Memory Sharing and The Compute Architecture of Intel® Processor Graphics Gen8

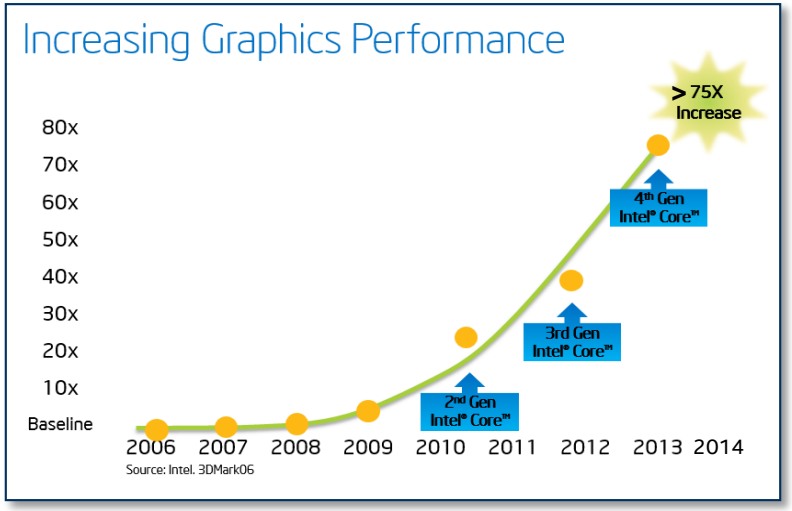
Stephen Junkins
GPU Compute Architect, Principal Engineer, Intel Corporation
(Visual & Parallel Computing Group)

Agenda

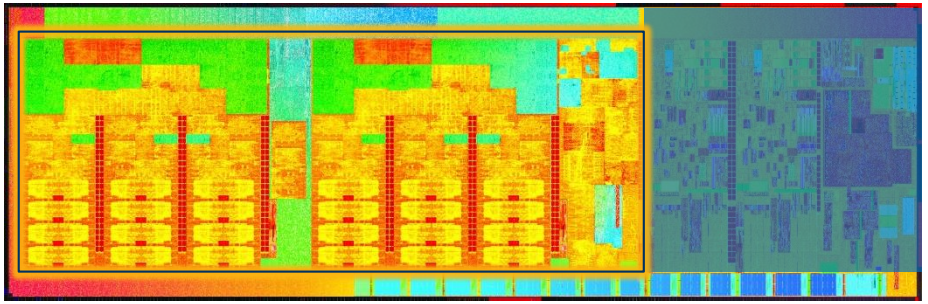
- Introduction
- Compute architecture:
  - Execution units
  - Subslices, slices, products
  - Chip level architecture
- Memory:
  - Shared physical memory
  - Shared virtual memory & coherency
  - Application examples
- Summary

Intel® Processor Graphics?

- Intel® Processor Graphics: 3D Rendering, Media, *and* *Compute*
- Discrete class performance but… integrated on-die for true heterogeneous computing, SoC power efficiency, and a fully connected system architecture
- Some products are near TFLOPS performance
- The foundation is a highly threaded, data parallel compute architecture
- Today: focus on compute components of Intel Processor Graphics **Gen8**

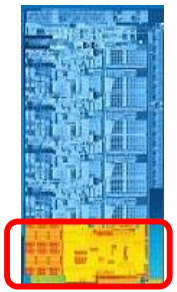
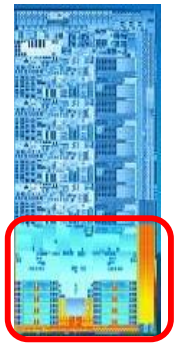
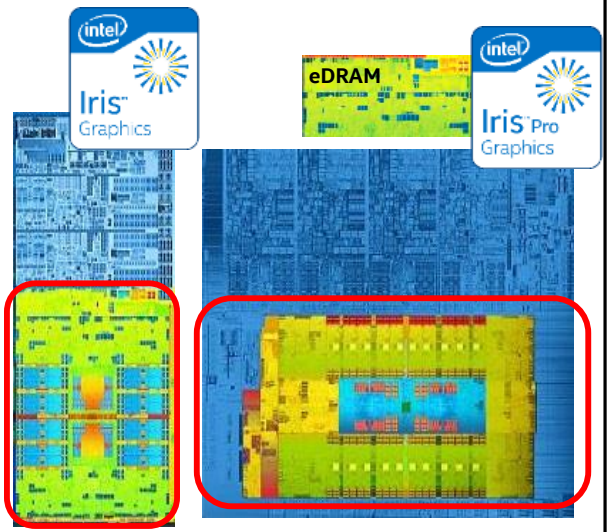
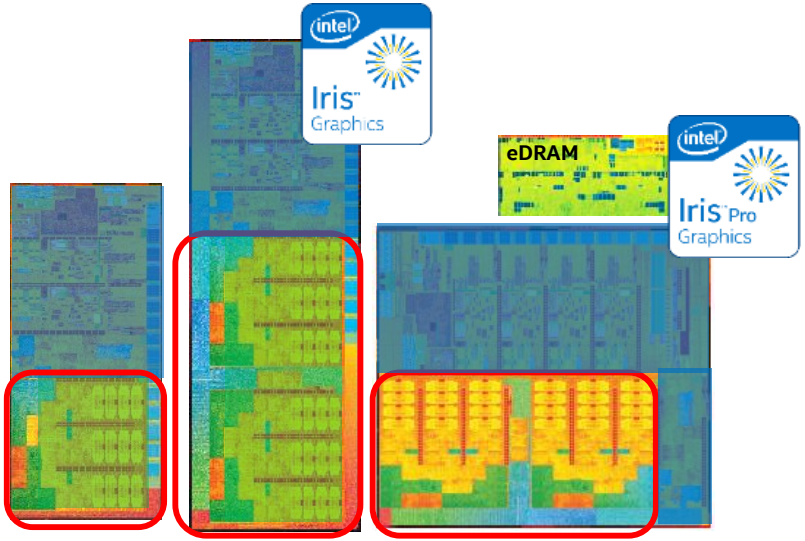


Intel® Core™ i5 with Iris graphics 6100:



Intel Processor Graphics is a key Compute Resource

Processor Graphics is a Key Intel Silicon Component

Processor graphics gen6	Processor graphics gen7	Processor graphics gen7.5	Processor graphics gen8 ...
 Intel HD Graphics	 Intel HD Graphics		
Intel 2nd Gen Core™	Intel 3rd Gen Core™	Intel® 4th Gen Core™	Intel® 5th Gen Core™

Example OEM Products with Processor Graphics

Apple* Macbook* Pro 15”

