Battlefield 6: Pushing visual fidelity while optimizing for all hardware

Notices & Disclaimers

- The preceding presentation contains product features that are currently under development. Information shown through the presentation is based on current expectations and subject to change without notice.
- Results that are based on pre-production systems and components as well as results that have been estimated or simulated using an Intel Reference Platform (an internal example new system), internal Intel analysis or architecture simulation or modeling are provided to you for informational purposes only. Results may vary based on future changes to any systems, components, specifications or configurations.
- Performance varies by use, configuration and other factors. Learn more at www.intel.com/PerformanceIndex.
- AI features may require software purchase, subscription or enablement by a software or platform provider, or may have specific configuration or compatibility requirements. Details at www.intel.com/AIPC.
- No product or component can be absolutely secure. Intel technologies may require enabled hardware, software or service activation.
- All product plans and roadmaps are subject to change without notice.
- Some images may have been altered or simulated and are for illustrative purposes only.
- Codec capabilities may vary by device and configuration. Contact your manufacturer to understand the enabled hardware acceleration and codec capabilities for individual device.
- Built into the hardware, Intel® Thread Director is provided only in performance hybrid architecture configurations of 12th Gen or newer Intel® Core processors; OS enablement is required. Available features and functionality vary by OS. Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.
- Statements in this document that refer to future plans or expectations are forward-looking statements. These statements are based on current expectations and involve many risks and uncertainties that could cause actual results to differ materially from those expressed or implied in such statements. For more information on the factors that could cause actual results to differ materially, see our most recent earnings release and SEC filings at www.intc.com.
- SPEC®,SPECrate® and SPEC CPU® are registered trademarks of the Standard Performance Evaluation Corporation. See http://www.spec.org/spec/trademarks.html for more information.
-

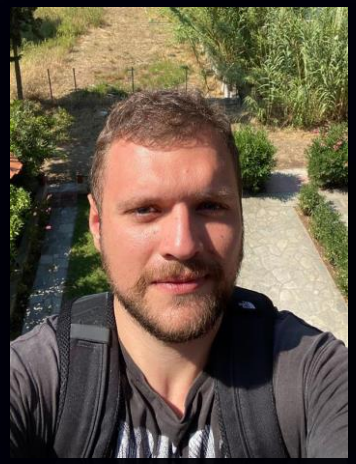

16
www.pegi.info
In-game Purchases
GAMEPLAY NOT FINAL. REPRESENTS A GAME IN DEVELOPMENT.
CONTAINS A MIXTURE OF CINEMATIC AND GAMEPLAY FOOTAGE.

Introduction

Agenda

- Performance Targets
- Testing
- The Pipeline
- Walking Through a Frame
- GPU Culling
- Why Battlefield 6 Performs well on Intel Built-In GPUs
- Taking Performance Further with Intel XeSS 2
- Conclusion

Lyubomir Kozlovski,
Rendering Engineer,
EA DICE



Christos Loukovikas,
Rendering Engineer,
Criterion Games

Leigh Davies,
Principal Engineer,
Intel

Performance Targets

Performance Targets

- No Upscaling or FG for base targets
- Min Spec 1080p30 Low Settings - Lunar Lake (140V)
- Rec Spec 1440p60 High Settings - Battlemage (B580)
- Treat each hardware tier as an individual platform

