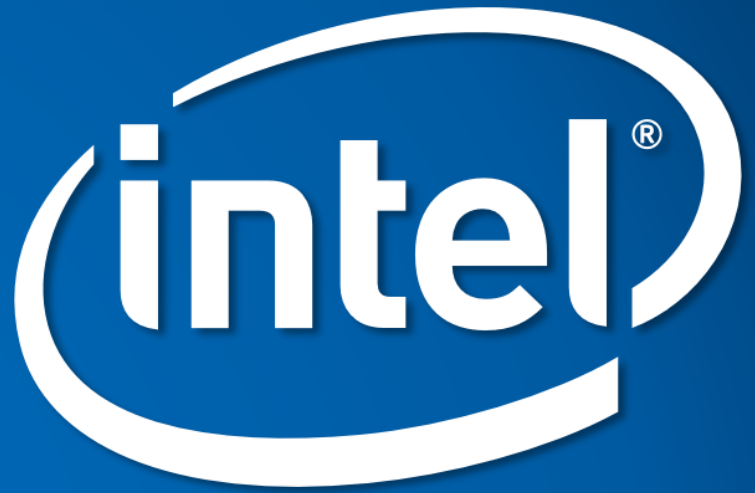


PROGRAMMING GPU

Overview.

Bartosz Boczula
Intel Technology Poland

AGENDA

1. HARDWARE
    1. CPU, GPU, SOC, Memory, TDP
2. DRIVER
3. DIRECTX
4. GPA Live Session

WHAT IS CPU?

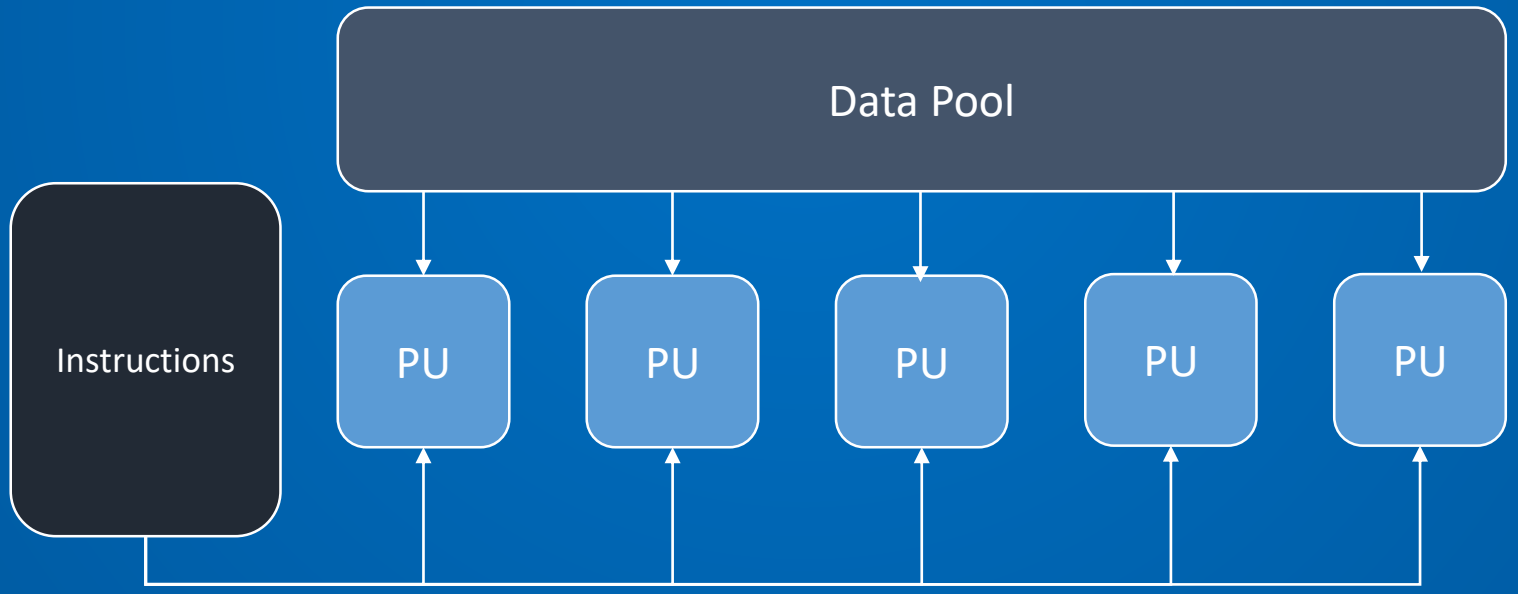
- A processing unit
- Used to executes a computer program
- Wide instruction set
- High frequency, not too much cores
- Big cache, branch prediction

WHAT IS GPU?

- A processing unit, like CPU
- Also executes programs, called **shaders**
- Optimized for 3D graphics computation
- Implements number of graphics primitive operatrions
- Either on separate card or integrated with the CPU
- Highly parallel structure
- Efficient for algorithms where the processing of large block of data is done in parallel

WHAT IS SIMD?

- Architecture class
- Multiple processing units that perform the same operation on multiple data points simultaneously



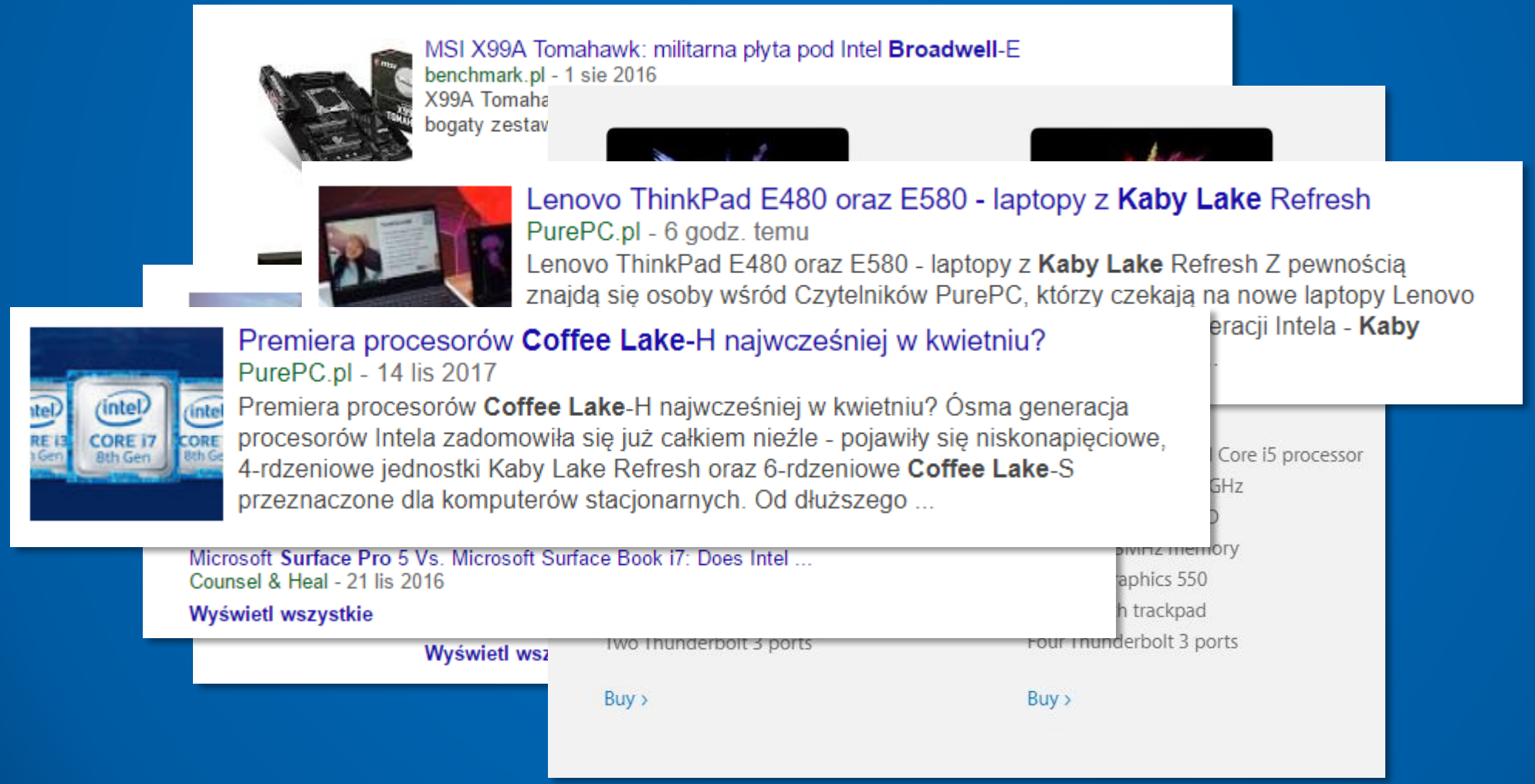
GPU VS. CPU

CPUs composed of few cores and are really good in making single threads go really fast	GPUs are composed of hundreds of cores and are really good in making thousands of threads in aggregate to go really fast
CPUs tend to skip all over the place during execution	GPUs are optimized for taking huge batches of data and performing the same simple linear operation very quickly

GPU VS. CPU

Device	CPU	GPU
 ZenPhone 2	?	?
 PlayStation 4	?	?
 MacBook Pro	?	?
 Infinity LC-900	?	?

WHAT IS SOC?


