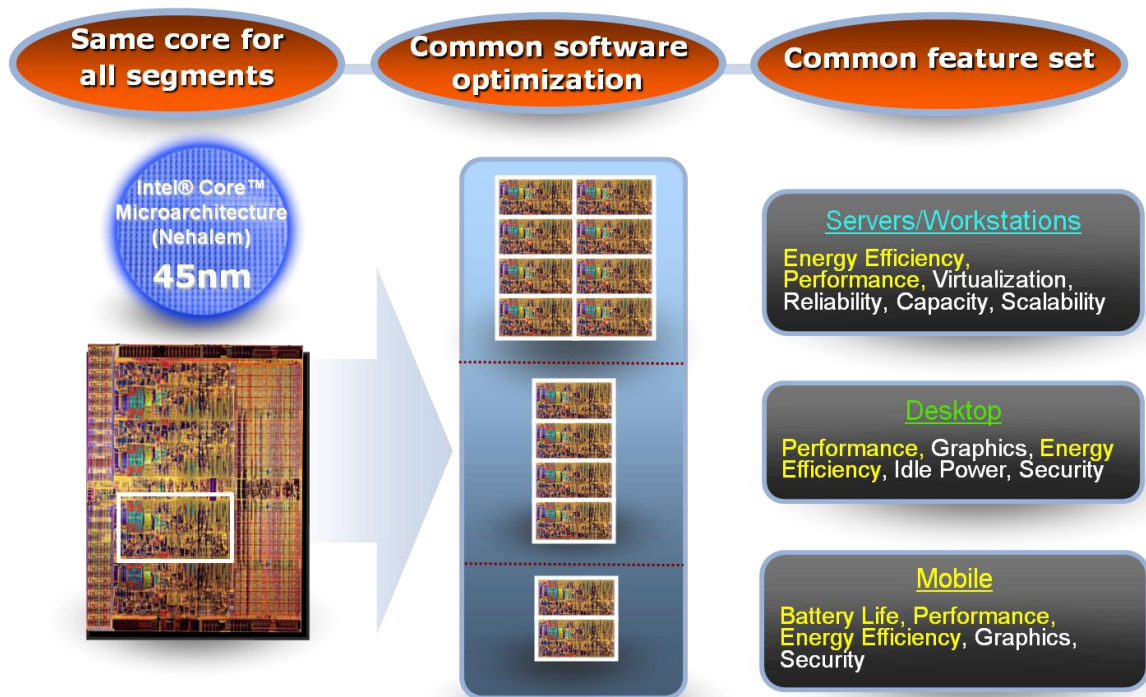
これが、Nehalem のシングルコアになります。これらのコアには、メモリコントローラ、L3 キャッシュなどは含まれません。インテルは、Nehalem から、“Core”と”Un-Core”を明確に分離しています。このプロセッサコアを見て明確なように、演算実行ユニットは、全体の 1/3 以下で、残りの大部分は、“Out-of-Order”での実行スケジューリングを行うためのロジックになっています。



クアドコア Nehalem の全体では、4つの“Core”と”Un-Core”から構成されます。”Un-Core”には、L3 キャッシュ(共有)、I/O 及びメモリコントローラ、Quick Path インターコネクタ(QPI)が含まれます。デスクトップ向けプロセッサでは、QPI リンクは一つ(QPI0)で、サーバ/ワークステーション向けでは、2つの QPI(QPI0 と QPI2)を持ちます。

Nehalem の特徴としては、このようにシステム構成を完全にモジュール化することで、“Dual-core”、“Quad-core”、“Eight-core” バージョンをスケラブルに構成することが可能となることです。

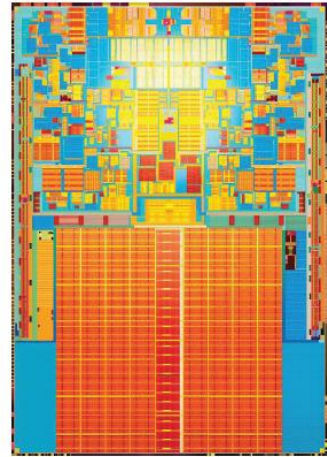




## コア・アーキテクチャ

前回のマイクロアーキテクチャの刷新を行った”Tock”では、NetBurst アーキテクチャから、Core マイクロアーキテクチャへの変更という非常に大きなマイクロプロセッサアーキテクチャの変更がなされました。Pentium4 に代表される NetBurst アーキテクチャは、プロセッサの性能を引き出すことが困難であり、ソフトウェア上の最適化が難しいという問題がありました。今回の Nehalem では、基本アーキテクチャは Core マイクロアーキテクチャであり、ユーザはソフトウェアの再コンパイルや最適化なしでも高い性能が得られるように設計されています。