Sony* Vaio* Tap 21

Gigabyte* Brix* Pro

Toshiba* Encore* 2 Tablet

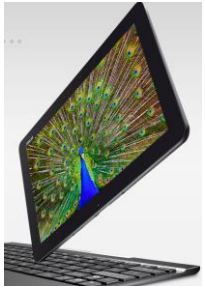
Microsoft* Surface* Pro 3

Apple Macbook Pro 13”

JD.com – Terran Force
Clevo* Niagara*

Asus MeMO* Pad 7*

Lenovo* Miix* 2



Asus* Transformer Pad*

Apple iMac* 21.5”

Zotac* ZBOX* EI730

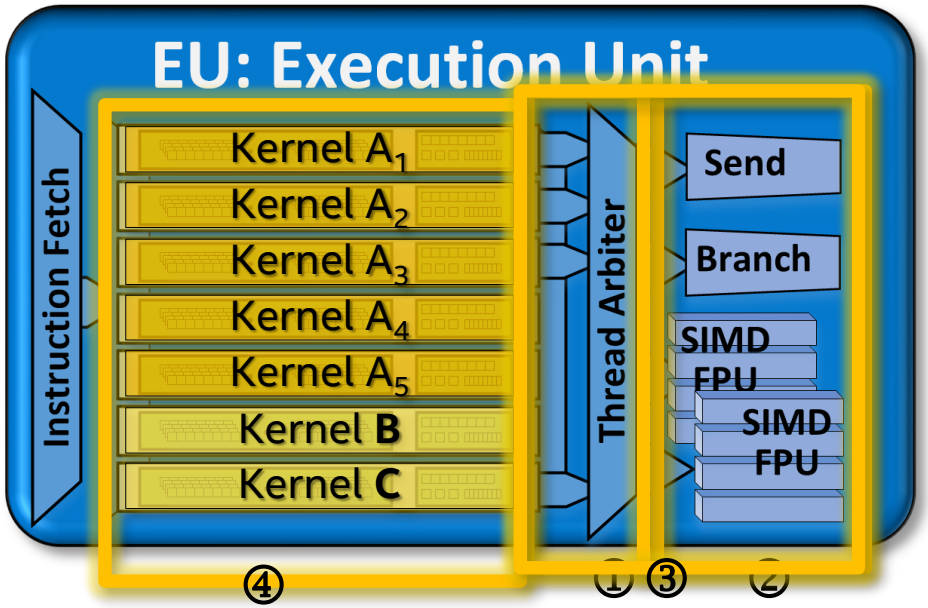
Asus Zenbook Infinity*

The Graphics Architecture for many OEM DT, LT, 2:1, tablet products

Agenda

- Introduction
- Compute Architecture:
  - Execution units
  - Subslices, slices, products
  - Chip level architecture
- Memory:
  - Shared physical memory
  - Shared virtual memory & coherency
  - Application examples
- Summary

EU: Multi-threaded execution



➢ 7 HW threads, 4K register file each

① Combination of **SMT** & **IMT**

- Thread Arbiter picks instructions to run from runnable thread(s)

② Dispatches instruction to appropriate functional unit

③ Each cycle: can co-issue multiple instructions, from up to four different threads

④ Each thread executes a unique kernel

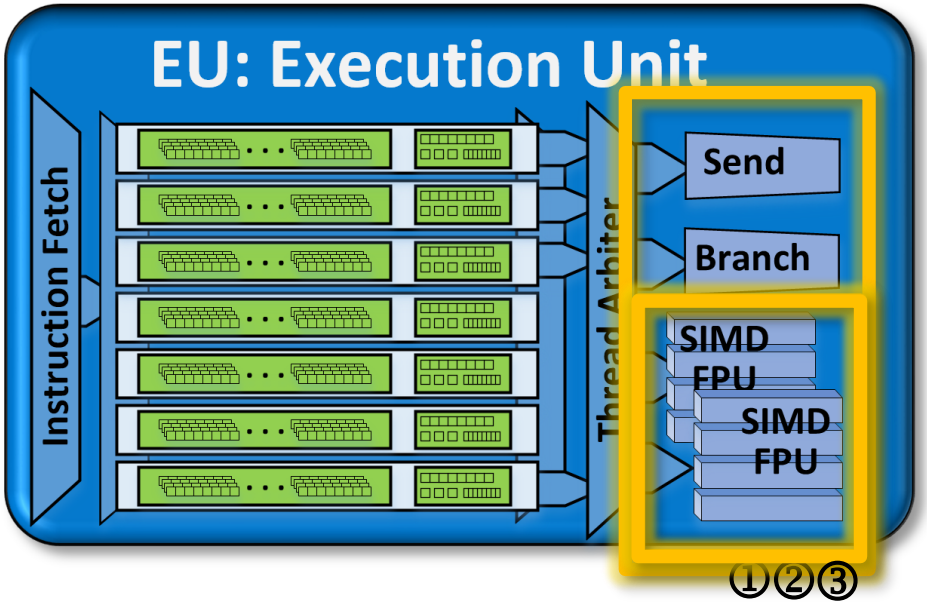
- Different instances of same src kernel

  Or

- Different src or new kernels

SMT – Simultaneous Multi-Threading
IMT – Interleaved Mutli-Threading

EU: Instructions & FPUs



① Instructions:

- 2 or 3 src registers, 1 dst register
- Instructions are ***variable width SIMD.***
- Logically programmable as 1, 2, 4, 8, 16, 32 wide SIMD
- SIMD width can change back to back w/o penalty
- Optimize register footprint, compute density

② 2 Arithmetic, Logic, Floating-Pt Units

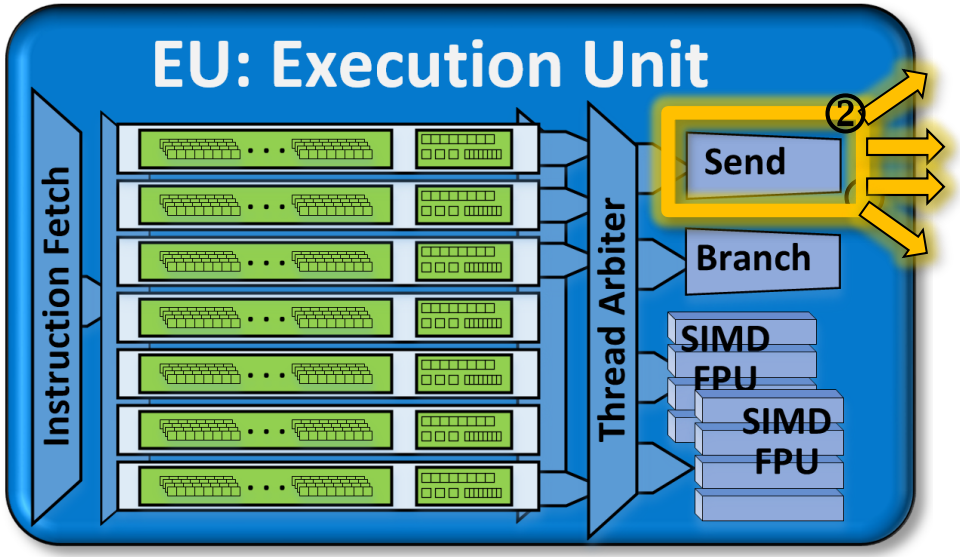
- *Physically* 4-wide SIMD, 32-bit lanes

③ Min FPU instruction latency is 2 clocks

- SIMD-1, 2, 4, 8 float ops: 2 clocks
- SIMD-16 float ops: 4 clocks
- SIMD-32 float ops: 8 clocks

FPUs are fully pipelined across threads: instructions complete every cycle.