MSI X99A Tomahawk: militarna płyta pod Intel **Broadwell**-E
benchmark.pl - 1 sie 2016
X99A Tomaha
bogaty zestav
Lenovo ThinkPad E480 oraz E580 - laptopy z **Kaby Lake** Refresh
PurePC.pl - 6 godz. temu
Lenovo ThinkPad E480 oraz E580 - laptopy z **Kaby Lake** Refresh Z pewnością znajdą się osoby wśród Czytelników PurePC, którzy czekają na nowe laptopy Lenovo
eracji Intela - **Kaby**
Premiera procesorów **Coffee Lake**-H najwcześniej w kwietniu?
PurePC.pl - 14 lis 2017
Premiera procesorów **Coffee Lake**-H najwcześniej w kwietniu? Ósma generacja procesorów Intela zadomowiła się już całkiem nieźle - pojawiły się niskonapięciowe, 4-rdzeniowe jednostki Kaby Lake Refresh oraz 6-rdzeniowe **Coffee Lake**-S przeznaczone dla komputerów stacjonarnych. Od dłuższego ...
intel
CORE i7
8th Gen
Microsoft **Surface Pro** 5 Vs. Microsoft Surface Book i7: Does Intel ...
Counsel & Heal - 21 lis 2016
Wyświetl wszystkie
Core i5 processor
GHz
aphics 550
h trackpad
Two Thunderbolt 3 ports
Four Thunderbolt 3 ports
Buy ›
Buy ›

WHAT IS SOC?

2H '15

6TH GEN

SKYLAKE

15%[1]
BETTER PERFORMANCE on SysMark

2H '16

7TH GEN

KABYLAKE

>15%[2]
BETTER PERFORMANCE on SysMark

2H '17*

8TH GEN

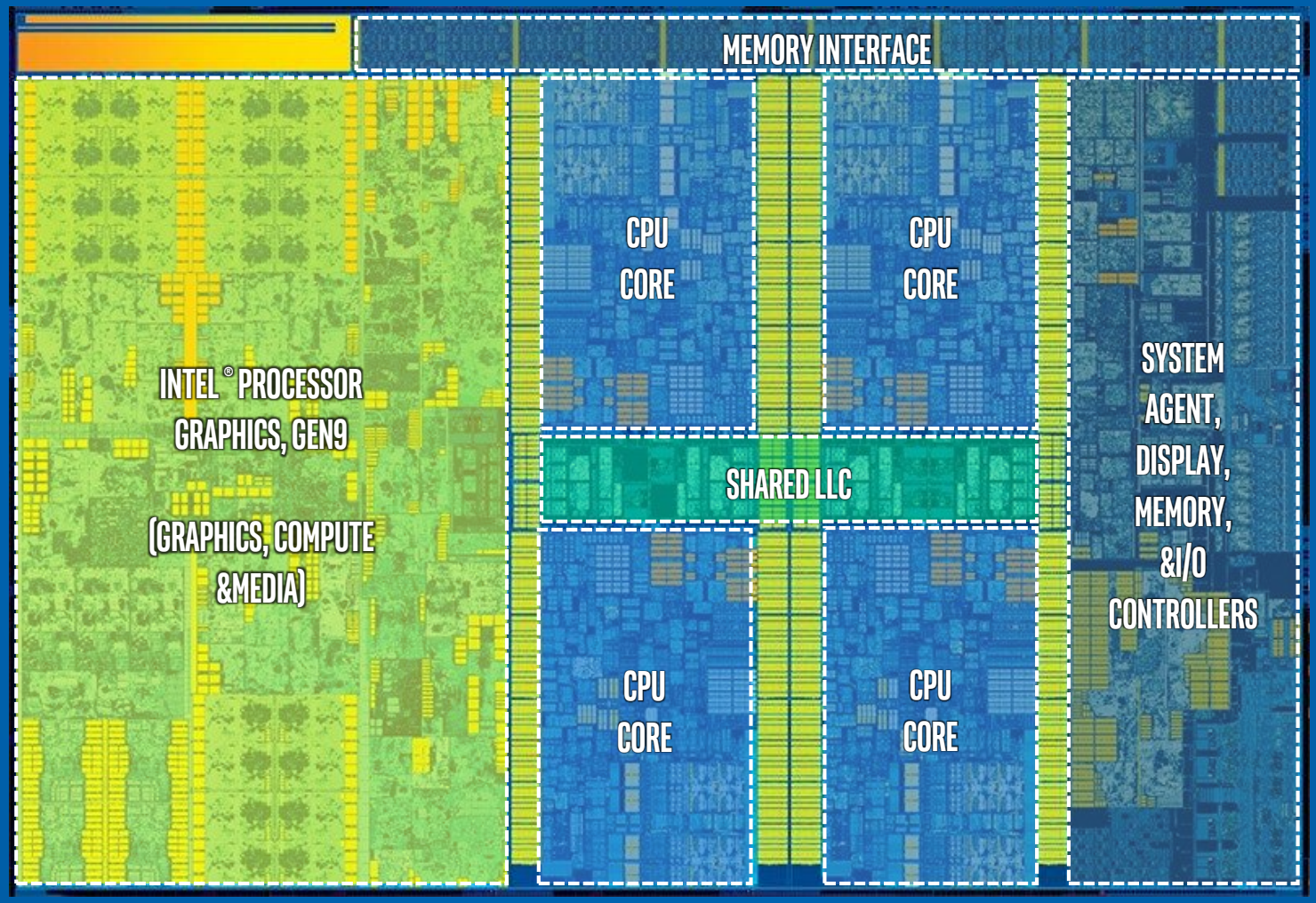
COFFEELAKE

WHAT IS SOC?

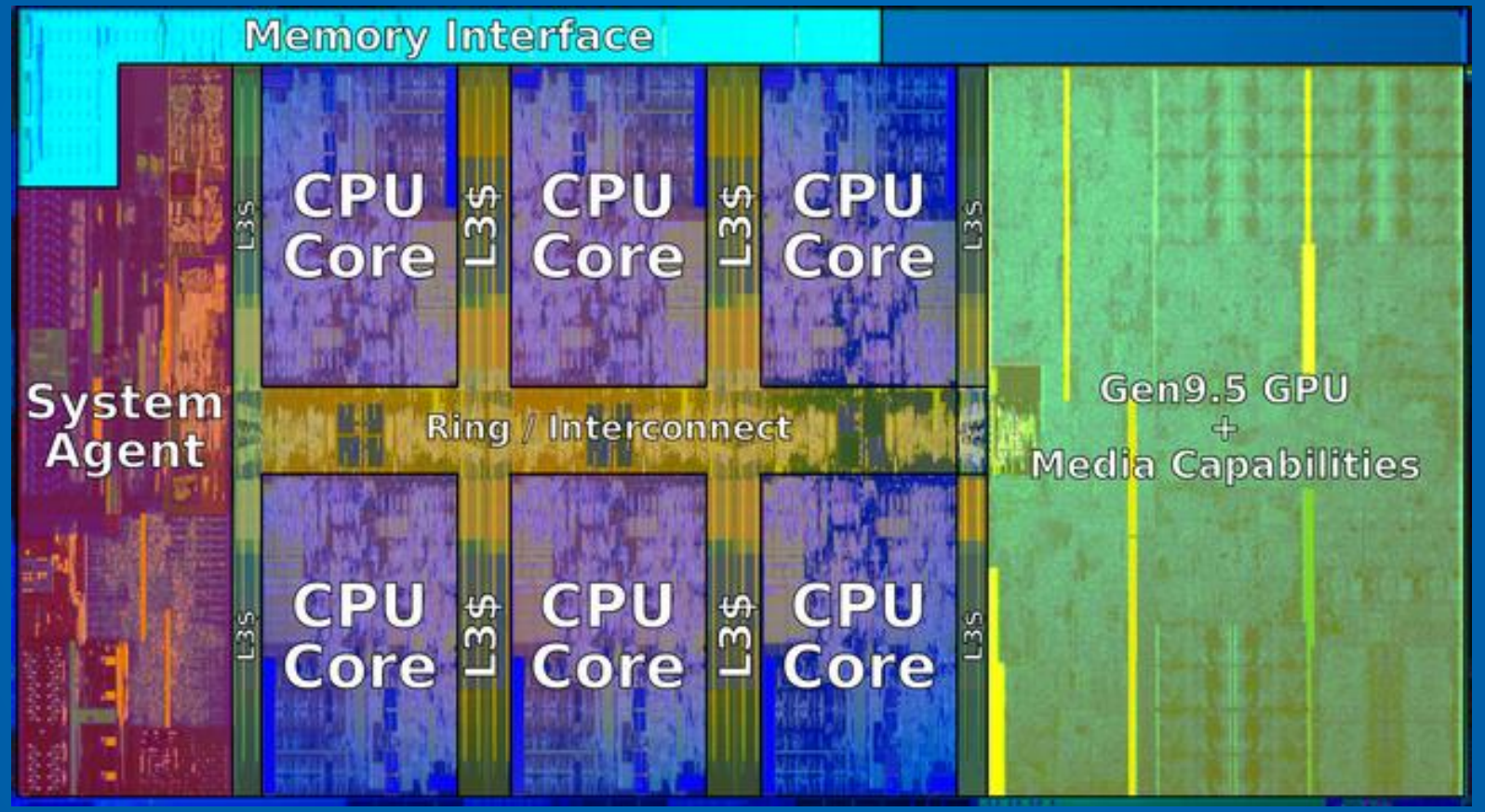
System-on-a-Chip (SoC) integrates a complete set of system components into a single chip:

- Usually contains CPU cores
- With integrated graphics, also contains GPU
- Additional components (busses, controllers, etc.)
- Caches, EDRAM, LLC