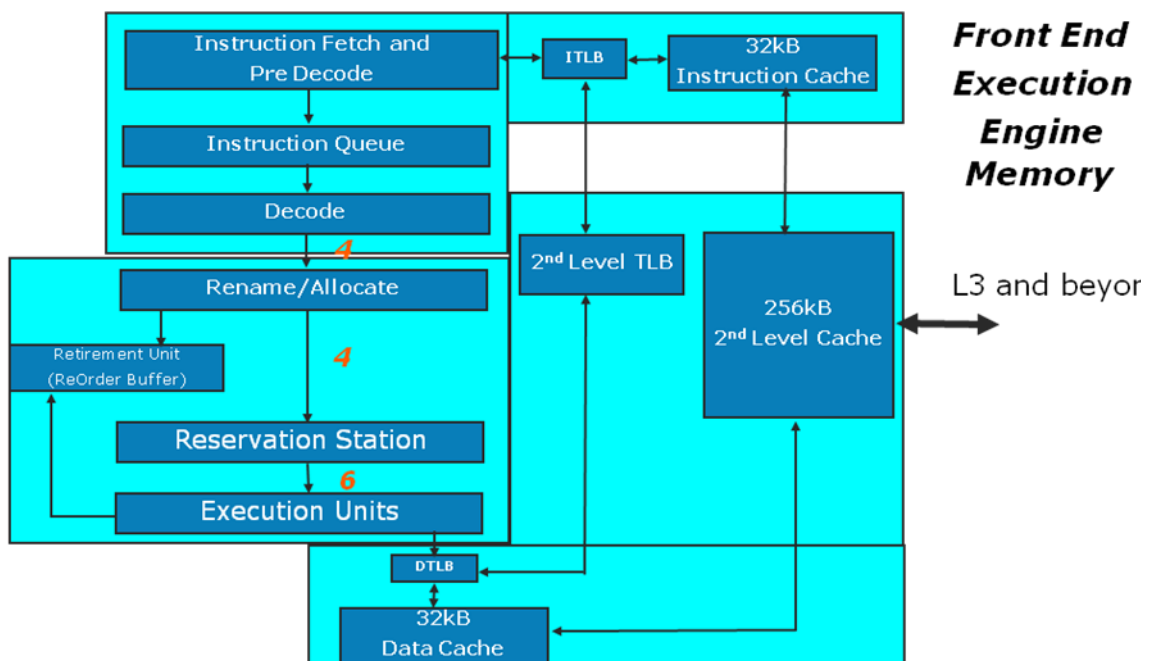
- Wide Dynamic Execution
  - 4-wide decode/rename/retire
- Advanced Digital Media Boost
  - 128-bit wide SSE execution units
- Intel HD Boost
  - New SSE4.1 Instructions
- Smart Memory Access
  - Memory Disambiguation
  - Hardware Prefetching
- Advanced Smart Cache
  - Low latency, high BW shared L2 cache



*Nehalem builds on the great Core microarchitecture*

Core マイクロアーキテクチャのマイクロプロセッサは、フロントエンドで 4 命令の処理が可能で、プロセッサは同時に 4 つの Micro-ops のデコード、リネーム、リタイアが可能な設計になっています。

Core マイクロアーキテクチャの特徴は、このようなワイドな同時実行性能であり、多くのアプリケーションで高い性能を示すことを可能としています。Nehalem では、更にこの命令の同時実行性能を高めるための改良が加えられています。



Intel は、Core マイクロアーキテクチャで、Macro-ops Fusion と呼ぶ機能を追加しました。これは、2 つの x86 命令を一つの命令に融合 (FUSION) するものであり、従来の 2 つの命令実行を一つの命令のデコード、実行、リタイアで行うことを可能とします。これによって、同時実行の幅を更に広げることが可能となります。

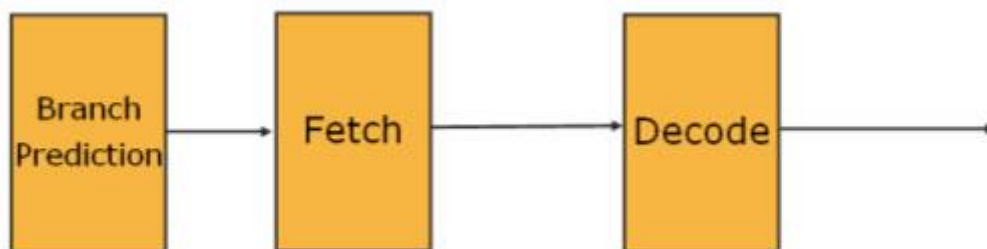
Nehalem では、従来の Core マイクロアーキテクチャでサポートされていた x86 命令の組み合わせの他に、以下の組み合わせでの Macro-ops Fusion がサポートされています。

- Support all the cases in Intel Core 2 microarchitecture **PLUS**
  - CMP+Jcc macrofusion added for the following branch conditions
    - JL/JNGE
    - JGE/JNL
    - JLE/JNG
    - JG/JNLE
  - Intel® Core™ microarchitecture (Nehalem) supports macrofusion in both 32-bit and 64-bit modes
    - Intel Core2 microarchitecture only supports macrofusion in 32-bit mode

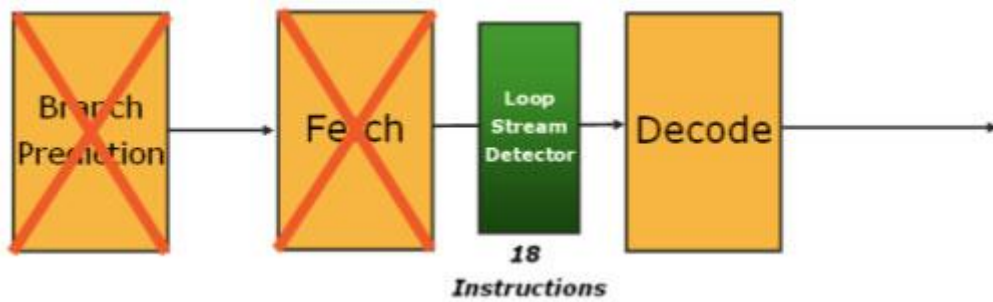
従来の Core マイクロアーキテクチャでは、Macro-ops Fusion は、32 ビットモードだけでサポートされていましたが、Nehalem では、32 ビットと 64 ビットの双方のモードでサポートされます。これによって、64 ビットアプリケーションも Macro-ops Fusion による性能向上が可能になります。

#### ループストリーミングの改善

Core マイクロアーキテクチャでは、Loop Stream Detector (LSD) によって、ソフトウェア上のプログラムのループ処理を判断し、そのループに関する分岐予測などの処理を停止し、ループ処理のストリーミング処理を行います。