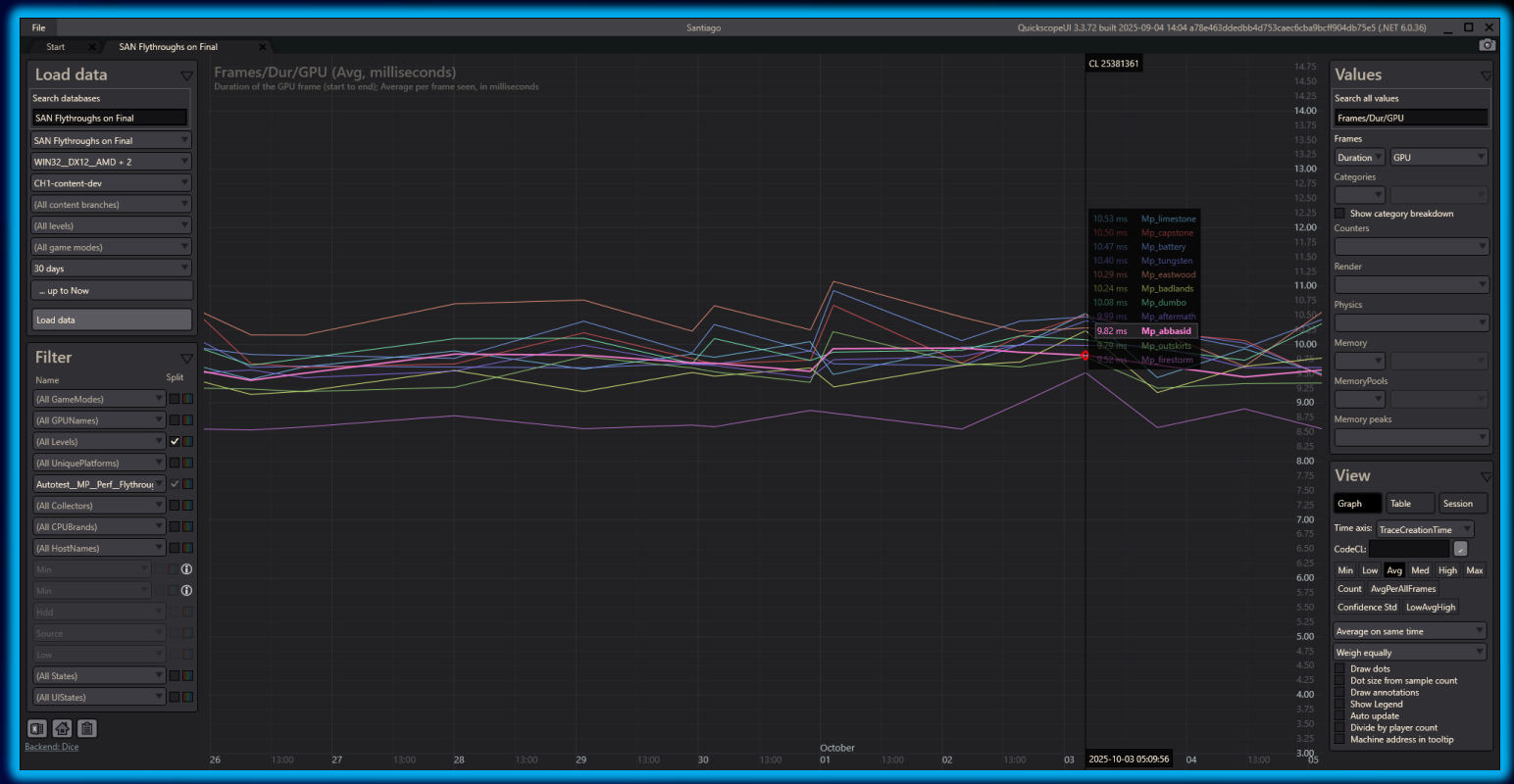
	MINIMUM	RECOMMENDED		ULTRA	
		BALANCED	PERFORMANCE	BALANCED	PERFORMANCE
GRAPHIC SETTINGS	1080p @ 30 FPS Low Settings	1440p @ 60 FPS High Settings	1080p @ 80 FPS+ Low Settings	2160p(4K) @ 60 FPS Ultra Settings	1440p @ 144 FPS High Settings
UPSCALER	Native	Native		Native	
GPU	NVIDIA GeForce RTX 2060 AMD Radeon RX 5600 XT Intel Arc A380	NVIDIA GeForce RTX 3060 Ti AMD Radeon RX 6700 XT Intel Arc B580		NVIDIA GeForce RTX 4080 AMD Radeon RX 7900 XTX	
VIDEO MEMORY	6GB	8GB		16GB	
CPU	Intel Core i5-8400 AMD Ryzen 5 2600	Intel Core i7-10700 AMD Ryzen 7 3700X		Intel Core i9-12900k AMD Ryzen 7 7800X3D	
RAM	16GB (Dual channel 2133mhz)	16GB (Dual Channel 3200mhz)		32GB (Dual Channel 4800mhz)	
OS	Windows 10**	Windows 11 64-bit		Windows 11 64-bit	
DIRECTX	DirectX12	DirectX12		DirectX12	
STORAGE	55GB* HDD (At Launch)	90GB* SSD (At Launch)		90GB* SSD (At Launch)	
TPM 2.0 ENABLED UEFI SECURE BOOT ENABLED HVCI CAPABLE VBS CAPABLE	Required	Required		Required	