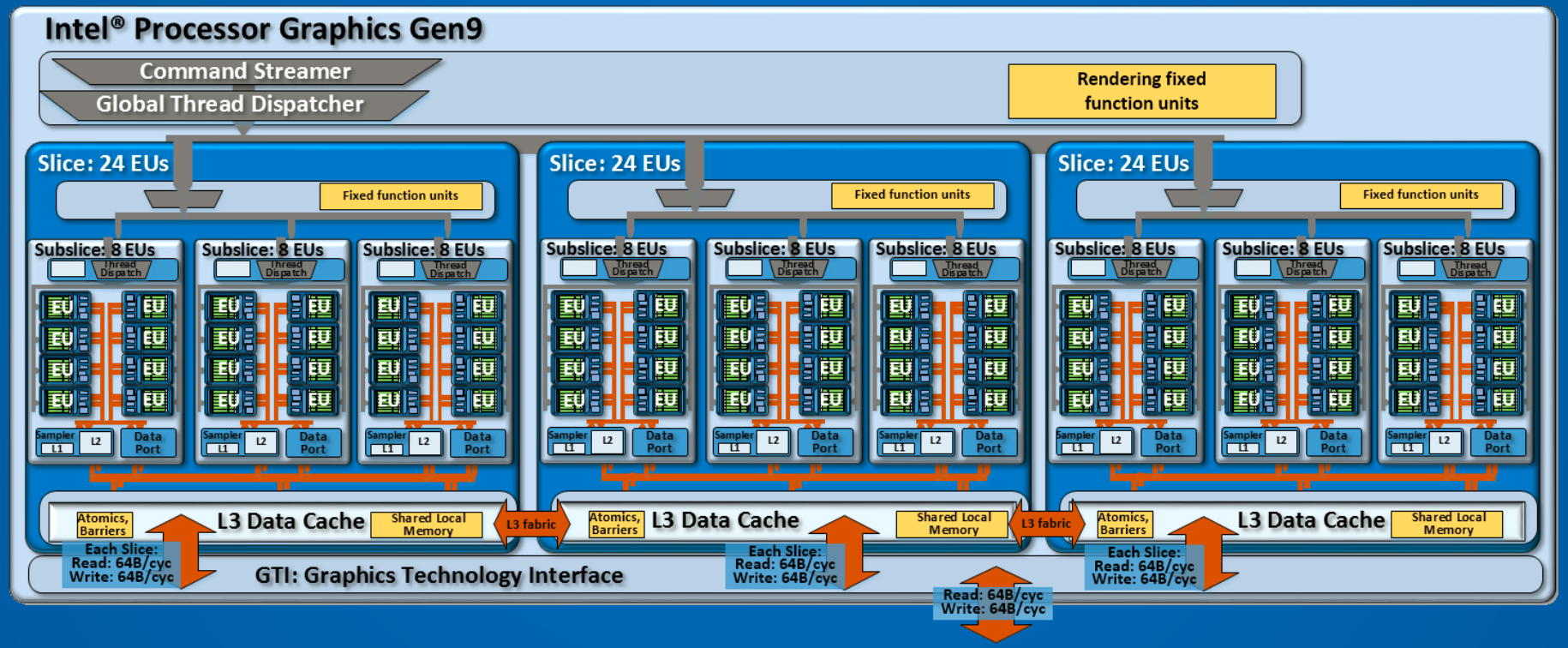
WHAT IS SOC?


MEMORY INTERFACE
CPU CORE
CPU CORE
INTEL © PROCESSOR GRAPHICS, GEN9
(GRAPHICS, COMPUTE &MEDIA)
SHARED LLC
SYSTEM AGENT, DISPLAY, MEMORY, &I/O CONTROLLERS
CPU CORE
CPU CORE

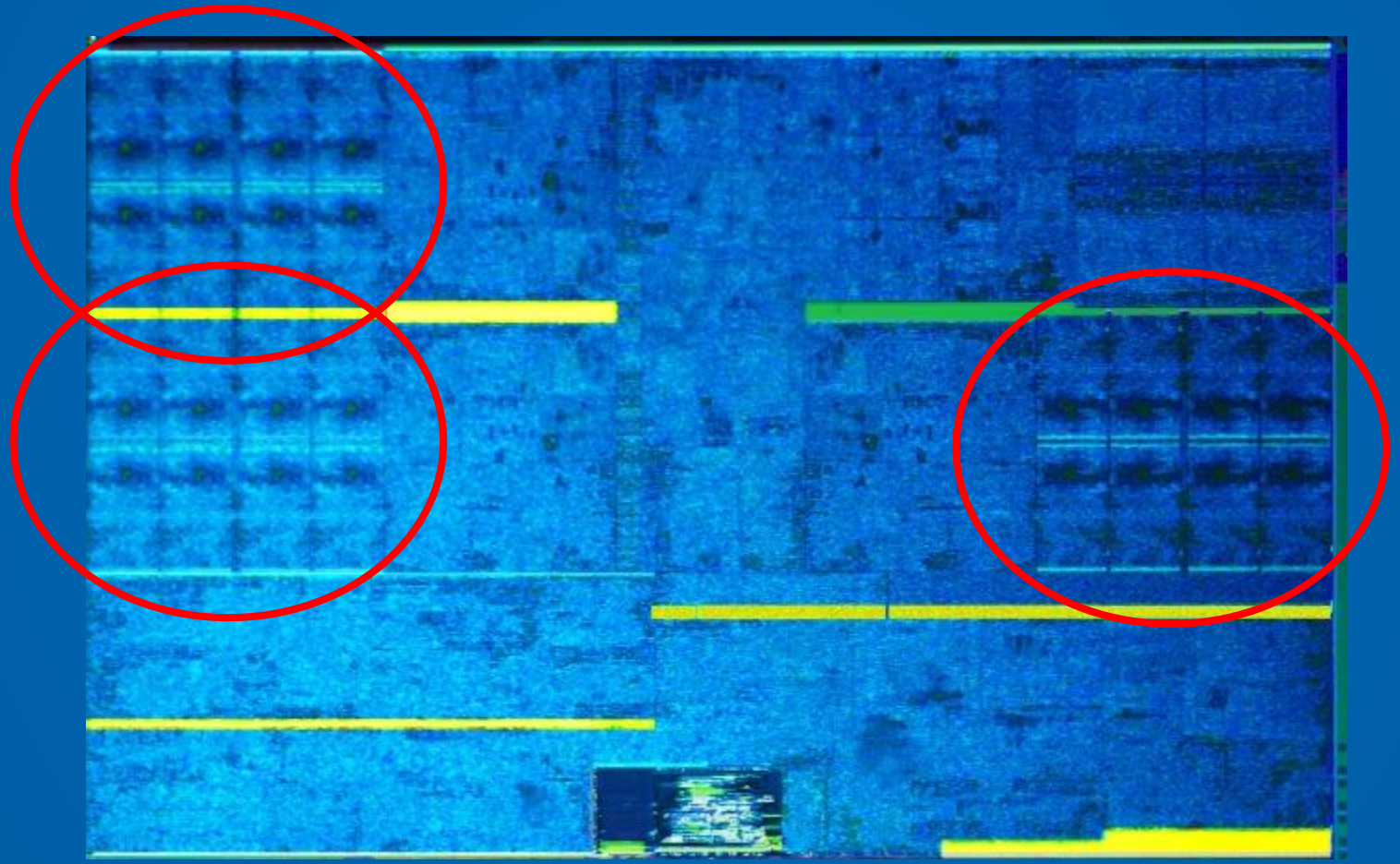
WHAT IS SOC?


Memory Interface
CPU Core
CPU Core
CPU Core
CPU Core
CPU Core
CPU Core
L3$
L3$
L3$
L3$
L3$
L3$
System Agent
Ring / Interconnect
Gen9.5 GPU
+
Media Capabilities