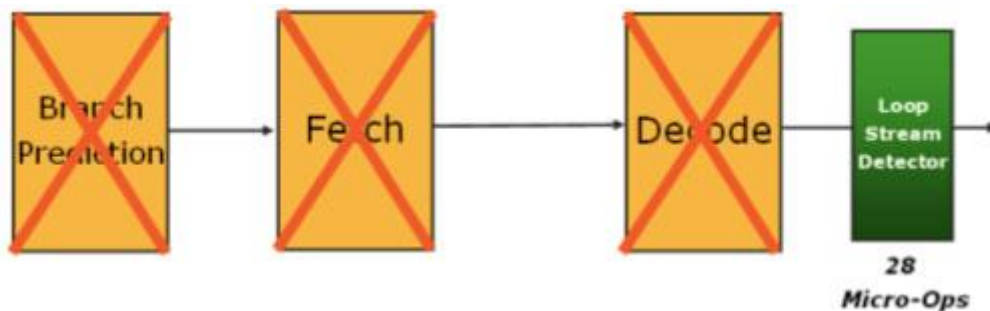


従来の分岐予測を伴う処理



Penryn での LSD による処理

分岐と Fetch 命令の実行を停止、LSD が保持可能な最大 18 命令が連続してデコードされます。この処理はループ処理が完了するか、LSD 内の命令が終了するまで続きます。



Nehalem での LSD の処理

Nehalem では、更にこの LSD がデコードの後ろに配置され、デコードされた micro-ops をキャッシュします。ループ実行が認識された場合には、Nehalem では、この LSD からデコードされた命令がストリーム処理されることになります。Nehalem では、28 個の micro-ops をキャッシュすることが可能となります。これによって、更に従来の Core マイクロアーキテクチャと比較しても、より多くの命令の実行が可能になります。

### Nehalem のサーバ向け拡張

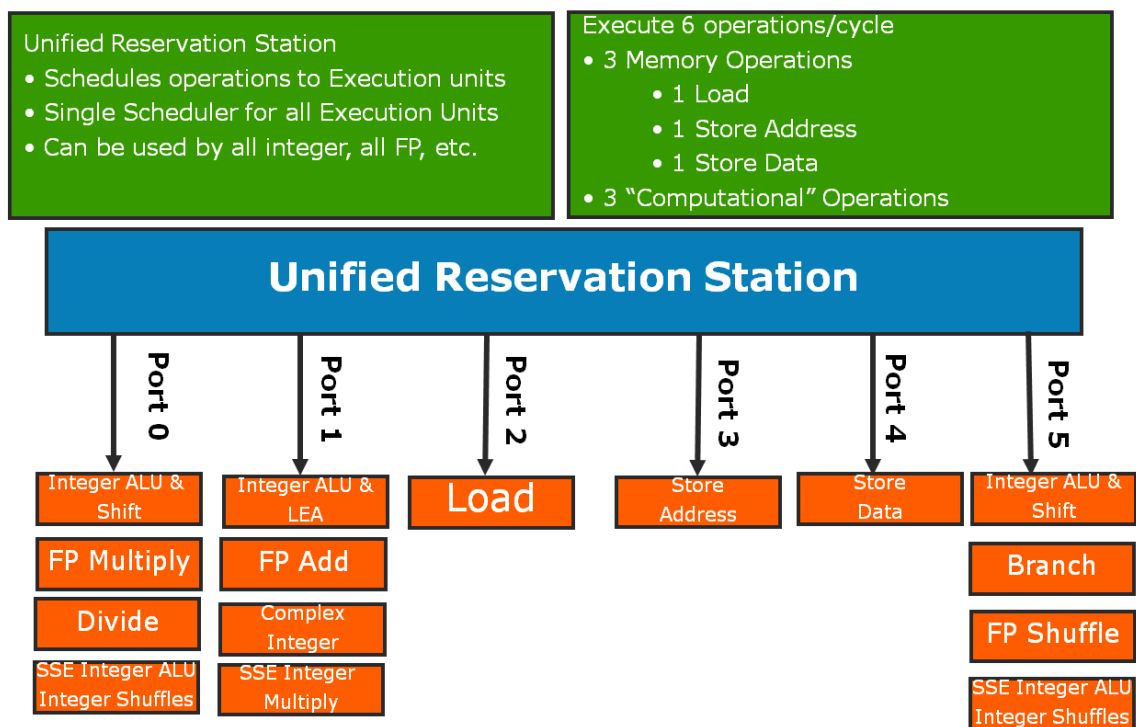
Core マイクロアーキテクチャは、非常に強力な分岐予測機能を持ちます。Nehalem ではこの分岐予測については、より長い履歴に対応できるレベル 2 の分岐予測を追加しています。非常に大きなコードサイズをもつアプリケーションについては、このような L2 分岐予測は、その性能向上に有効であり、分岐予測の正確さの向上に寄与します。同様にメモリ上のデータ構造を保持して、リターン時のスタックをリネームする Renamed Return Stack Buffer が実装されています。

インテルの Core マイクロアーキテクチャは、従来の NetBurst(Pentium4)からの転換を

図った大きなマイクロアーキテクチャの変革でした。この Core マイクロアーキテクチャは、デスクトップやモバイルをターゲットとして開発されたモバイルプロセッサ (PentiumM)を設計のオリジナルにしていたため、デスクトップとモバイルに対して、若干サーバマーケットでの利用に関して、弱点がありました。今回の Nehalem では、このデスクトップとモバイルだけでなく、サーバも含んだ全製品での高い性能を実現するための改良が加えられています。

同時にその開発に際しては、パフォーマンスと消費電力の比率に関して、最大限の関心が払われています。Pentium4 のような NetBurst アーキテクチャは、可能な限り性能を向上させるために、消費電力を犠牲にしてきましたが、Nehalem(や Atom)では、消費電力の上昇に見合うだけの性能向上が得られない場合には、その機能の追加を行わないという厳格なルールのもとで設計されたプロセッサとなりました。