EU: Universal I/O Messages



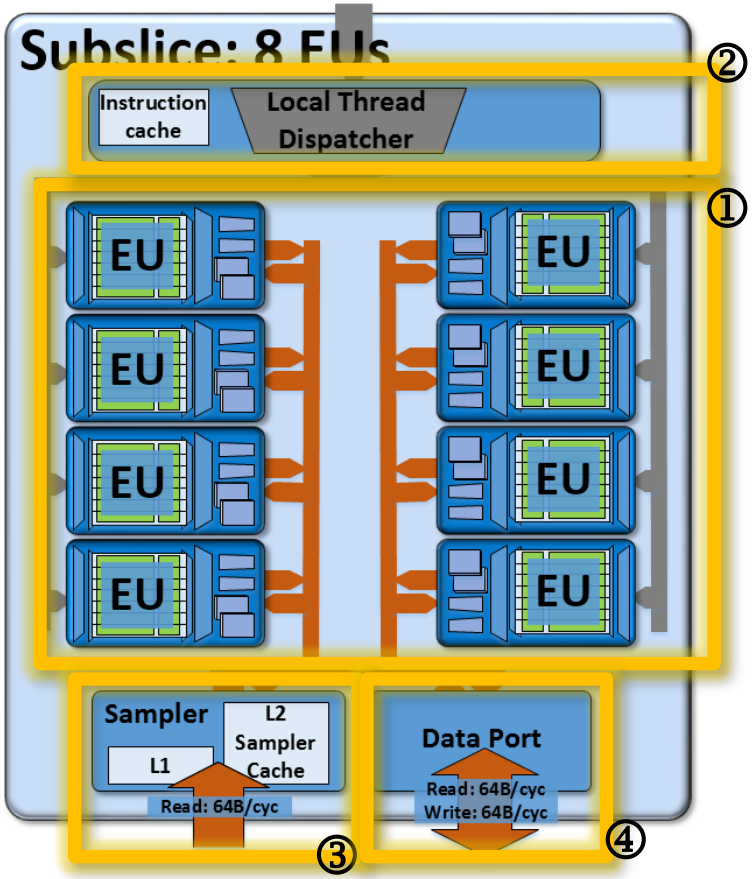
The Messaging Unit

① **Send** is the universal I/O instruction

② Many send message types:

- Mem stores/reads are messages
- Mem scatter/gathers are messages
- Texture Sampling is a message with u,v,w coordinates per SIMD lane
- Messages used for synchronization, atomic operations, fences etc.

Subslice: An Array of 8 EU's



Each: Subslice

① Eight Execution Units

② Local Thread Dispatcher & Inst $

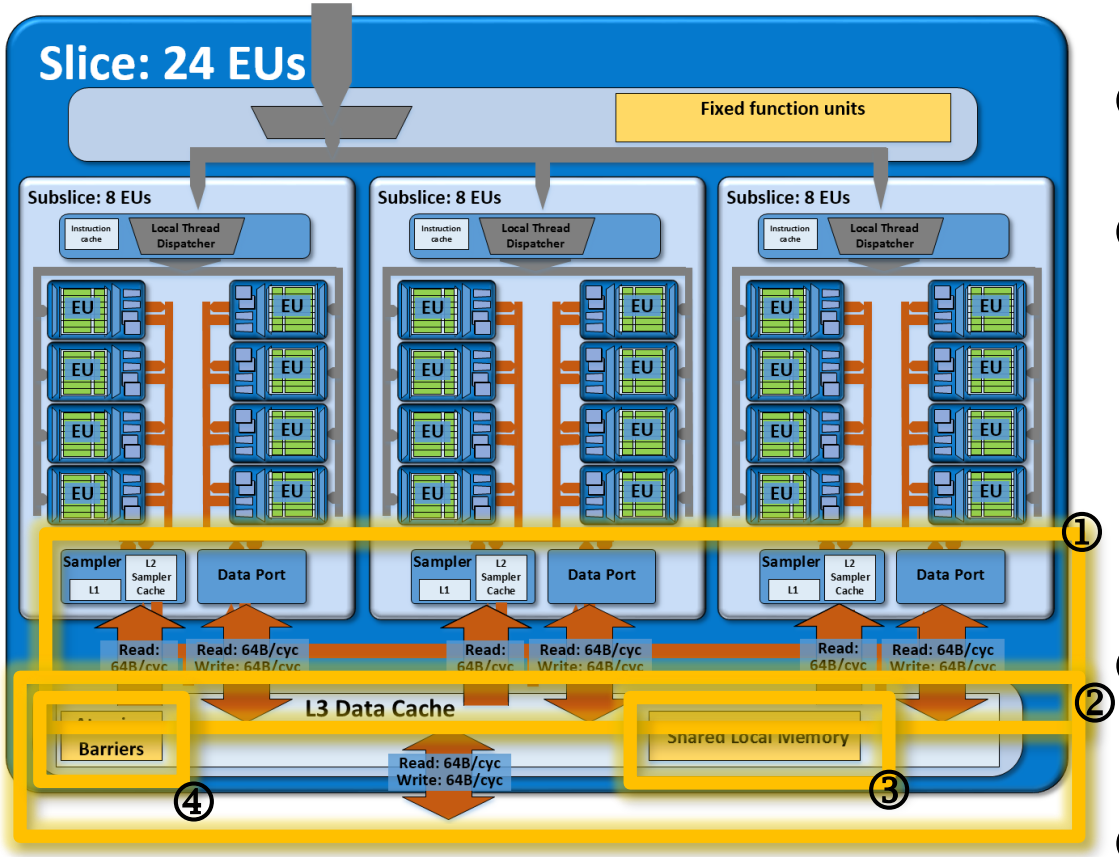
③ Texture/Image Sampler Unit:

- Includes dedicated L1 & L2 caches
- Dedicated logic for dynamic texture decompression, texel filtering, texel addressing modes
- 64 Bytes/cycle read bandwidth

④ Data Port:

- General purpose load/store S/G Mem unit
- Memory request coalescence
- 64 Bytes/cycle read & write bandwidth

Slice: 3x Subslices



Each Slice: 3 x 8 = 24 EU's

- 3x8x7 = 168 HW threads
- 3x8x7xSIMD32 = 5376 kernel insts

① Dedicated interface for every sampler & data port

② Level-3 (L3) Data Cache:

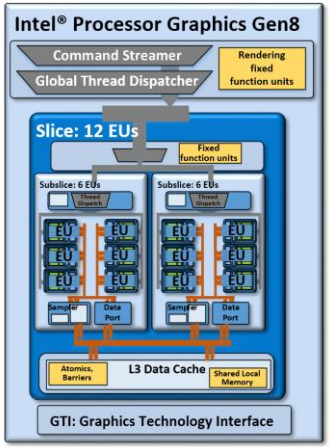
- Typically 384 KB / slice in multiple banks, (allocation sizes are driver reconfigurable)
- 64 byte cachelines
- Monolithic, but distributed cache
- 64 bytes/cycle read & write
- Scalable fabric for larger designs

③ Shared Local Memory:

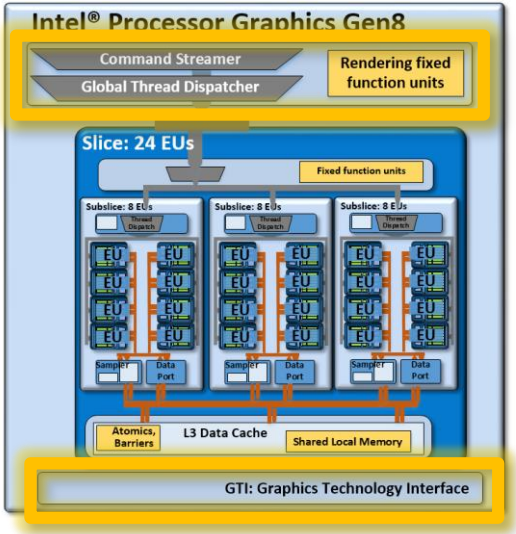
- 64 KB / subslice
- More highly banked than rest of L3