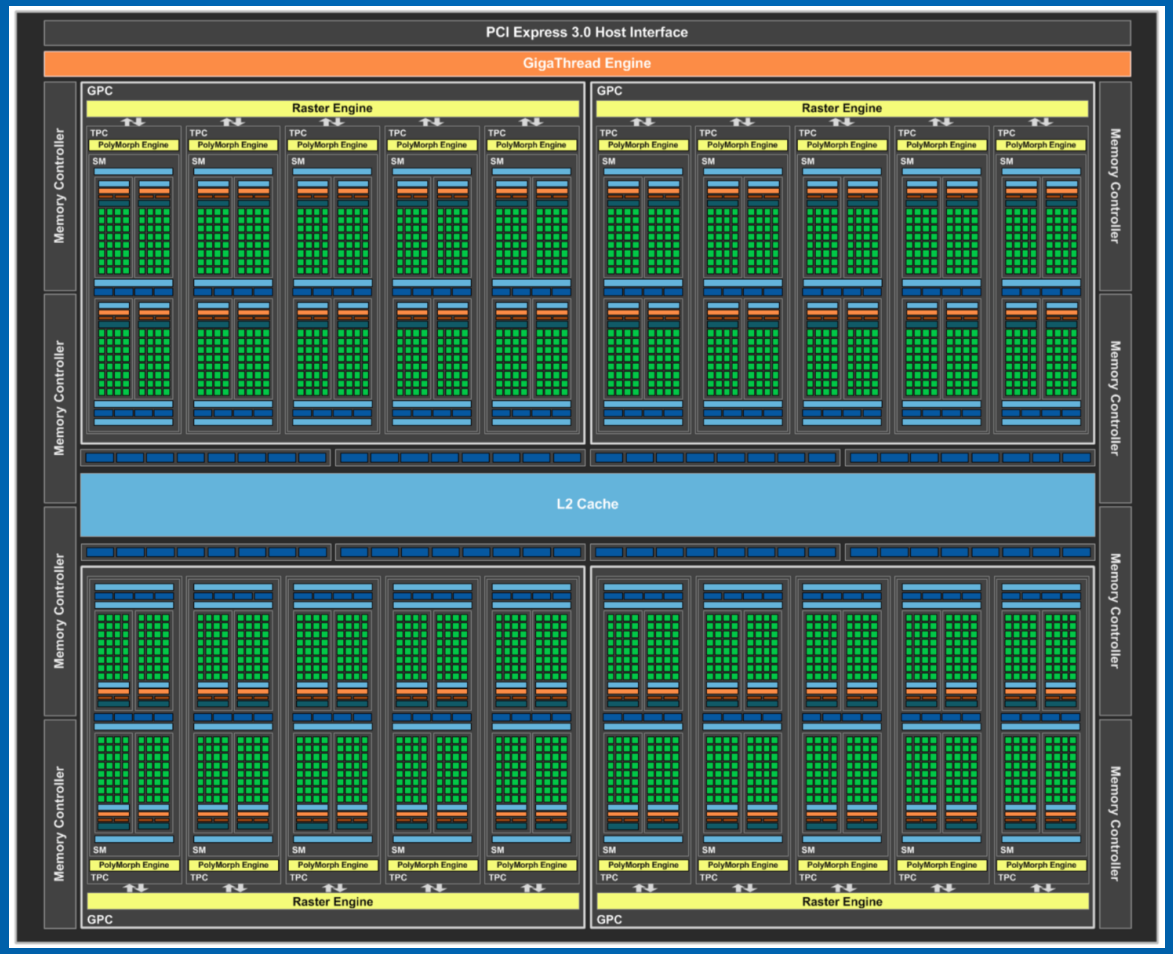
INTEL GPU ARCHITECTURE



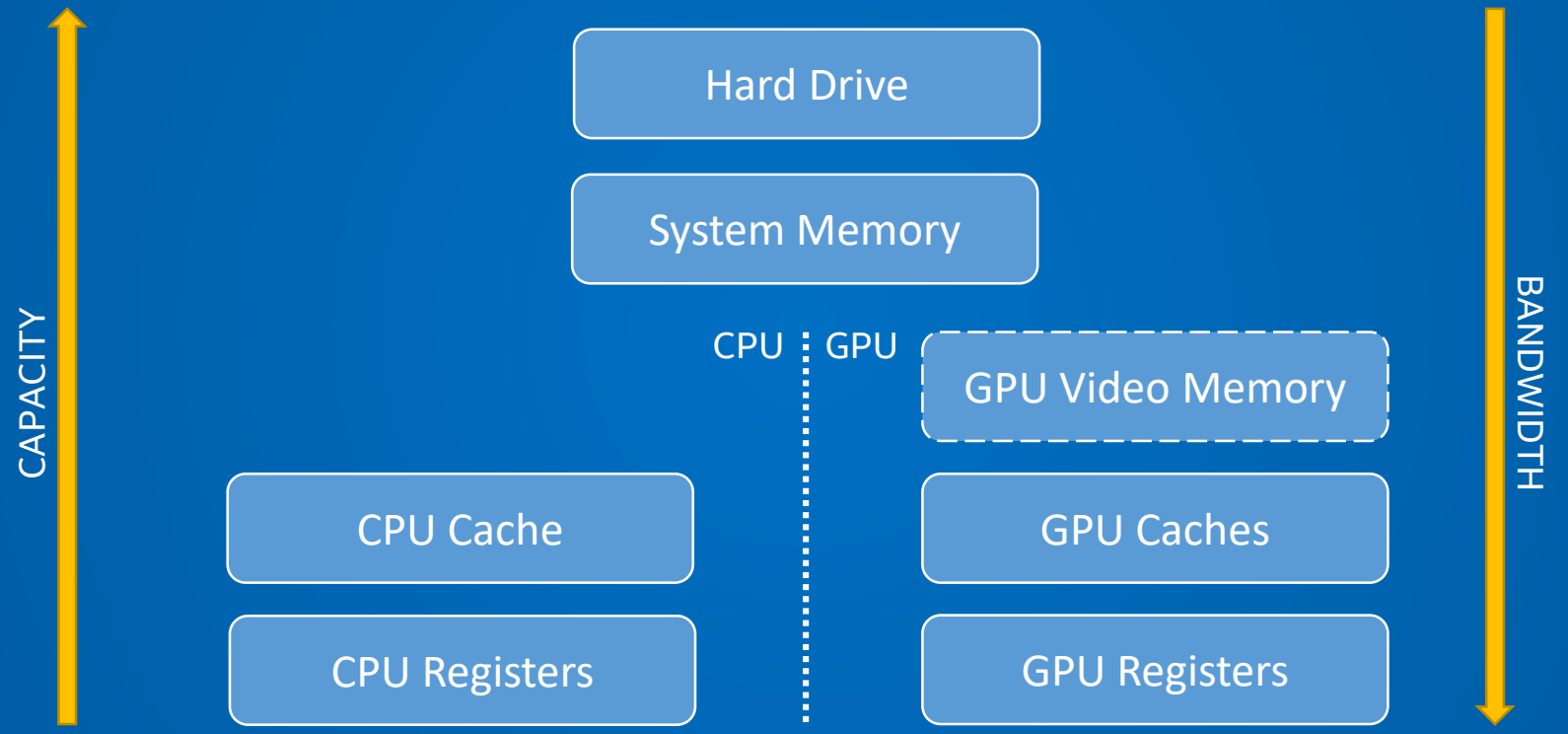
INTEL GPU ARCHITECTURE



NVIDIA GPU ARCHITECTURE

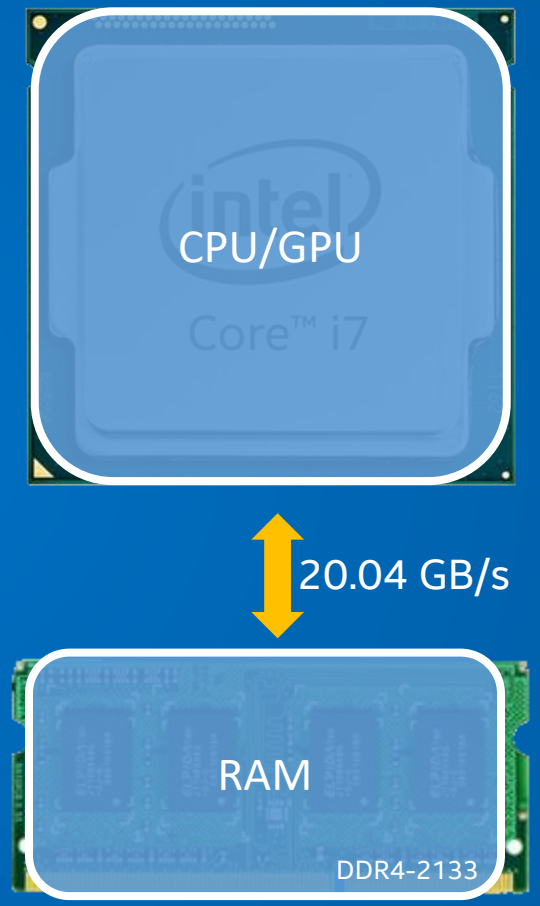


MEMORY MODEL

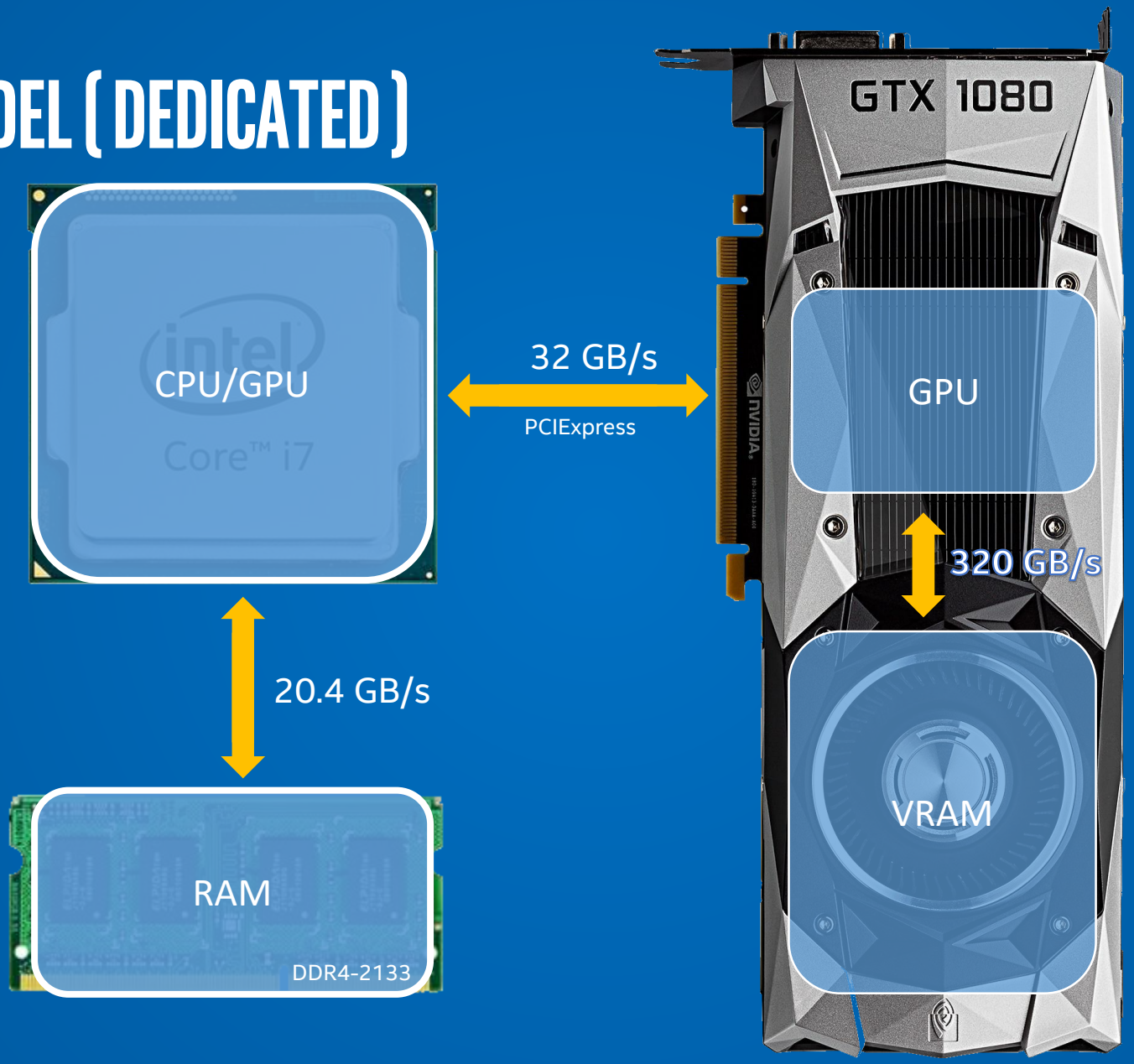


MEMORY MODEL (INTEGRATED)