### 実行ユニット



Nehalem では、命令実行ユニットも強化されています。Core マイクロアーキテクチャでは、2つのメモリ命令と3つの計算命令の同時実行が可能な構成となっていました。Nehalem で、3つのメモリ命令と3つの計算命令の合計6つのオペレーションを1クロックで実行できる構成になっています。計算処理の性能は、浮動小数点 SIMD 演算の加算は1ユニット、乗算は1ユニットで、いずれも128bit幅。基本的なユニットの

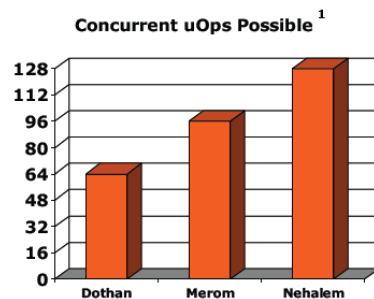


構成は Core マイクロアーキテクチャと変わりません。

Nehalem での改良点として、もう 1 つは、アウトオブオーダーウィンドウの拡大で、Nehalem は 128 個の Micro-ops をオンザフライで制御可能となっています。Core マイクロアーキテクチャでは 96 個です。。Nehalem では、それ以外のバッファも深くなっていて、リザーベーションステーションやロードバッファとストアバッファは、いずれも Nehalem でエン트리数が増やされています。

## Increased Parallelism

- Goal: Keep powerful execution engine fed
- Nehalem increases size of out of order window by 33%
- Must also increase other corresponding structures



Structure	Intel® Core™ microarchitecture (formerly Merom)	Intel® Core™ microarchitecture (Nehalem)	Comment
Reservation Station	32	36	Dispatches operations to execution units
Load Buffers	32	48	Tracks all load operations allocated
Store Buffers	20	32	Tracks all store operations allocated

## TLB とキャッシュライン

サーバアプリケーションは、TLB のサイズによる性能の影響を受けやすいものです。この物理アドレスのキャッシュである TLB を 2 階層化し、そのサイズを大きくしています。

	# of Entries
<b>1<sup>st</sup> Level Instruction TLBs</b>	
Small Page (4k)	128
Large Page (2M/4M)	7 per thread
<b>1<sup>st</sup> Level Data TLBs</b>	
Small Page (4k)	64
Large Page (2M/4M)	32
<b>New 2<sup>nd</sup> Level Unified TLB</b>	
Small Page Only	512

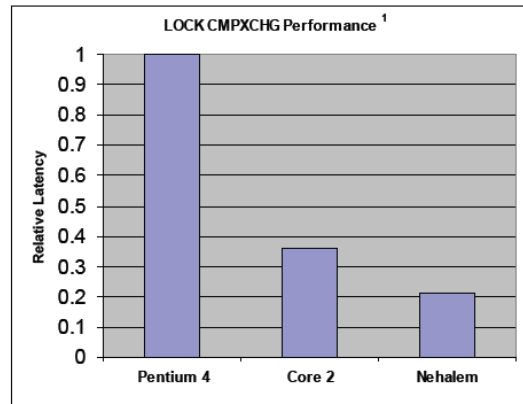
Nehalem での強化ポイントとして、重要なことに、キャッシュアクセスでのアラインメントの強化です。SSE でのメモリオペレーションでの最大のデータサイズは、16 バイト(128 ビット)になります。これらのロード/ストアオペレーションには、2 つのバージョンのオペレーションがあります。一つは、16 バイト境界でアラインメントするものとアラインメントを取らないオペレーションです。

コンパイラは、16 バイトの境界上でのアラインメントを保証出来ないメモリアクセスについては、コンパイラは、アンアラインでのアクセスを行う命令コードを使います。このアンアラインオペレーションは、実際にデータがアラインメントされていたとしても、非常に遅いオペレーションとなります。

コンパイラがデータのアラインメントについて、コンパイル時に判断することは非常に難しく、一般にはアンアラインオペレーションを行うこととなります。Core マイクロアーキテクチャでも、このデータのアラインメントは、プログラムの性能に大きく影響するものです。Nehalem では、このアンアラインオペレーションの性能がアラインオペレーションと同じになるように改良されています。Core マイクロアーキテクチャでのプログラム最適化では、データのアラインメントのためのソースコードの変更などが必要でしたが、Nehalem ではこのようなソースコードの変更の必要がありません。

## Faster Synchronization Primitives

- Multi-threaded software becoming more prevalent
- **Scalability** of multi-thread applications can be limited by synchronization
- Synchronization primitives: LOCK prefix, XCHG
- Reduce synchronization latency for legacy software