④ Hardware Barriers, 32bit atomics

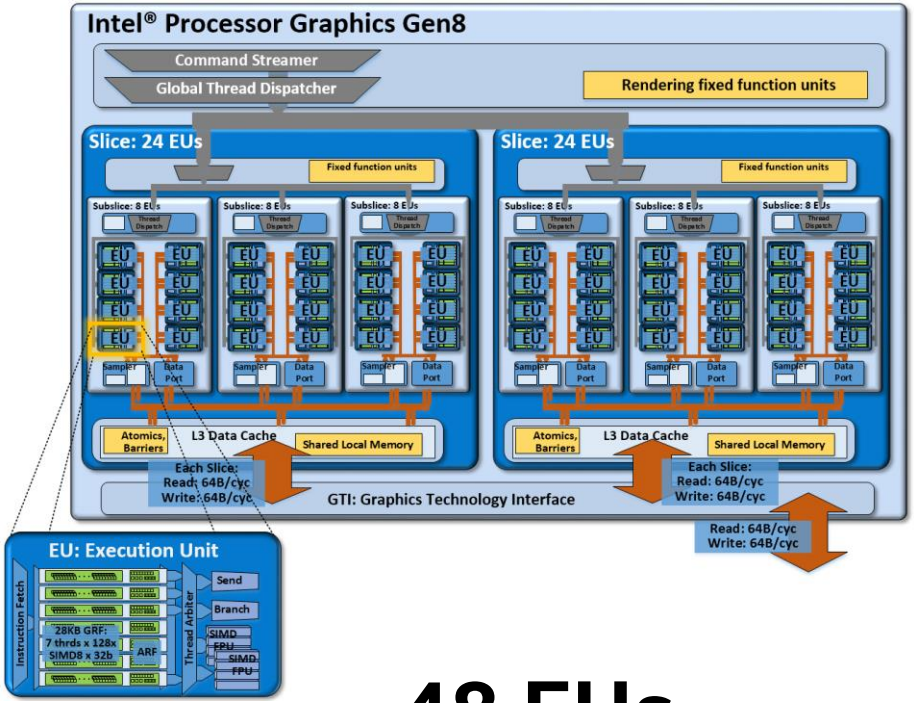
Product Configuration Examples



12 EUs

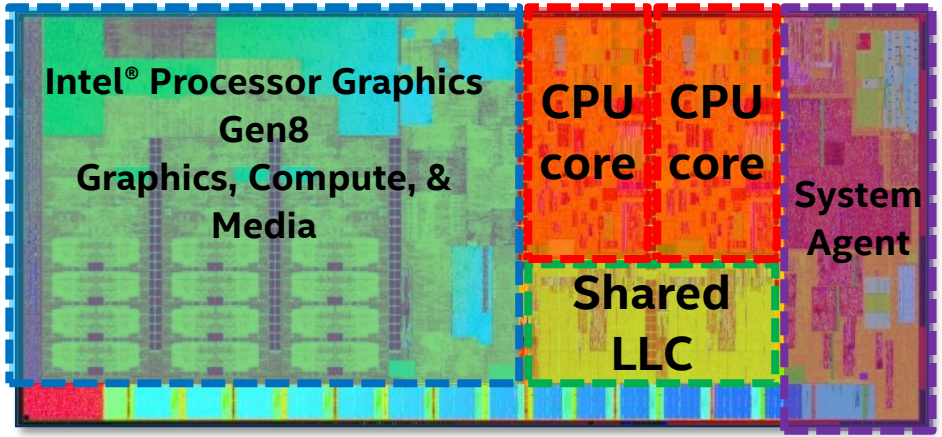


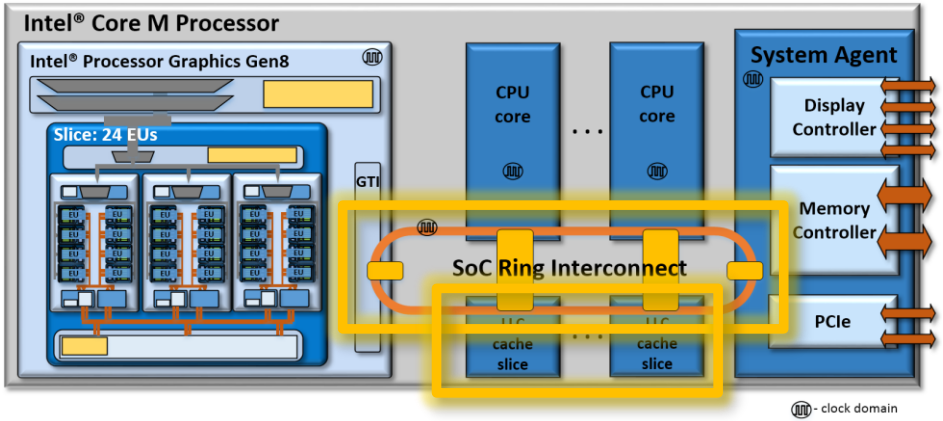
24 EUs



48 EUs

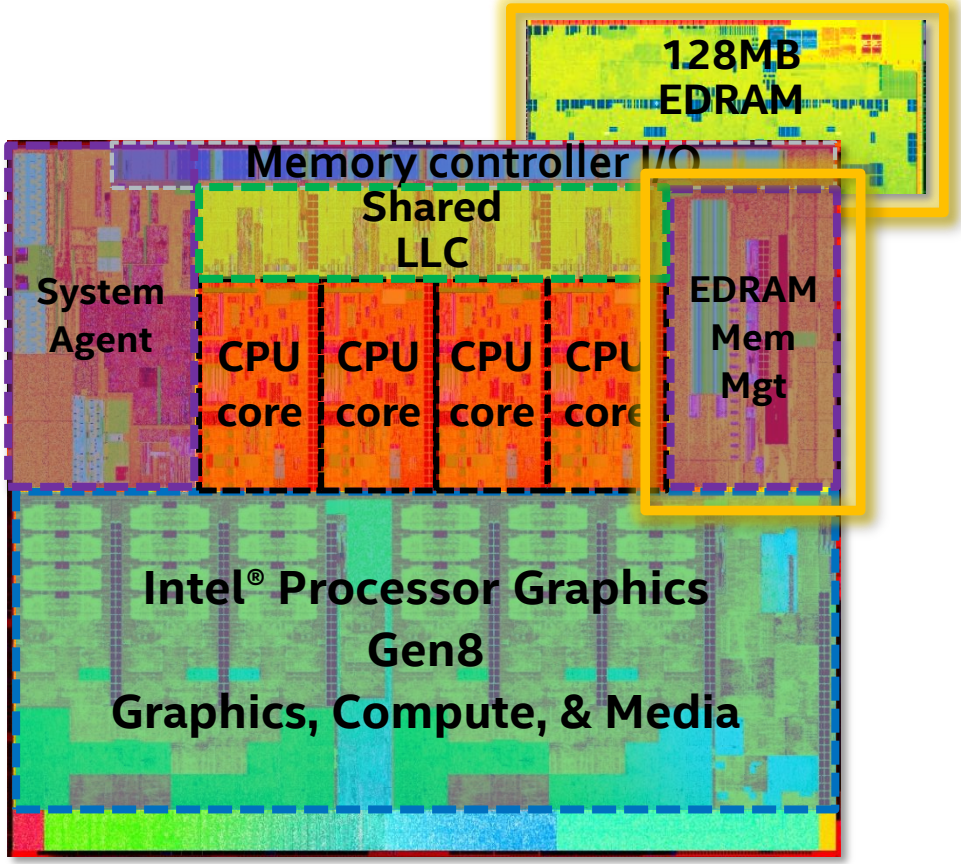
Chip Level Architecture

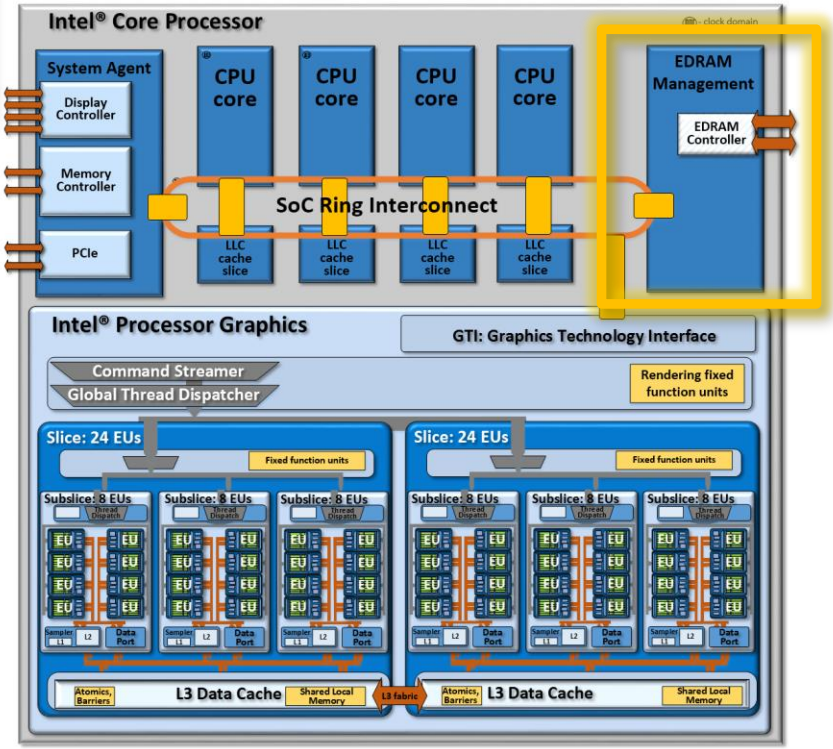




- Ring Interconnect:
  - Dedicated “stops”: each CPU Core, Graphics, & System Agent
  - Bi-directional, 32 Bytes wide
- Shared Last Level Cache (LLC)
  - Both GPU & CPU cores
  - 2-8MB, depending on product
  - Inclusive
- Optimized for CPU & GPU coherency
  - Address hashing to multiple concurrent LLC request queues
  - LLC avoids needless snoops “upwards”

Chip Level Architecture :
4 CPU cores & Iris Pro Graphics: 48 EUs, & EDRAM



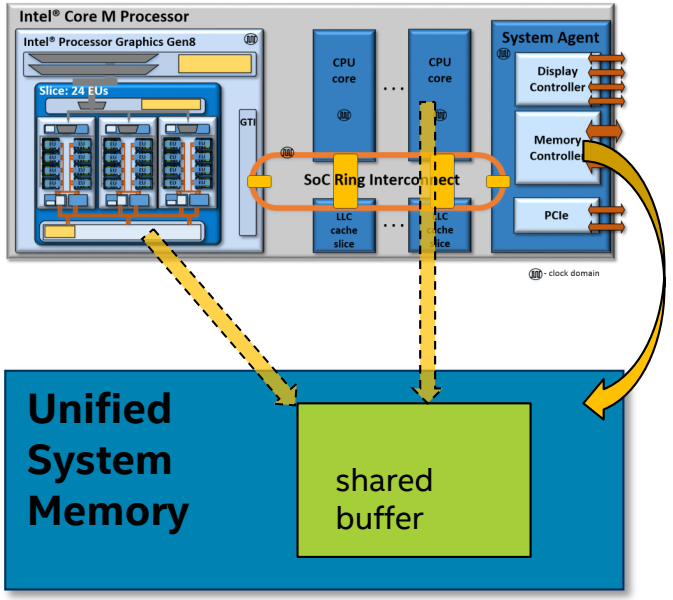


Agenda

- Introduction
- Compute Architecture:
  - Execution Units
  - Subslice, Slices, Products
  - Chip Level Architecture
- Memory:
  - Shared physical memory
  - Shared virtual memory & coherency
  - Application examples
- Summary