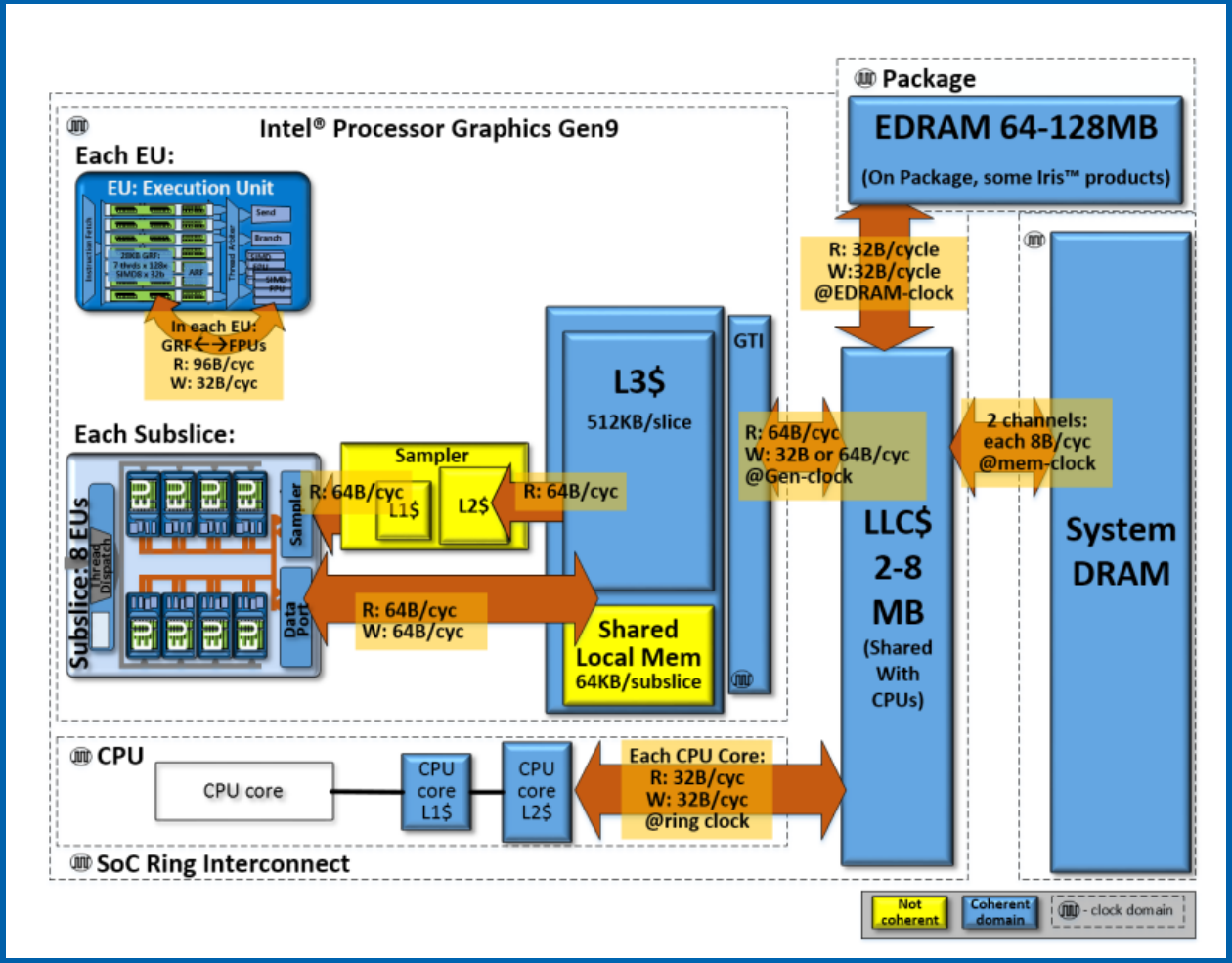
- Integrated graphics
- Unified Memory Architecture
- Zero copy data between CPU and GPU



MEMORY MODEL (DEDICATED)



MEMORY MODEL



TDP

- Thermal Power Design
- Represents the average power, in watts, the SOC dissipates when operating at base frequency with all cores active
- On integrated graphics, the power is shared between CPU, GPU, LLC, EDRAM, and every other component

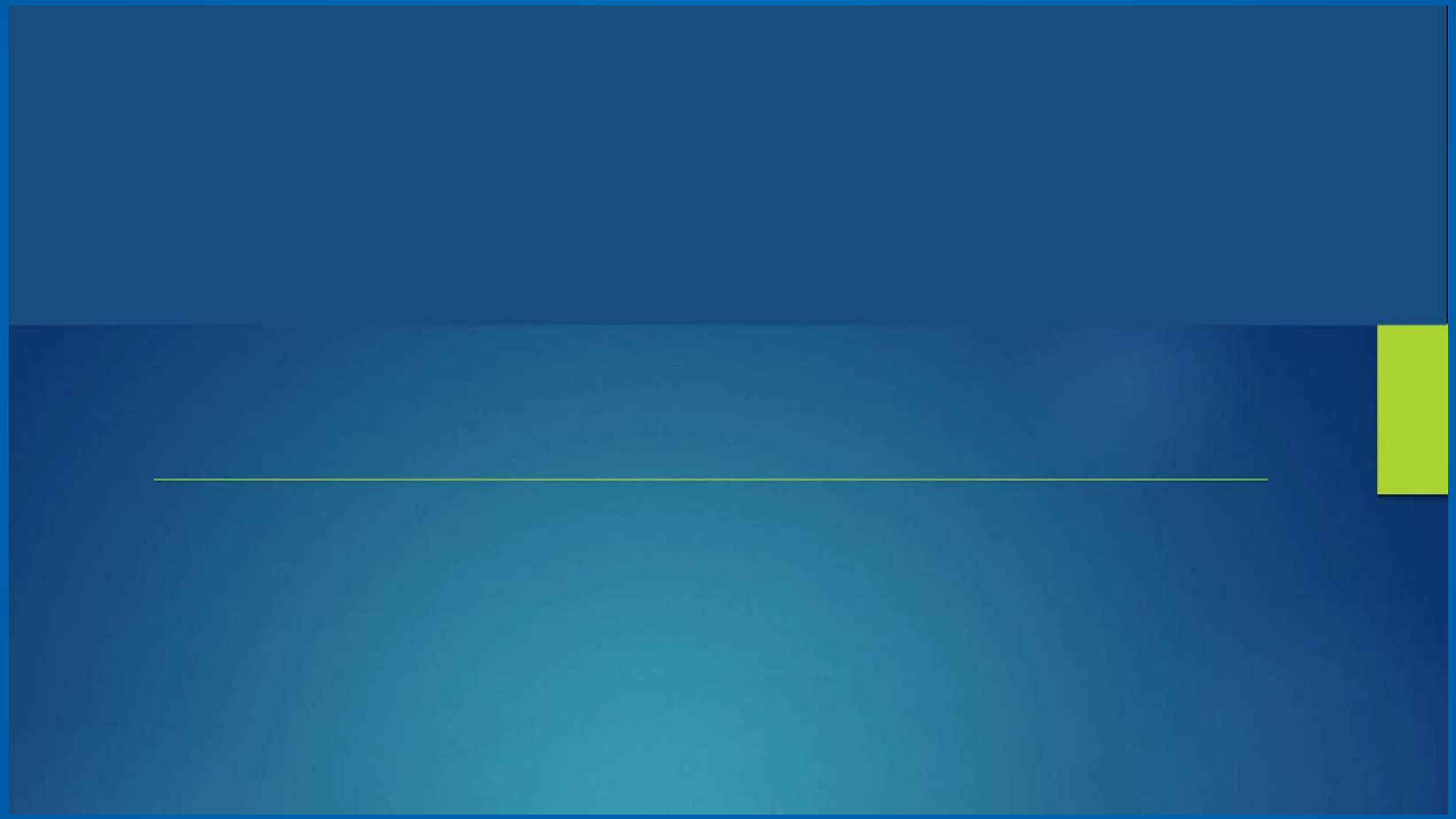
GFLOPS

- FLOPS, Floating Point Operations Per Second
- Used for measuring performance
- Calculation is complicated
- Based on frequency and GPU architecture

GFLOPS

PLATFORM	GPU	GFLOPS
	AMD Radeon GCN	1311
	Nvidia G70	240
	AMD Radeon GCN	1840
	Skylake GT4e	1152
	GP104	8228