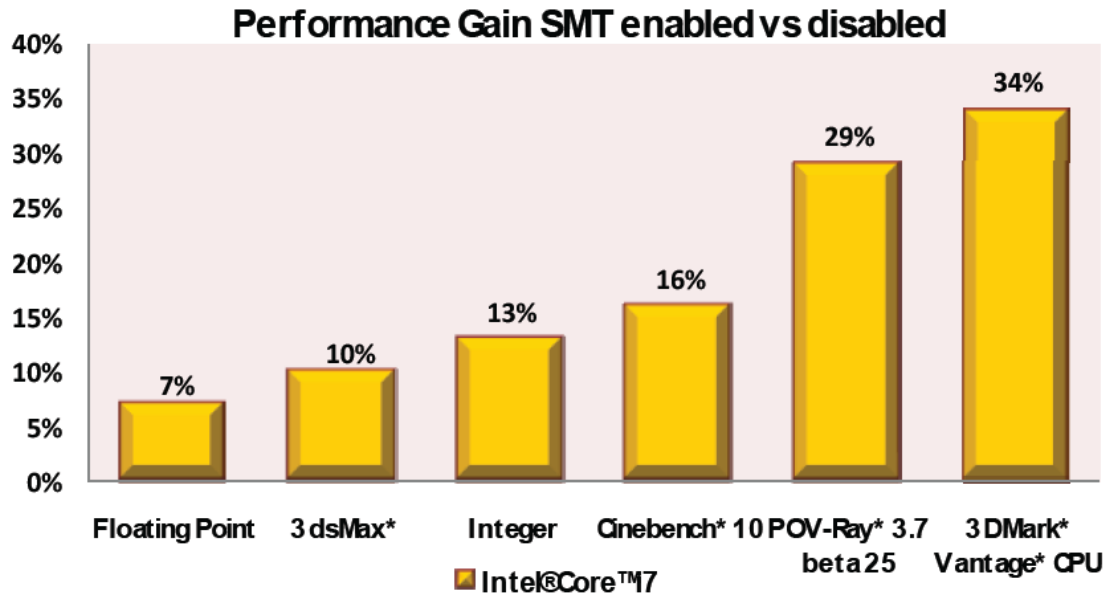
マルチスレッドアプリケーションで重要となる同期プリミティブの高速化も実現されています。

### Simultaneous Multi-Threading (SMT)

Nehalem で最も大きな拡張ポイントの 1 つは、SMT(Simultaneous Multithreading)です。2 つのスレッドを 1 つの CPU コアで走らせることができる SMT が Nehalem には実装されています。インテルは、SMT を独自に“Hyper Threading”と呼んで、Pentium4 などに実装されていました。Core マイクロアーキテクチャでは、この SMT は採用されずに、CPU コア上は一つのスレッドだけが実行できるアーキテクチャになっていました。

Nehalem での SMT 実装は、“Hyper Threading”と基本的には同じ設計方針で実装されていますが、Pentium4 での実装と比較して、Nehalem 自身のマイクロアーキテクチャの強化によって、SMP による性能向上がかなり顕著になっています。

- 1) Nehalem は、非常に高いメモリバンド幅と低いメモリアクセスのレイテンシを持ち、また、Pentium4 などと比較して、より大規模なキャッシュを持つため、マルチスレッド実行に際して、より多くのデータアクセスが可能になります。
- 2) Nehalem は、Pentium4 よりも、より拡張性の高いアーキテクチャとなっています。また、プロセッサ自身の設計がマルチスレッドの高速実行のために最適化されたアーキテクチャのため、スレッド実行に最適化されています。



Floating Point is based on SPECfp\_rate\_base2006\* estimate  
 Integer is based on SPECint\_rate\_base2006\* estimate

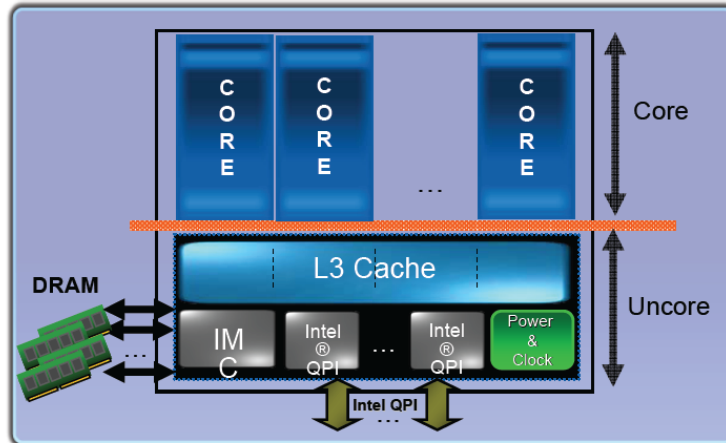
Nehalem の SMT は、CPU コア部分のダイ面積の 5～10%の増加分で実現しており、非常にパフォーマンス/消費電力の効率が良いとインテルは説明しています。

<u>Policy</u>	<u>Description</u>	<u>Intel® Core™ Microarchitecture (Nehalem) Examples</u>
Replicated	Duplicate logic per thread	Register State Renamed RSB Large Page ITLB
Partitioned	Statically allocated between threads	Load Buffer Store Buffer Reorder Buffer Small Page ITLB
Competitively Shared	Depends on thread's dynamic behavior	Reservation Station Caches Data TLB 2 <sup>nd</sup> level TLB
Unaware	No SMT impact	Execution units

## キャッシュ階層

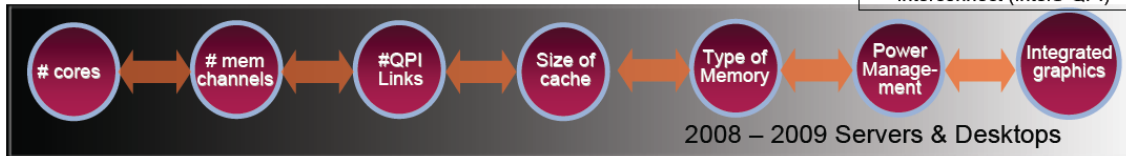


## Designed For Modularity



Differentiation in the "Uncore":

Intel® QPI: Intel® QuickPath Interconnect (Intel® QPI)



32k L1 I-cache	32k L1 I-cache	32k L1 I-cache	32k L1 I-cache
32K L1 D-cache	32K L1 D-cache	32K L1 D-cache	32K L1 D-cache
256K L2 cache data + inst.	256K L2 cache data + inst.	256K L2 cache data + inst.	256K L2 cache Data + inst.