* ESTIMATED DISK SPACE. **ENABLING HARDWARE ACCELERATED GPU SCHEDULING IS RECOMMENDED

Testing

Testing - Quickscope

- ~12K tests a day
- Integrated in our build process
- Track performance over time
- Multiple hardware configurations
- Easy to spot unexpected regressions
- Dedicated performance team to make sure we stay on track
- Collaborative effort from all crafts



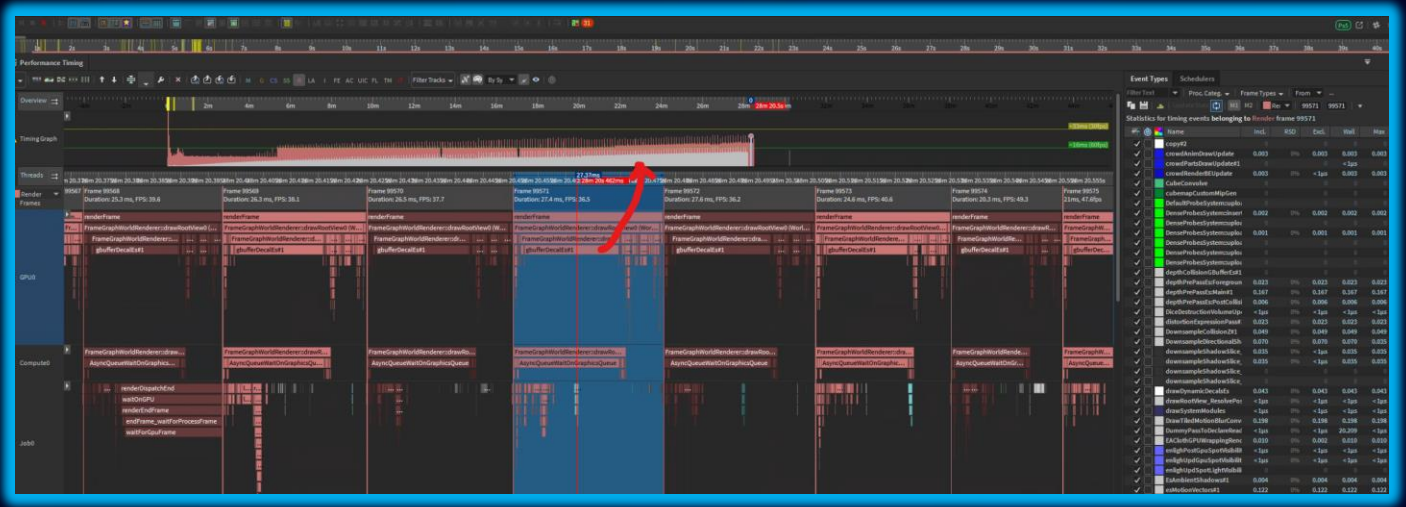
Testing - Quickscope

Top GPU changes

FrameGraphWorldRenderer::drawRoot...	8.93	->	7.99	=	-0.94 ms	High: -0.87 ms
renderFrame	9.52	->	8.58	=	-0.94 ms	High: -0.88 ms
GraphicsQueueWaitOnAsyncQueue	0.82	->	0.35	=	-0.47 ms	High: -0.35 ms
RenderPassBookmark_FirstViewUpdate	0.81	->	1.10	=	0.29 ms	High: 0.53 ms
shadowAccum/shade	0.19	->	0.47	=	0.28 ms	High: 0.41 ms
FrameGraphWorldRenderer::drawRoot...	1.21	->	1.02	=	-0.19 ms	High: -0.30 ms
GenerateCascadedShadowMaps	0.21	->	0.07	=	-0.14 ms	High: -0.24 ms
AccumDirectionalShadow	0.14	->	0.28	=	0.14 ms	High: 0.20 ms
TerrainTextureGenerateParallel	0.41	->	0.54	=	0.13 ms	High: 0.34 ms
TextureCompositorLayerGraphs	0.34	->	0.47	=	0.13 ms	High: 0.33 ms
GpuMeshScattering::update	0.15	->	0.27	=	0.12 ms	High: 0.12 ms
GpuMeshScattering::cullView	0.13	->	0.25	=	0.11 ms	High: 0.14 ms
RenderPassBookmark_PreDepthPrepass	0.13	->	0.25	=	0.11 ms	High: 0.14 ms
VisualTerrainRenderPassModule::on...	0.48	->	0.59	=	0.11 ms	High: 0.34 ms
mainGBufferEs	0.73	->	0.62	=	-0.11 ms	High: -0.20 ms
RenderPassBookmark_RootViewBegin	0.13	->	0.02	=	-0.11 ms	High: -0.15 ms
mainGBufferEsFg	0.10	->	0.00	=	-0.10 ms	High: -0.20 ms
GpuMeshScattering_animation	0.11	->	0.21	=	0.10 ms	High: 0.10 ms
accumShadow1	0.07	->	0.16	=	0.10 ms	High: 0.18 ms
slice_1	0.07	->	0.16	=	0.10 ms	High: 0.18 ms
AccumDistantDirectionalShadowCache	0.13	->	0.22	=	0.10 ms	High: 0.19 ms
LightCulling	0.10	->	0.19	=	0.09 ms	High: 0.18 ms
GpuMeshScattering_cull	0.13	->	0.21	=	0.08 ms	High: 0.12 ms
CascadedShadowMaps	0.81	->	0.89	=	0.08 ms	High: -0.00 ms
temporalAA	0.54	->	0.46	=	-0.08 ms	High: -0.07 ms