Shared Physical Memory: a.k.a. Unified Memory Architecture (UMA)

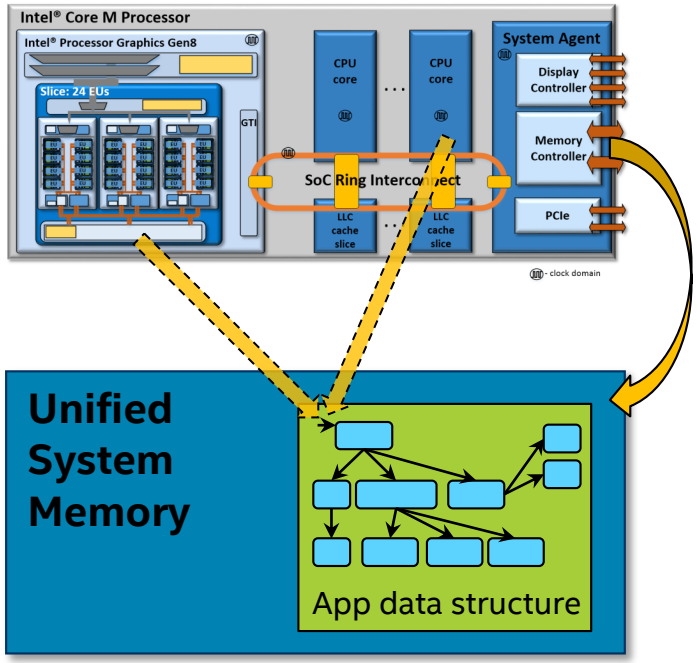
- Long History: …Gen2…Gen6, Gen7, Gen7.5, Gen8 all employed shared physical memory
- No need for additional GDDR memory package or controller. Conserves overall system memory footprint & system power
- Intel® Processor Graphics has full performance access to system memory
- “Zero Copy” CPU & Graphics data sharing
- Enabled by buffer allocation flags in OpenCL™, DirectX*, etc.



Shared Physical Memory means “Zero Copy” Sharing

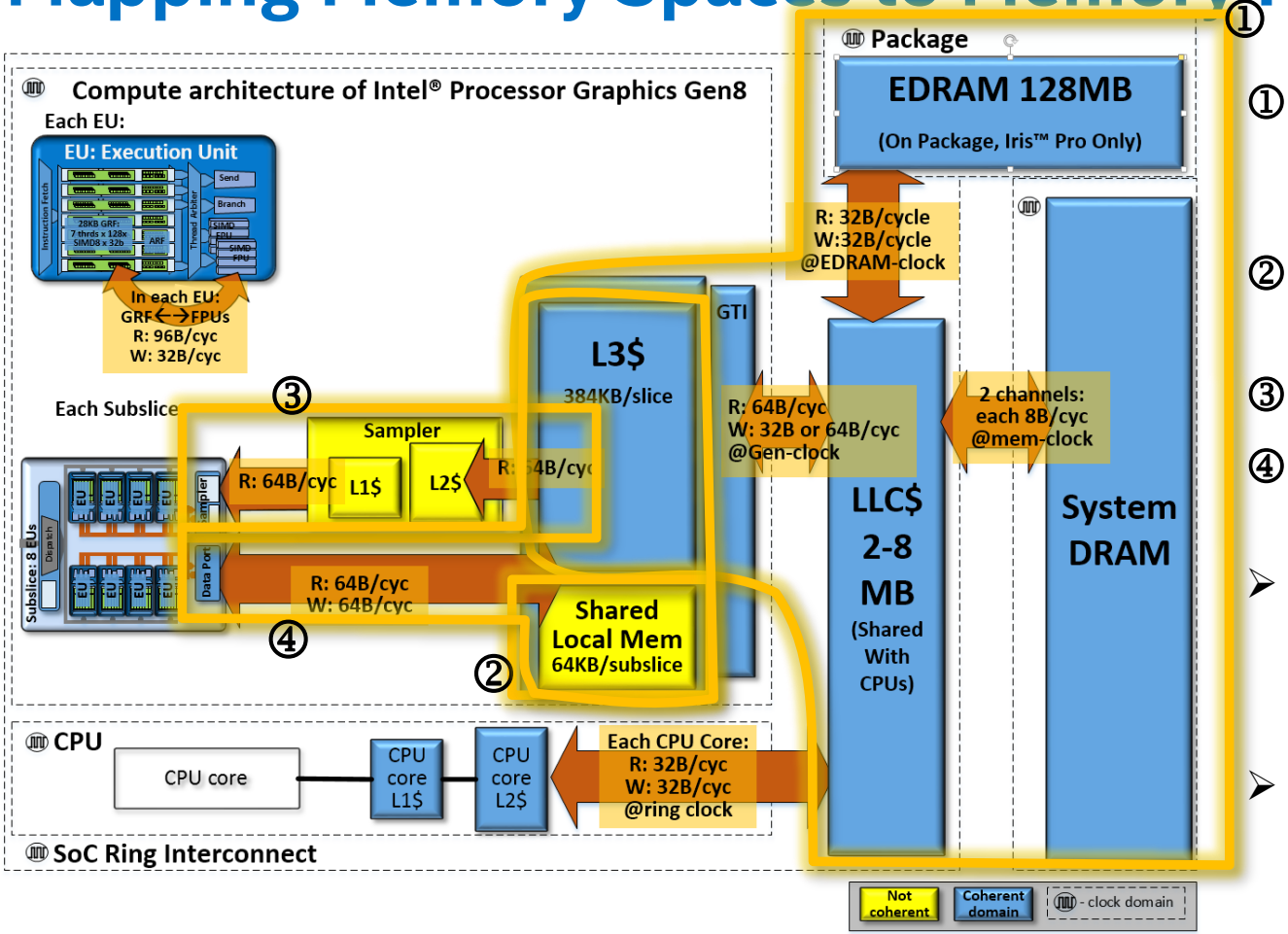
Shared Virtual Memory

- Significant feature, new in Gen8
- Seamless sharing of pointer rich data-structures in a shared virtual address space
- Hardware-supported byte-level CPU & GPU coherency, cache snooping protocols…
- Spec'd Intel® VT-d IOMMU features enable heterogeneous virtual memory, shared page tables, page faulting.
- Facilitated by OpenCL™ 2.0 Shared Virtual Memory:
  - Coarse & fine grained SVM
  - CPU & GPU atomics as synchronization primitives