**8 MB L3 cache**

For all applications to share      Inclusive cache policy to minimize traffic from snoops

Nehalem では、L1、L2、L3 の 3 階層のキャッシュ構成となっています。各 CPU コア毎に 32KB の L1 命令キャッシュと L1 データキャッシュ、それに 256KB の専用 L2 キャッシュを備え、さらに 3 層目として、CPU 中の全ての CPU コアで共有する大容量の L3 キャッシュを備える構成となっています。

Penryn と比較して、L1 キャッシュはサイズは同じですが、若干遅くなっています。(4 cycles vs. 3 cycles)。これは Nehalem のコアが Penryn より大きくまた、複雑な構成に

なっている影響で、性能としては、2-3%の低下が予想されるということです。

Penryn では、6MB キャッシュを 2 つのコアで共有することになっていましたが、Nehalem では、L2 キャッシュを各コア上に持ちます。この L2 キャッシュのサイズは、256MB のサイズで、従来のインテルのマイクロプロセッサと比較しても、L2 キャッシュのサイズとしては、少ない容量になっています。各 CPU コア占有の L2 キャッシュを導入した理由は、レイテンシの低減とスケーラビリティのためだと説明されています。キャッシュを小容量化することで高速アクセス(10 サイクルでのデータアクセス)を可能としています。実際には、L2 キャッシュは、L3 キャッシュのバッファとしての役割を想定されています。その L3 は、Core i7 processors では、8MB のサイズとなっており、コア数や用途別の製品では、この L3 キャッシュのサイズを調整することになります。

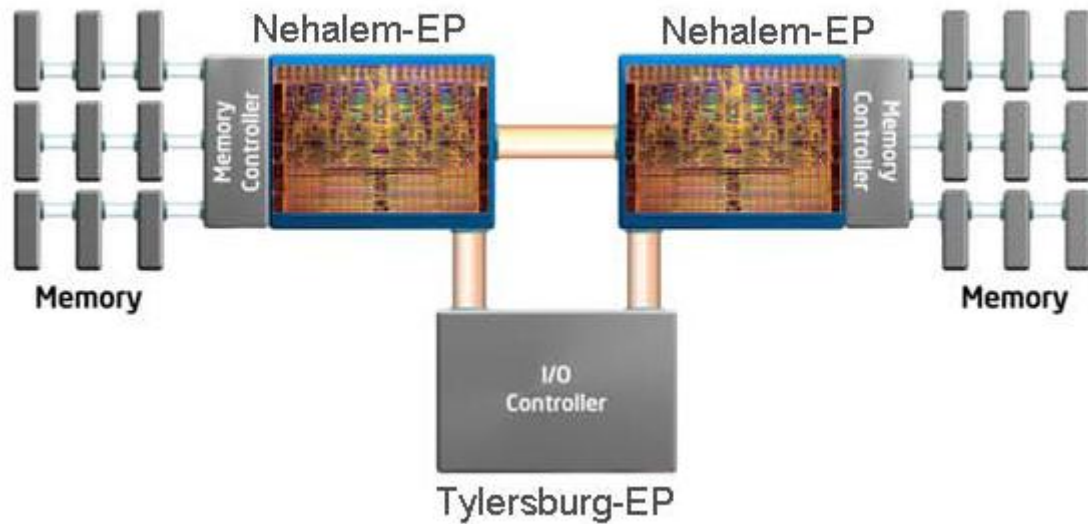
L1 - L3 などの階層化されたキャッシュの場合、L2 キャッシュに記憶されているデータを L3 キャッシュにも置くかどうかで 3 種類に分類されます。1 つは、L2 キャッシュに置かれたデータもそのまま L3 キャッシュに保持する「Inclusive Cache」、もう 1 つは、L2 キャッシュに置かれたデータは L3 キャッシュには保持しないが、データのセットとしては、L2 キャッシュに入らないものを L3 キャッシュに保持しておく「Non-Inclusive Cache」です。最後の 1 つは、必要なデータセットをすべて L2 キャッシュに置き、不要になるまでは L3 キャッシュへ移さない「Exclusive Cache」です。インテルは、NCID(Non-inclusive Cache, Inclusive Directory)とよぶ方式のキャッシュアーキテクチャの研究を発表しており、これは、キャッシュするデータは Non-Inclusive とするが、キャッシュデータと実メモリの対応を記憶する Directory は L3 キャッシュ側で共有するものです。これにより、L2 キャッシュに含まれるデータは、L3 キャッシュに記憶されないために、利用効率が上がり、外部からのスヌープに対しては、L3 キャッシュ側のディレクトリを使ってスヌープフィルタリング対応が可能になるとインテルは説明しています。

今回の Nehalem では、「Inclusive Cache」アーキテクチャを採用しており、2 次キャッシュに置かれたデータもそのまま 3 次キャッシュに保持されます。このため、プロセッサは L3 キャッシュに存在しないデータについては、L1 と L2 に存在しないことになり、他のプロセッサからのスヌープ(データの同一性を確保するためにキャッシュ状態をお互いにチェックすること)でのメリットがあります。