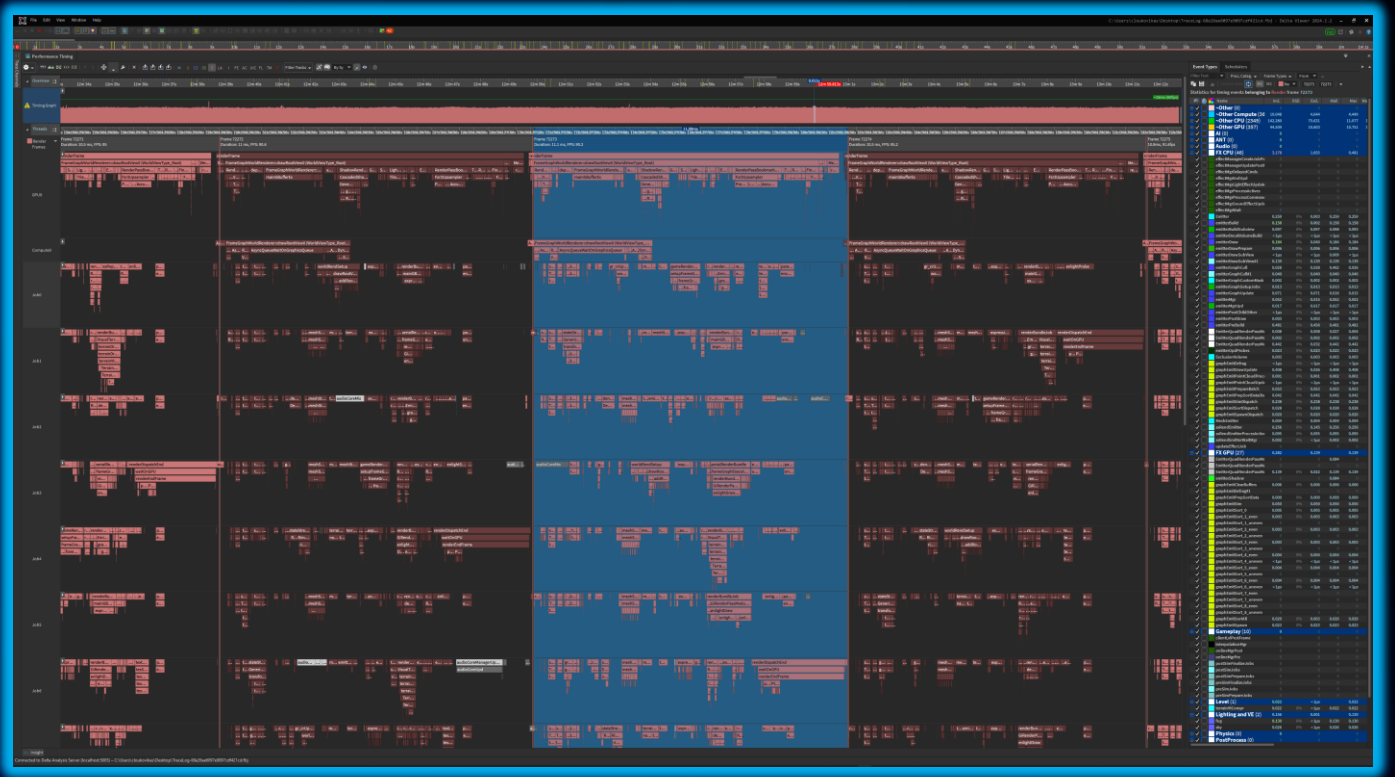
Testing - Automation

Testing - Automation



Testing - Delta Viewer

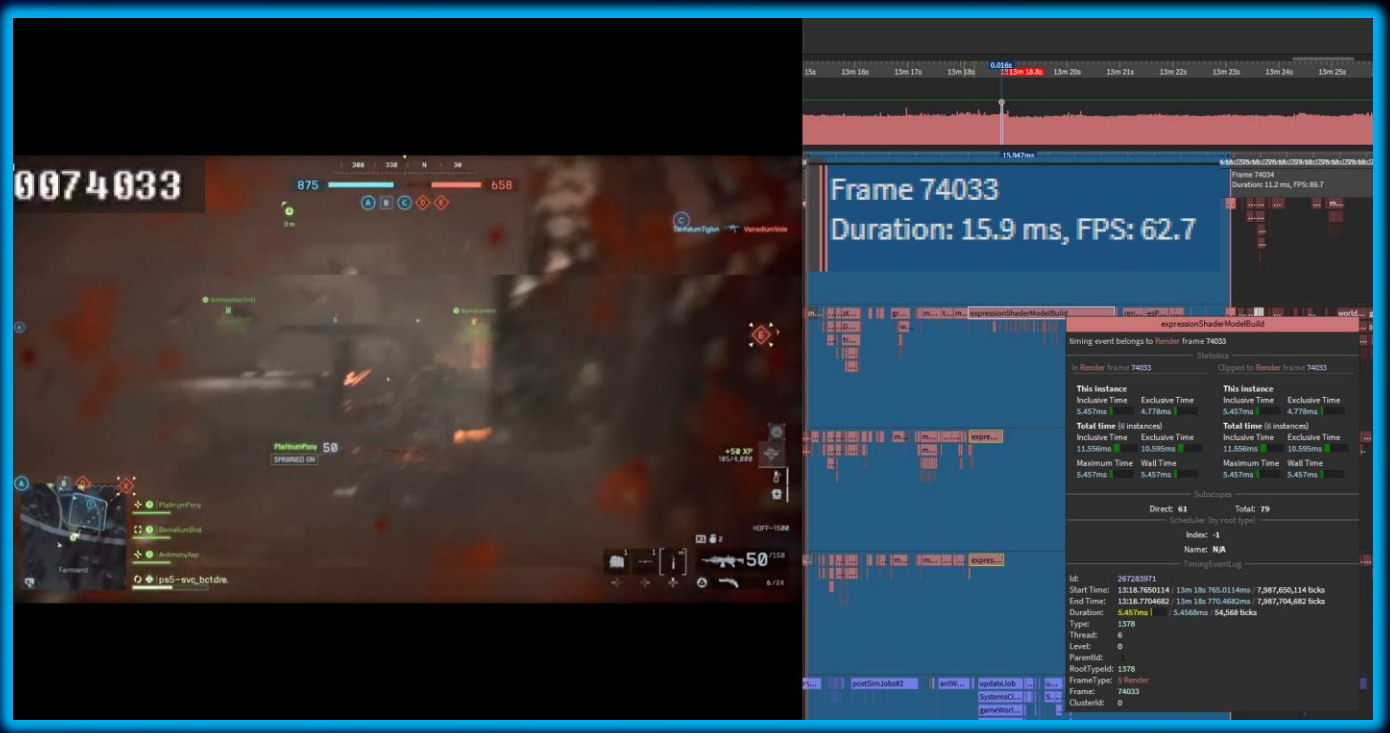
- Internal tool to spot performance bottlenecks
- Makes it easy to find bubbles in the frame
- Purpose built for Frostbite, lots of systems plug directly to it
- Easy to add markers



```
FB_PROFILE_SCOPE("GpuDecal_DrawView", math::Color32::DeuterMint);
```