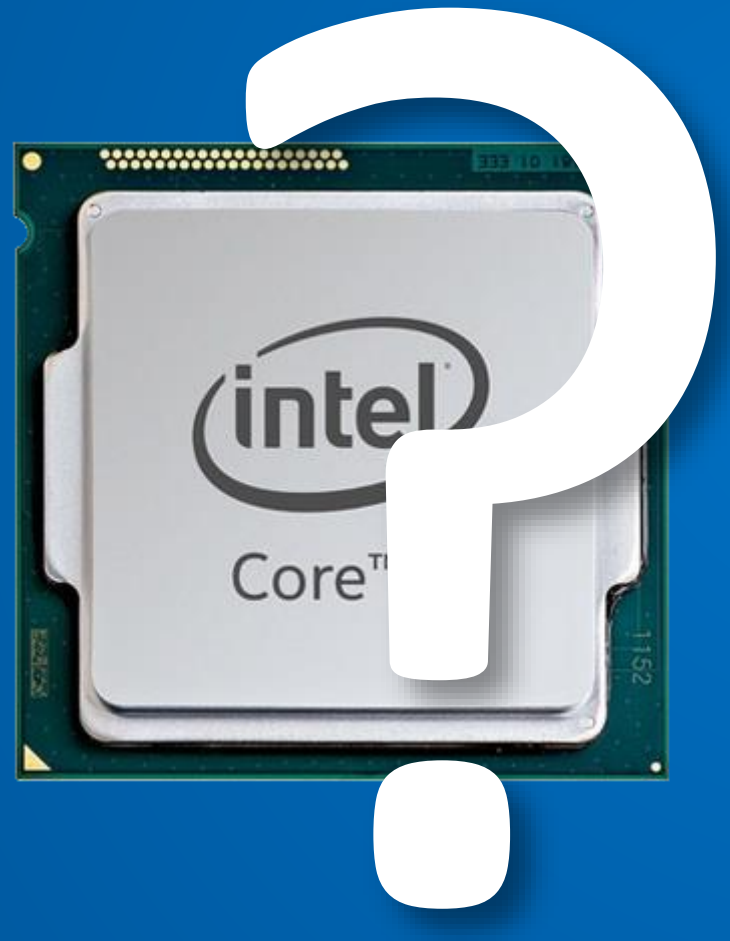
SUMMARY



SUMMARY

- Three resources: CPU, GPU, Memory
- Always have that in mind
- Many different configuration of those resources
- Each resource have it’s own advantages and disadvantages

DRIVER

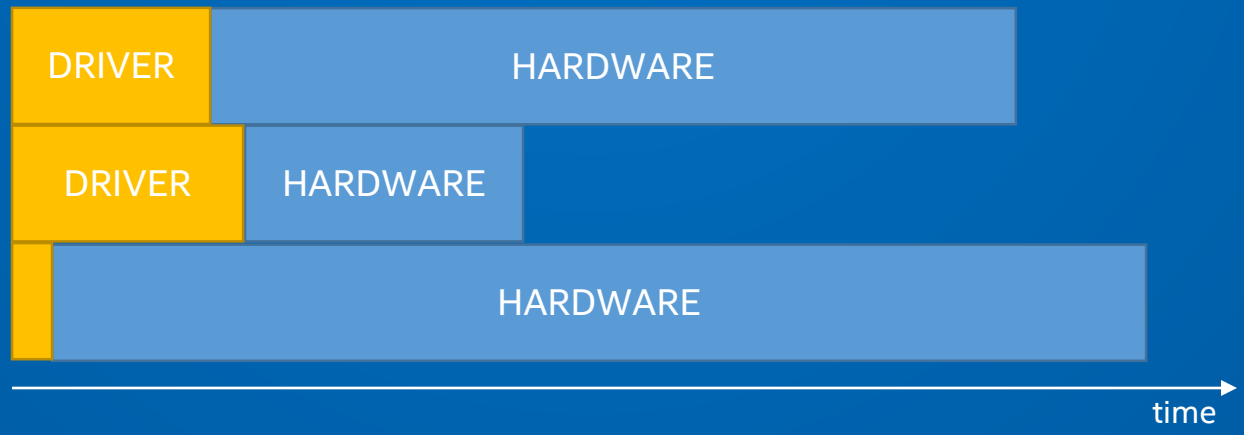


- Don’t know hardware details
- Don’t really want to know hardware details ☺
- Hardware vendors also supply drivers
- Between hardware and API
- You can always update software

DRIVER

WHAT DRIVER DOES?

- Not a tiny program (~600 MB!)
- Also uses and consumes HW resources – REMEMBER!
- Keeps track of the graphics resources
- Validates the data
- Controls the hardware (...but how?)
- Compiles shaders



WHAT DRIVER DOES?

- DirectX provides function *CreateTexture2D()*
- Application doesn’t really care how this will be done
- Application just want to have a pointer to a valid texture
- Driver has to allocate the memory for it
- Decides how this texture will be stored
- Allocates memory for mip-maps
- Copy the data from system memory (or not)

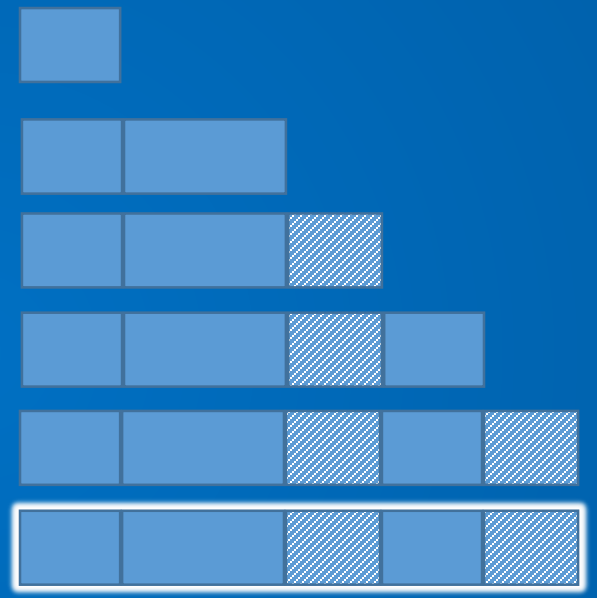
- DirectX provides function *ClearRenderTargetView()*
- Driver may spawn own shaders
- A lot to do, so also consumes resources

HOW DRIVER CONTROLS GPU?

- Driver interprets API calls and prepares a work for GPU in a form of command buffers
- Command buffers are then delivered to the GPU and enqueued in command buffer queue
- GPU is constantly reading command buffers from the queue and executes them
- Sometimes there is no work to do on the GPU, sometimes there is too much work

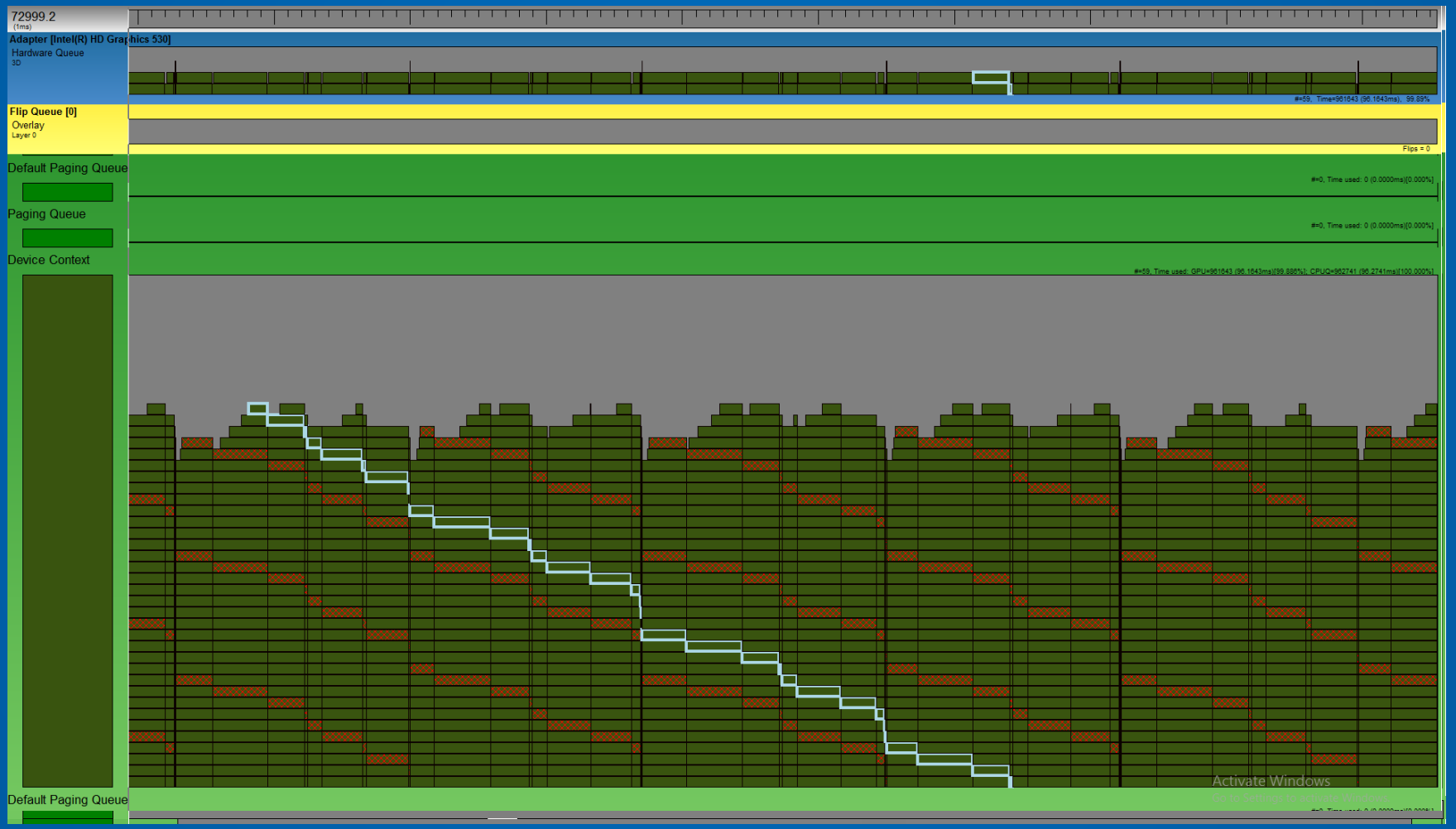
HOW DRIVER CONTROLS GPU?