## メモリコントローラ

Nehalem の”Un-Core”には、DDR3 メモリのコントローラが搭載されています。最初

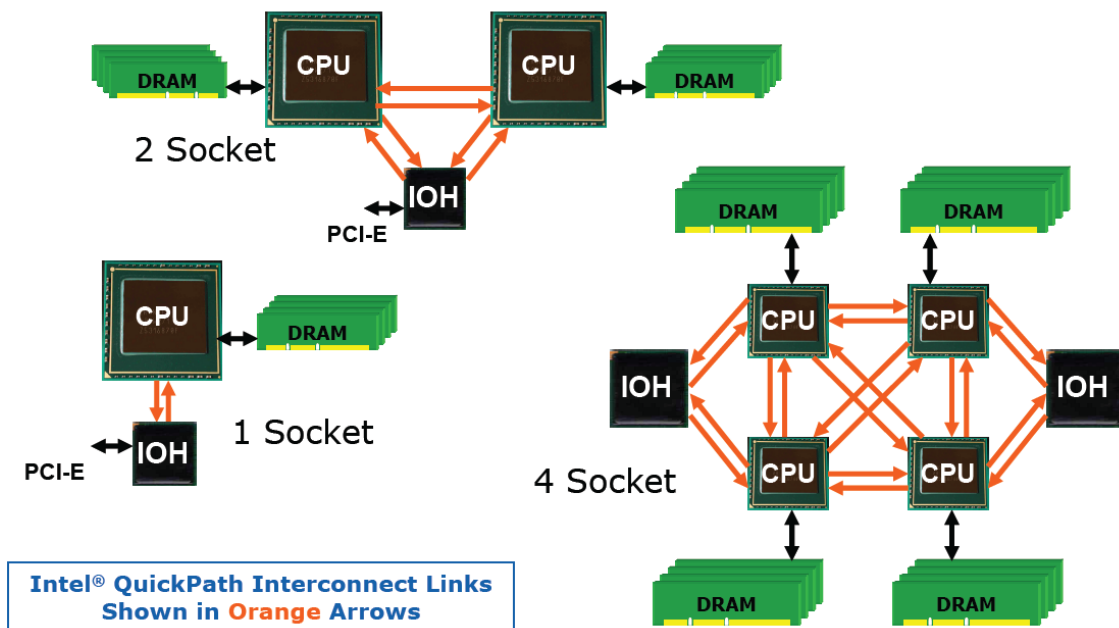
の Nehalem では、3本のメモリチャンネルが搭載されています。最大の性能を得るには、同一容量の DDR3 DIMM を3枚、各チャンネルに搭載する必要があります。1333MHz の DDR3 メモリを搭載した場合のピークバンド幅は、 $1333\text{MHz} \times 8\text{Bytes} (64\text{bit}) \times 3 = 32\text{GB/sec}$  になります。



## QPI

インテルがメモリコントローラをプロセッサに搭載したことにより、プロセッサ間的高速インターコネクが必要になっています。このためのインターコネクとして、Quick Path Interconnect (QPI) が開発され、Nehalem が最初の QPI 搭載のマイクロプロセッサになっています。

各 QPI リンクは、双方向でリンクあたり 6.4 GT/s の速度で、各リンクは、2 バイトのバス幅を持つため、リンクあたり、片方向で 12.8GB/s、双方向で 25.6GB/s のバンド幅となります。

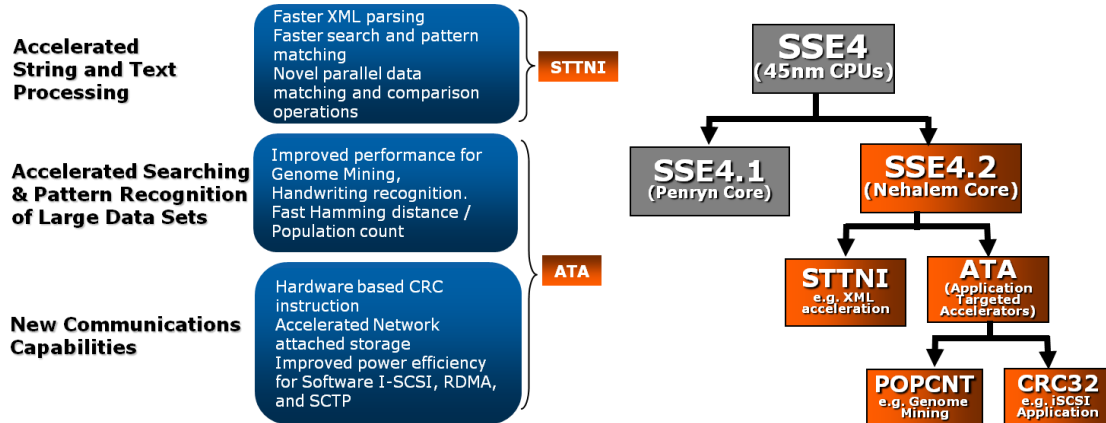


複数プロセッサが各メモリを持つことから、プラットフォーム的には、NUMA システムとして構成されることとなります。ただ、2 プロセッサ構成の場合でも、リモートメモリへのアクセスレイテンシは、従来の FSB でのメモリレイテンシ未満でのアクセスが可能ということです。

### 新しい命令セット

Penryn は、SSE4 の命令セットを拡張し、SSE4.1 を規定しています。Nehalem では、更に幾つかの命令が追加され、SSE4.2 となっています。SSE4.1 は 47 命令、SSE4.2 は 7 命令となっています。この SSE4.2 で追加された命令をインテルは、STTNI/ATA という呼び方をしています。STTNI 命令は、XMM レジスタを使って高速な文字列処理を行うためのものであり、ATA では、CRC 計算と POPCNT(ビットマスクの比較)を行うことが可能となっています。





今回、インテルは、Advanced Vector Extensions (AVX) と呼ばれる SSE の 256 ビットへの拡張も発表しています。現在の SSE が 128 ビットでの処理を行っているのに際して、256 ビットで処理することで、浮動小数点演算の性能などを 2 倍に引き上げることを目標としています。