Shared Virtual Memory enables seamless pointer sharing

Mapping Memory Spaces to Memory Hierarchy



① Cached memory hierarchy supporting global, constant, and image data

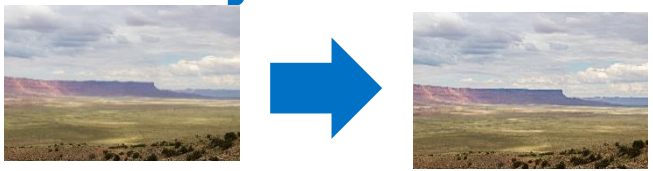
② Shared (local) memory reads & writes

③ Image reads

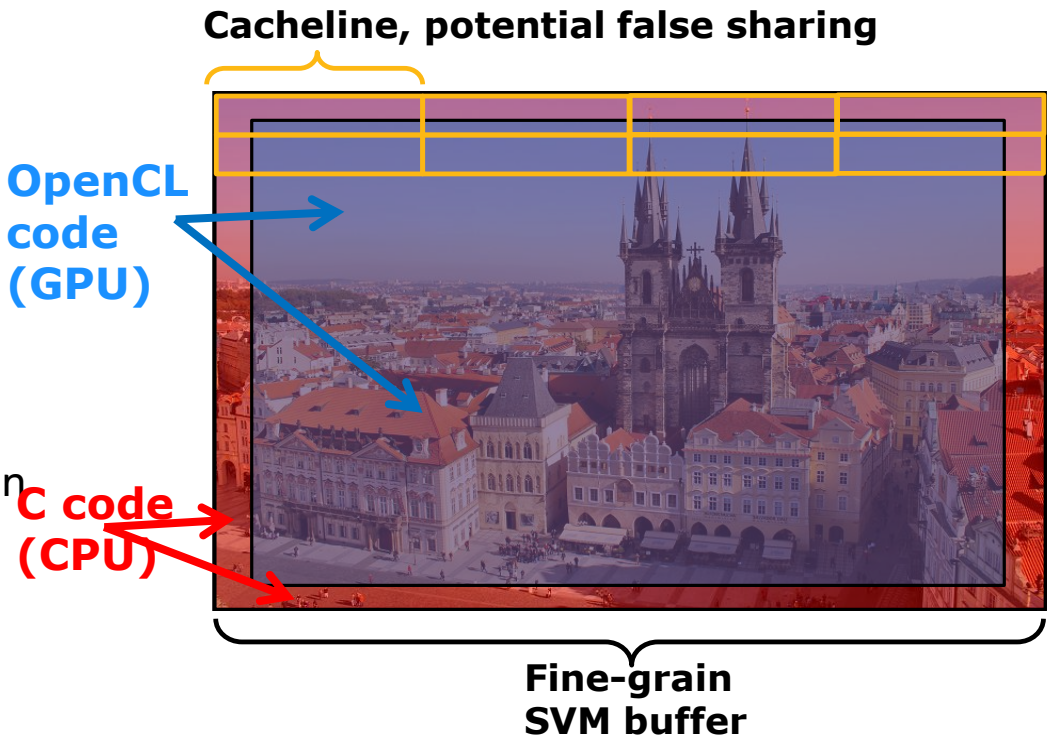
④ Buffer & local reads & writes. Also image writes

- All memory caches are globally **coherent** (except for sampler & shared local memory)
- CPU & GPU sharing at full bandwidth of LLC

SVM: Cyberlink Photo Director's "Clarify Effect"

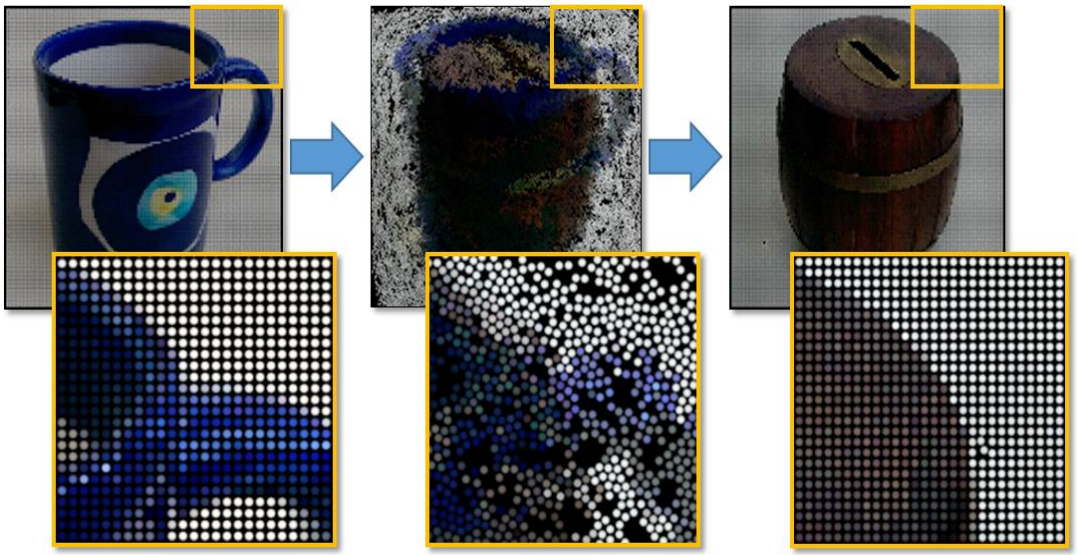


- Concurrent CPU & GPU computes applied to a single coherent buffer
- Border pixels have different algorithm, conditional degrades GPU efficiency
- SVM Implementation:
  - ✓ **CPU** does border
  - ✓ **GPU** does interior, with no conditionals
  - ✓ Seamless, correct sharing, even when cachelines cross border regions



SVM: Behavior Driven Crowd Simulation (UNC collab)

- A sea of autonomous "agents" from start to goal positions. Complex collisions and interactions in transit. (Visualized here as pixels.)
- C pointer rich agent spatial dynamic data structure developed for multi-core CPU
- SVM Implementation:
  - ✓ Ported quickly to GPU and SVM buffers *without* data-structure re-write
  - ✓ Enables both GPU & multiple CPU to concurrently support computation on single data-structure, plus GPU rendering



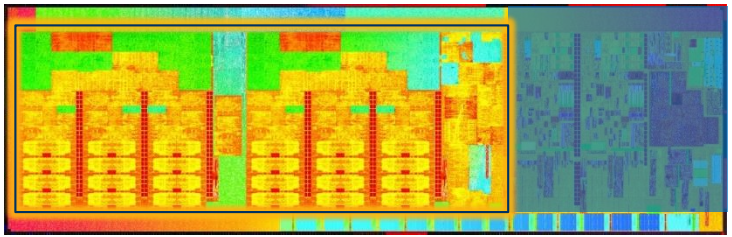
Images courtesy of Sergey Lyalin and UNC. More info: http://gamma.cs.unc.edu/RVO2/.