- PSSetShader()
- IASetVertexBuffer()
- Draw()
- PSSetShader()
- Draw()
- Present()



(command buffer)

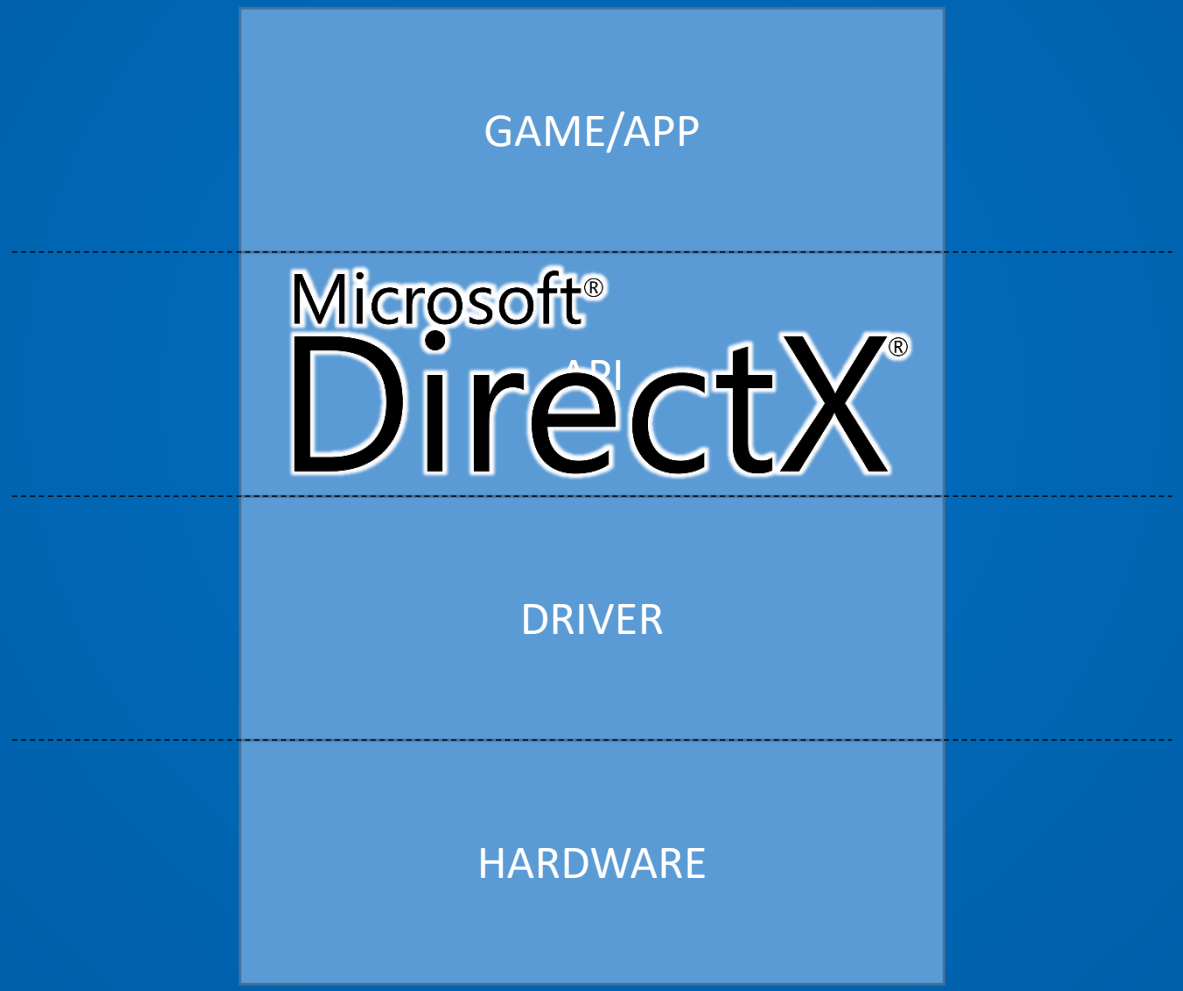
HOW DRIVER CONTROLS GPU?



DIRECTX

- Set of application programming interfaces
- Developed by Microsoft, only for Windows platforms
- First version introduced in 1995
- Latest version is DirectX 12, introduced in 2014

DIRECTX

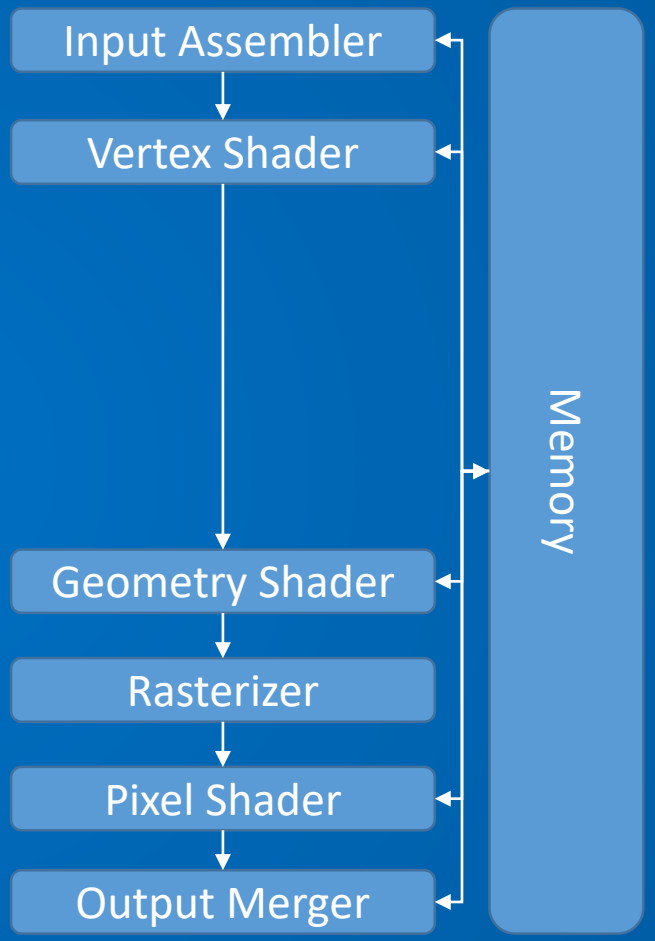


DRAWCALL

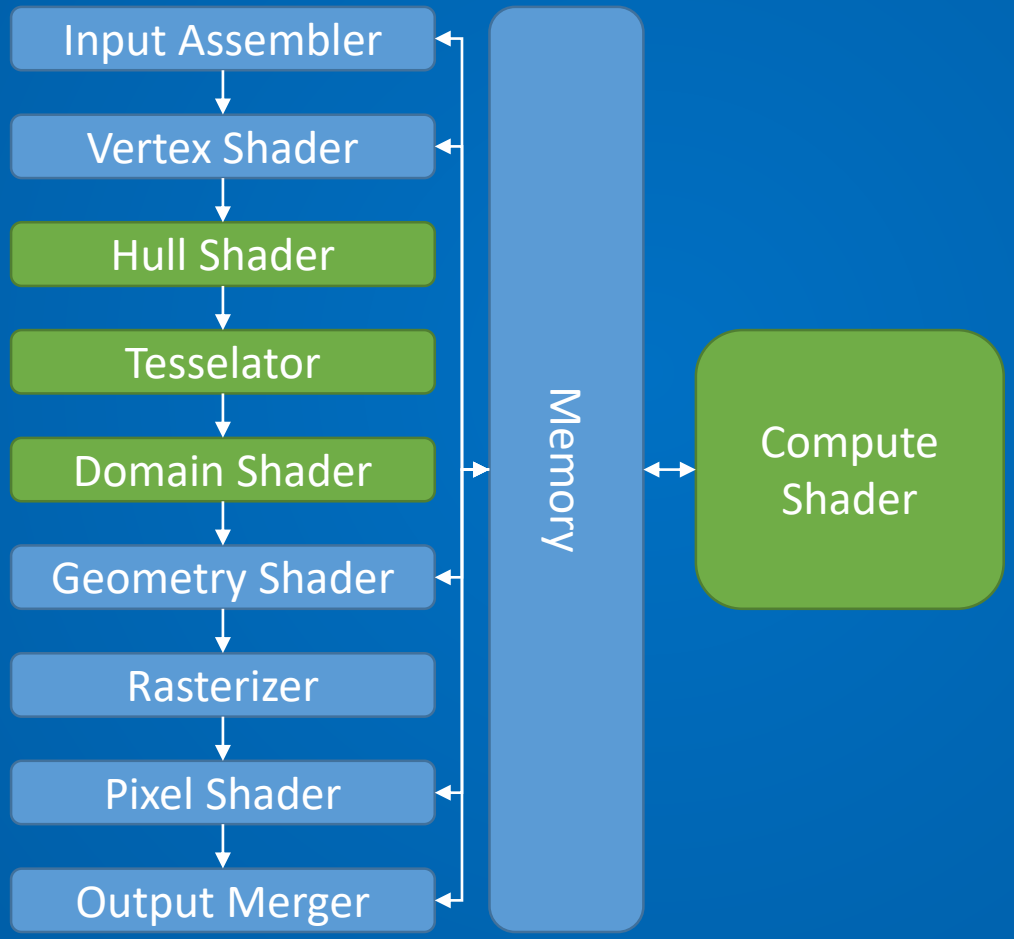
- Any API call that triggers pipeline to do some work
- Typically Draw(), DrawInstanced(), DrawIndexed()
- Also Clears and Copies