## Key Intel® Advanced Vector Extensions (Intel® AVX) Features

### KEY FEATURES

### BENEFITS

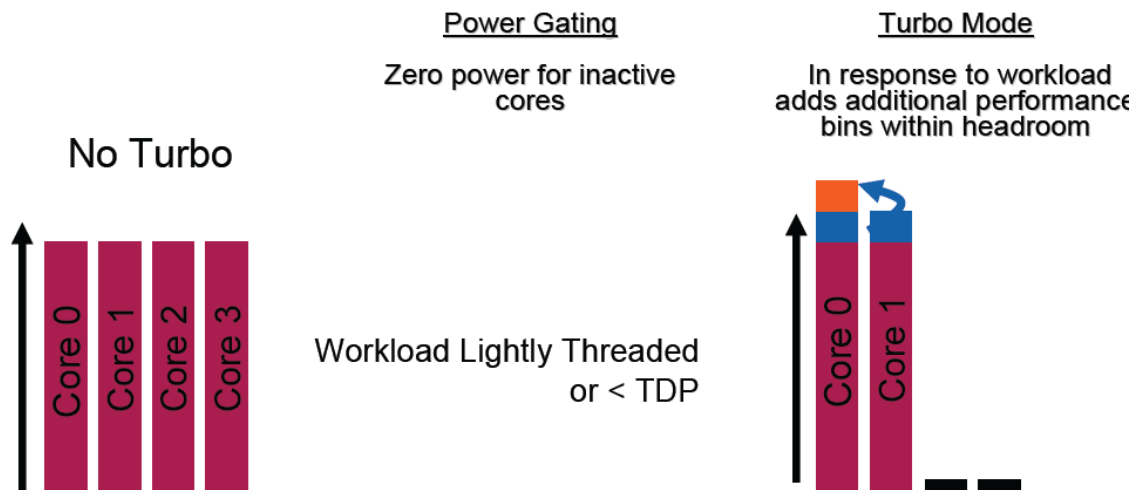
- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>• Wider Vectors                             <ul style="list-style-type: none"> <li>- Increased from 128 bit to 256 bit</li> </ul> </li> </ul>  | <ul style="list-style-type: none"> <li>• Up to 2x peak FLOPs (floating point operations per second) output with good power efficiency</li> </ul> |
| <ul style="list-style-type: none"> <li>• Enhanced Data Rearrangement                             <ul style="list-style-type: none"> <li>- Use the new 256 bit primitives to broadcast, mask loads and permute data</li> </ul> </li> </ul> | <ul style="list-style-type: none"> <li>• Organize, access and pull only necessary data more quickly and efficiently</li> </ul>                   |
| <ul style="list-style-type: none"> <li>• Three and four Operands, Non Destructive Syntax                             <ul style="list-style-type: none"> <li>- Designed for efficiency and future extensibility</li> </ul> </li> </ul>     | <ul style="list-style-type: none"> <li>• Fewer register copies, better register use for both vector and scalar code</li> </ul>                   |
| <ul style="list-style-type: none"> <li>• Flexible unaligned memory access support</li> </ul>  | <ul style="list-style-type: none"> <li>• More opportunities to fuse load and compute operations</li> </ul>                                       |
| <ul style="list-style-type: none"> <li>• Extensible new opcode (VEX)</li> </ul>   | <ul style="list-style-type: none"> <li>• Code size reduction</li> </ul>  |

Turbo Mode

この機能は、既にモバイル版の Penryn で実装されたものであり、モバイル Penryn では、クロックを 1 ステップ

アップさせることが可能になっています。インテルは、この技術を、「Intel Turbo Boost Technology」と呼び、Nehalem では更に強化を行っています。ターボモードは、CPU の TDP(Thermal Design Power:熱設計消費電力)スペックに対して、CPU の動作状況から余裕がある場合に、アクティブな CPU コアに割り振るものです。これによって、アクティブな CPU コアの消費電力が増えても、プロセッサ全体では、TDP 以下での運用を可能とします。

Penryn はデュアルコアの片方のコアがスリープした時だけ、もう片方の CPU コアの動作周波数を引き上げていましたが、Nehalem では 4 個のコアのさまざまなスリープ/アクティブの組み合わせに応じて周波数を引き上げることが可能となります。この他、CPU のワークロードが軽く各 CPU コアの実際の電力消費が低い場合、TDP リミットまで周波数を引き上げることが出来ます。つまり、Nehalem では、ターボモードによって、パフォーマンスが重要なデスクトップ&サーバーで、より多くの機会に、より大きなパフォーマンスアップが得られることとなります。



Nehalem プロセッサでは、少なくとも 1 ステップ(133MHz)のクロックアップは可能で、更に TDP に余裕がある場合には、更にもう 1 ステップのクロックアップが可能になります。実際、どの程度までのクロックアップが可能であるかは、示されていません。

## 製品クロック

インテルは現時点では、Nehalem の価格とその動作周波数を発表していません。ただ、現状では、次のような仕様になることが予想されています。

Core i7 ブランドの Nehalem は、2.66GHz、2.93GHz、3.2GHz のクロックで、全て 8MB L3 キャッシュを持つクアッドコアプロセッサになると予想されます。ターボモードでは、最大 266MHz のクロックアップが可能であり、ケースによっては 133MHz のクロックアップになります。

性能については、多くの予想でサーバ分野で利用されるアプリケーションでの性能向上が期待される一方、デスクトップなどではそれほどの性能向上は期待できないかもしれないと予想されています。

マルチスレッドの性能は、Nehalem ではかなりの性能向上が期待できます。

Nehalem は相対的に AMD プロセッサに対して、弱いと指摘されているサーバ向け分野で求められる機能や性能向上に重点をおいた設計になっており、この製品において、恐らく唯一のインテルの弱点が克服されることになることが予想されます。同時に、Nehalem は、省電力化ということでは、インテルの Atom プロセッサが取り組んでいるような機能も積極的に採用しており、設計思想的には、より電力消費を抑える方向にも進んでいます。



〒102-0083 東京都千代田区麹町 3-5-2 BUREX 麹町 8F  
電話:03-5875-4718 FAX:03-3237-7612 [www.sstc.co.jp](http://www.sstc.co.jp)

スケーラブルシステム株式会社では、IT 技術と HPC システムに関する様々な調査レポートを発行しています。

社名、製品名などは、一般に各社の商標または登録商標です。

Copyright Scalable Systems Co., Ltd. , 2008. Unauthorized use is strictly forbidden.  
無断での引用、転載を禁じます。 2008/08/25