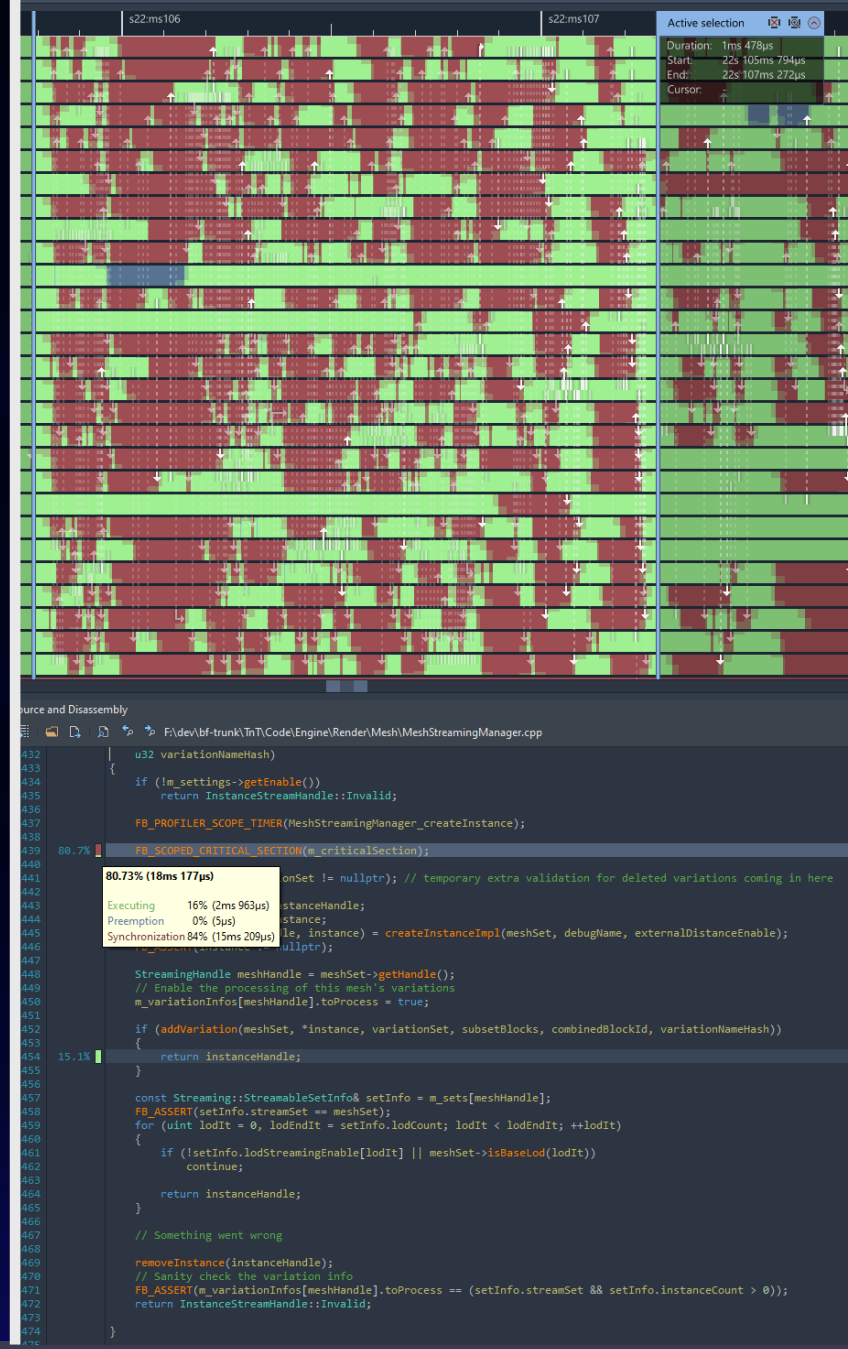
Testing - Delta Viewer

- Video captures and perf journals share frame index
- Doesn't tell us *why* something is expensive but provides more context



Testing - Profiling

- Superluminal, Razor, PIX and other tools for CPU and GPU profiling
- Multithreading profiling crucial for performance in Frostbite



Testing - Intel Partnership

- Getting player feedback early was crucial