Agenda

- Introduction
- Compute architecture:
  - Execution units
  - subslices, slices, products
  - Chip level architecture
- Memory:
  - Shared physical memory
  - Shared virtual memory & coherency
  - Application examples
- Summary

Summary

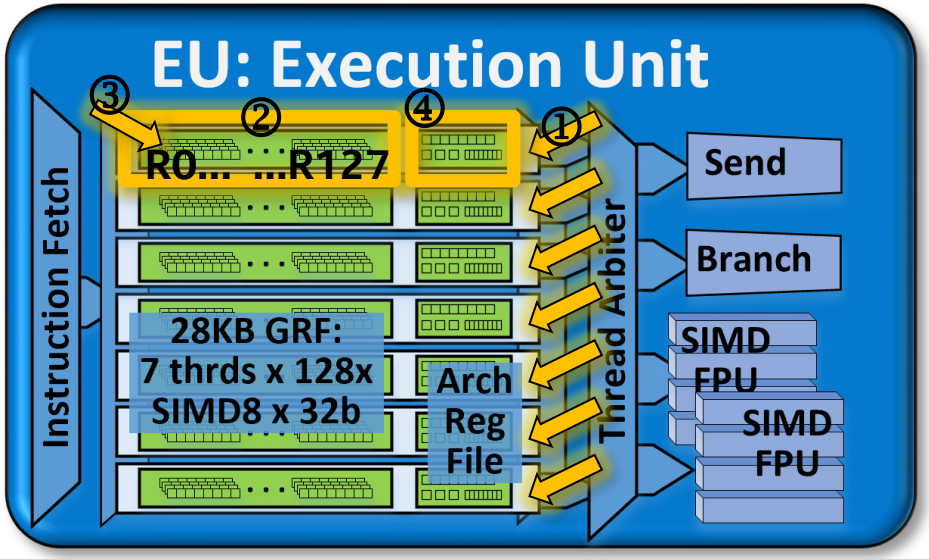
- Intel® Processor Graphics: 3D Rendering, Media, and Compute
- Many products, APIs, & applications using Intel® Processor Graphics for compute
- Gen8 Architecture:
  - Execution Units, Slices, SubSlices, Many SoC product configs
  - Layered memory hierarchy founded shared LLC.
- Shared Physical Memory, Shared Virtual Memory
  - No separate discrete memory, No PCIe bus to GPU.
  - SVM & *real* GPU/CPU cache coherency is here: use it, join us.
- Hint: See more at Intel Developer Forum in 1 week (Aug 18th, 2015)



Intel Processor Graphics: a key platform Compute Resource

Backup

EU: The Execution Unit



① Gen8: Seven hardware threads per EU

② 128 “GRF” registers per thread

- 4K registers/thread or 28K/EU

③ Each “GRF” register :

- 32 bytes wide
  - Eight: 32b floats or 32b integers
  - Sixteen: 16b half-floats or 16b shorts
- Byte addressable

④ Architecture Registers per thread:

- Program Counters
- Accumulators
- Index & predicate registers

Legal Notices and Disclaimers

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.

No computer system can be absolutely secure.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

Statements in this document that refer to Intel's plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel's results and plans is included in Intel's SEC filings, including the annual report on Form 10-K.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

Intel, Core, Iris, Iris Pro, and the Intel logo are trademarks of Intel Corporation in the United States and other countries.

*Other names and brands may be claimed as the property of others.

OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permision by Khronos.

Risk Factors

The above statements and any others in this document that refer to plans and expectations for the second quarter, the year and the future are forward-looking statements that involve a number of risks and uncertainties. Words such as "anticipates," "expects," "intends," "plans," "believes," "seeks," "estimates," "may," "will," "should" and their variations identify forward-looking statements. Statements that refer to or are based on projections, uncertain events or assumptions also identify forward-looking statements. Many factors could affect Intel's actual results, and variances from Intel's current expectations regarding such factors could cause actual results to differ materially from those expressed in these forward-looking statements. Intel presently considers the following to be important factors that could cause actual results to differ materially from the company's expectations. Demand for Intel's products is highly variable and could differ from expectations due to factors including changes in business and economic conditions; consumer confidence or income levels; the introduction, availability and market acceptance of Intel's products, products used together with Intel products and competitors' products; competitive and pricing pressures, including actions taken by competitors; supply constraints and other disruptions affecting customers; changes in customer order patterns including order cancellations; and changes in the level of inventory at customers. Intel's gross margin percentage could vary significantly from expectations based on capacity utilization; variations in inventory valuation, including variations related to the timing of qualifying products for sale; changes in revenue levels; segment product mix; the timing and execution of the manufacturing ramp and associated costs; excess or obsolete inventory; changes in unit costs; defects or disruptions in the supply of materials or resources; and product manufacturing quality/yields. Variations in gross margin may also be caused by the timing of Intel product introductions and related expenses, including marketing expenses, and Intel's ability to respond quickly to technological developments and to introduce new products or incorporate new features into existing products, which may result in restructuring and asset impairment charges. Intel's results could be affected by adverse economic, social, political and physical/infrastructure conditions in countries where Intel, its customers or its suppliers operate, including military conflict and other security risks, natural disasters, infrastructure disruptions, health concerns and fluctuations in currency exchange rates. Results may also be affected by the formal or informal imposition by countries of new or revised export and/or import and doing-business regulations, which could be changed without prior notice. Intel operates in highly competitive industries and its operations have high costs that are either fixed or difficult to reduce in the short term. The amount, timing and execution of Intel's stock repurchase program could be affected by changes in Intel's priorities for the use of cash, such as operational spending, capital spending, acquisitions, and as a result of changes to Intel's cash flows or changes in tax laws. Product defects or errata (deviations from published specifications) may adversely impact our expenses, revenues and reputation. Intel's results could be affected by litigation or regulatory matters involving intellectual property, stockholder, consumer, antitrust, disclosure and other issues. An unfavorable ruling could include monetary damages or an injunction prohibiting Intel from manufacturing or selling one or more products, precluding particular business practices, impacting Intel's ability to design its products, or requiring other remedies such as compulsory licensing of intellectual property. Intel's results may be affected by the timing of closing of acquisitions, divestitures and other significant transactions. A detailed discussion of these and other factors that could affect Intel's results is included in Intel's SEC filings, including the company's most recent reports on Form 10-Q, Form 10-K and earnings release.

Rev. 4/14/15