DIRECTX 10 PIPELINE

- Represents a fundamental architecture change



DIRECTX 11 PIPELINE



DIRECTX DEVICE

- Represented by ID3D11Device
- Used to create resources
  - CreateBuffer
  - CreateTexture2D
  - CreateRenderTargetView
  - CreateDepthStencilView
  - CreatePixelShader
  - CreateVertexShader

DIRECTX DEVICE CONTEXT

- Also called Device Context
- Used to set pipeline states
- Generates rendering commands
- Using resources owned by a Device
- Examples:
  - Draw
  - IASetVertexBuffer/IASetIndexBuffer
  - PSSetShader
  - OMSetRenderTarget

DIRECTX SWAP CHAIN

- SwapChain is a collection of buffers that are used for displaying frames to the user
- Each time application presents new frame, the first buffer (Front Buffer) in the Swap Chain takes the place of the displayed buffer
- We render to the last buffer (Back Buffer)
- This process is called swapping or flipping

BackBuffer FrontBuffer

DIRECTX RESOURCES

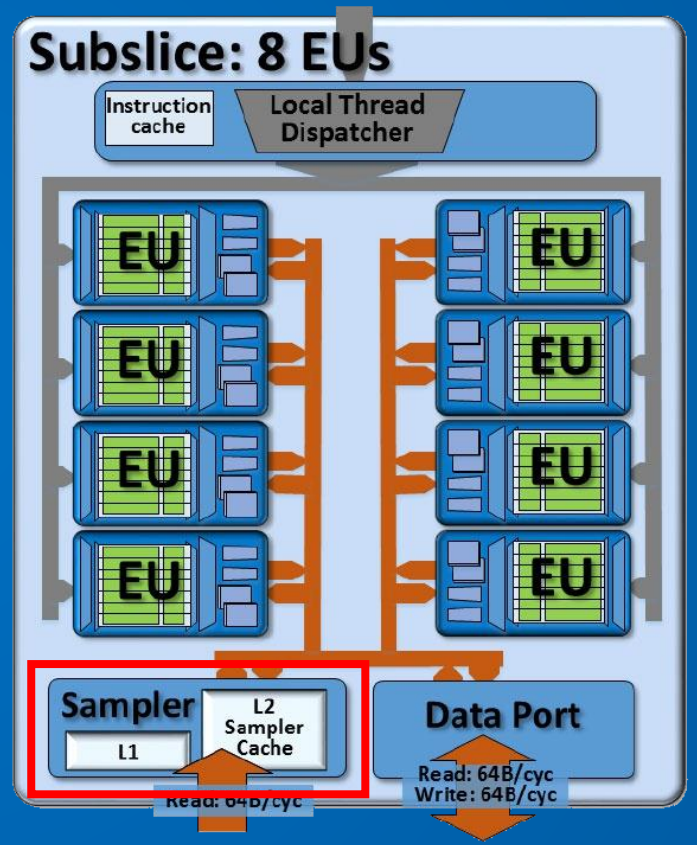
- Building blocks of the scene
- Areas in memory accessible by DirectX pipeline
- Buffers or Textures
- Read or Write access
- Accessible by CPU or GPU
- Create, Bind, Release

DIRECTX BUFFERS

- Collection of elements
- Vertex Buffer, Index Buffer, Constant Buffer
- Unstructured resource
- Cannot contain any mipmap levels
- Cannot get filtered when read
- Cannot be multisampled

DIRECTX TEXTURES

- Stores texels
- Can be filtered by samplers
- Can be read by shader units
- Hardware support - SAMPLERS
- Each texel can have 1 – 4 components



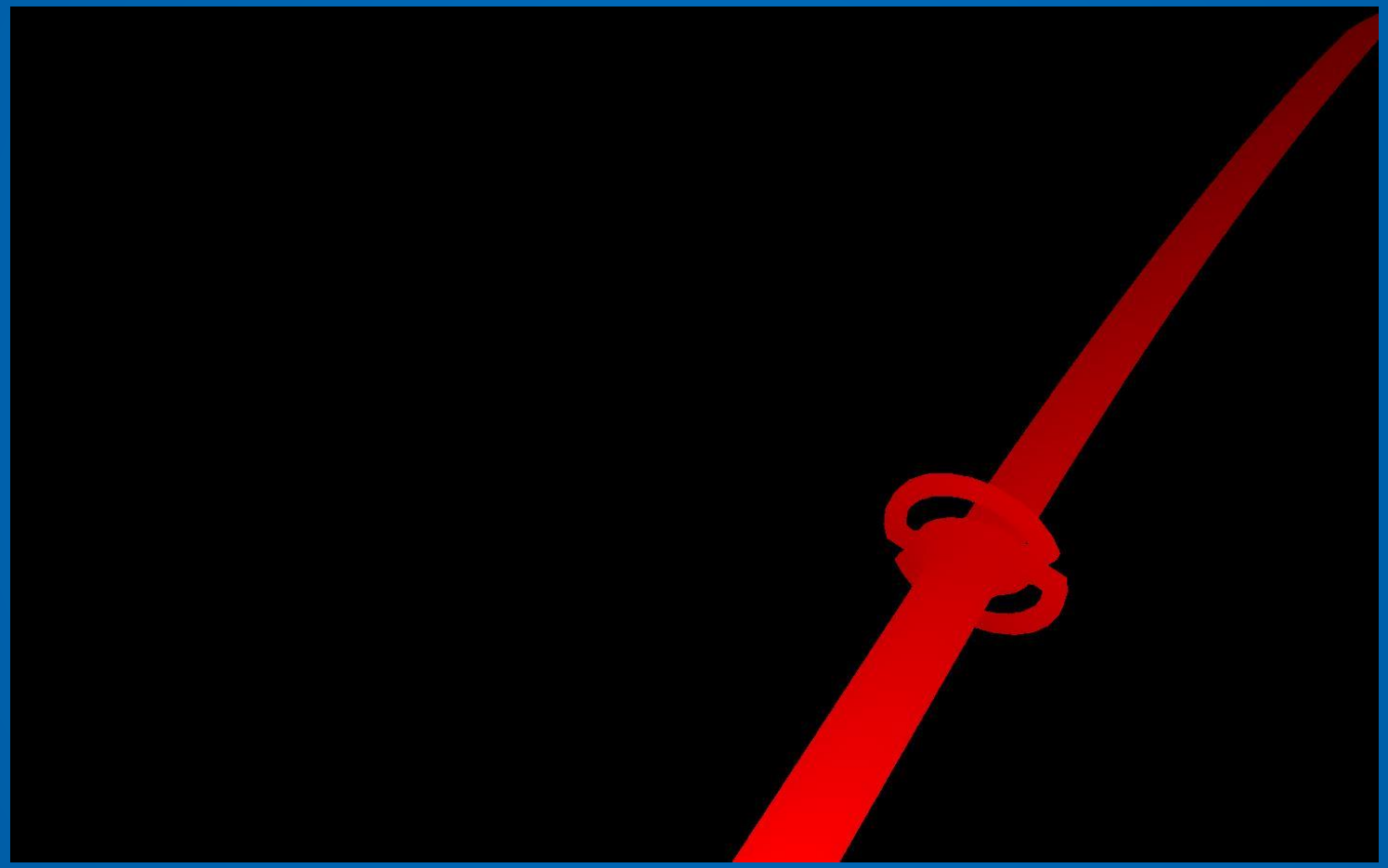
DIRECTX RENDER TARGET

- Usually a texture, where GPU draws pixels for a scene that is being rendered
- In DX11 it is Render Target View – a cast of a resource
- Render Target doesn't necesserily need to be a back buffer

DIRECTX DEPTH BUFFER

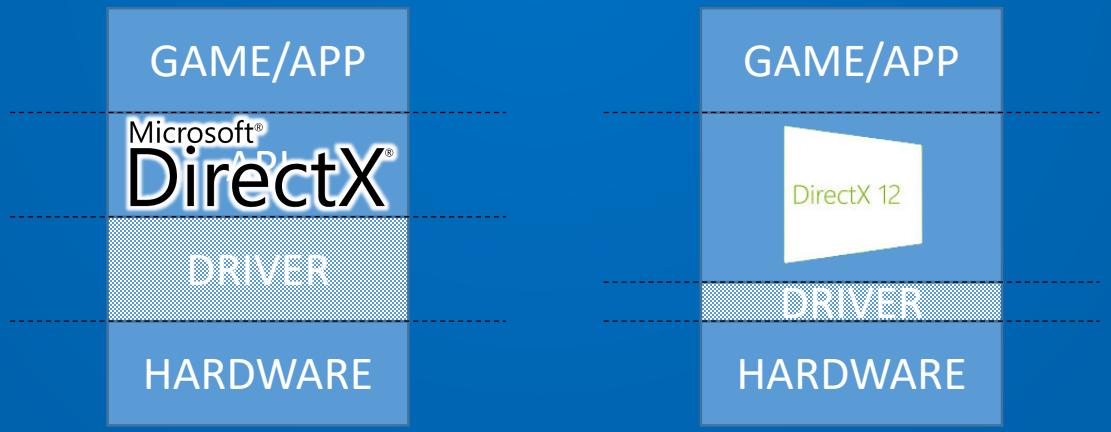
- Contains per-pixel data for z-depth of each pixel rendered
- Strongly related to Render Target
- Each render target has to have Depth Buffer
- Depth Buffer can be independent

DIRECTX DEPTH BUFFER



DIRECTX 12 / VULKAN

- So called low-level API
- Reduced CPU overhead
- Application has much more work
- Driver is very thin



THANK YOU!

bartosz.boczula@intel.com