- Shared builds with HW vendors around the same time BF Labs started
- Collaborated closely with Intel to bring XeSS 2 to BF6 and Frostbite
- Great feedback loop and iteration for performance issues, GPU hangs, XeSS integration, driver performance improvements and scaling across hardware

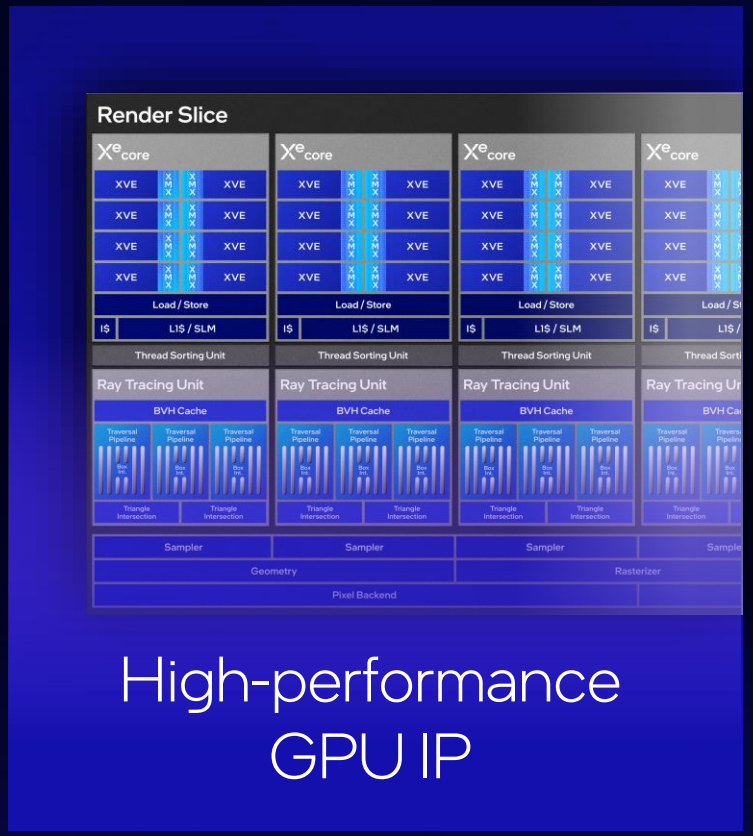
Beyond ultimate performance seekers, there are gamers who want more mobility and all-day battery life

Rank	Desired System Qualities	
1	**Longer battery life**	**29%**
2	**Lighter weight / thinner**	**27%**
3	More RAM	26%
4	Faster charging time	24%
5	More storage	22%
6	**Less hot**	**21%**
7	Better graphics	19%
8	Faster start-up time	19%
9	Faster performance	19%
10	Better design / appearance	16%
11	**Quieter fan / operation**	**15%**

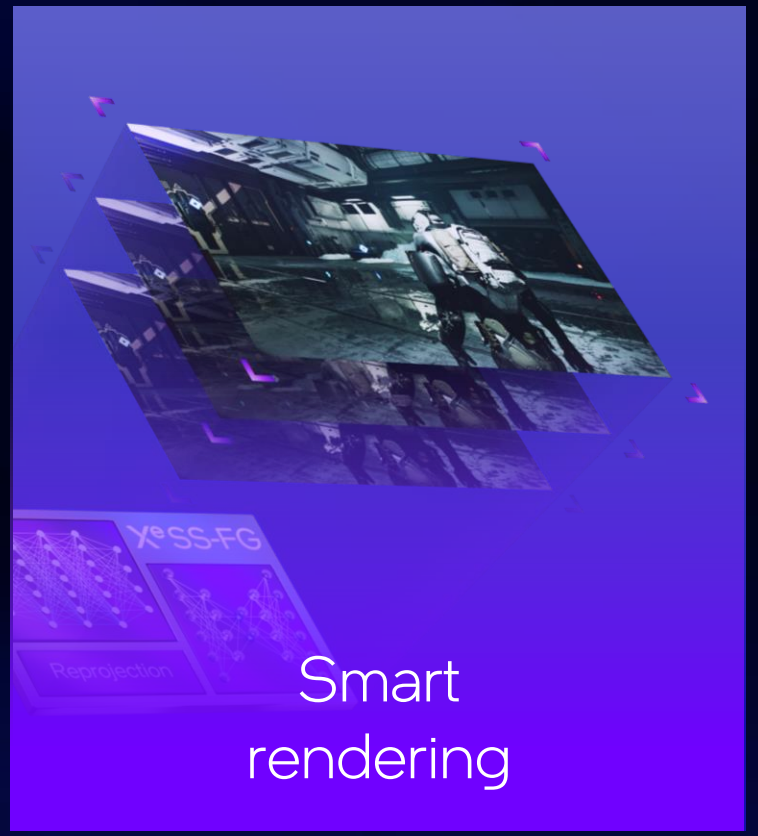
Powerful CPUs, and the emergence of built-in graphics with discrete-level performance at much lower power is making Ultra Slim Gaming a reality

Source: 2023 Laptop Buyers Survey, commissioned by Intel; Windows Gaming Laptop >$500

What Do We Need for Good Gaming at Low Power?



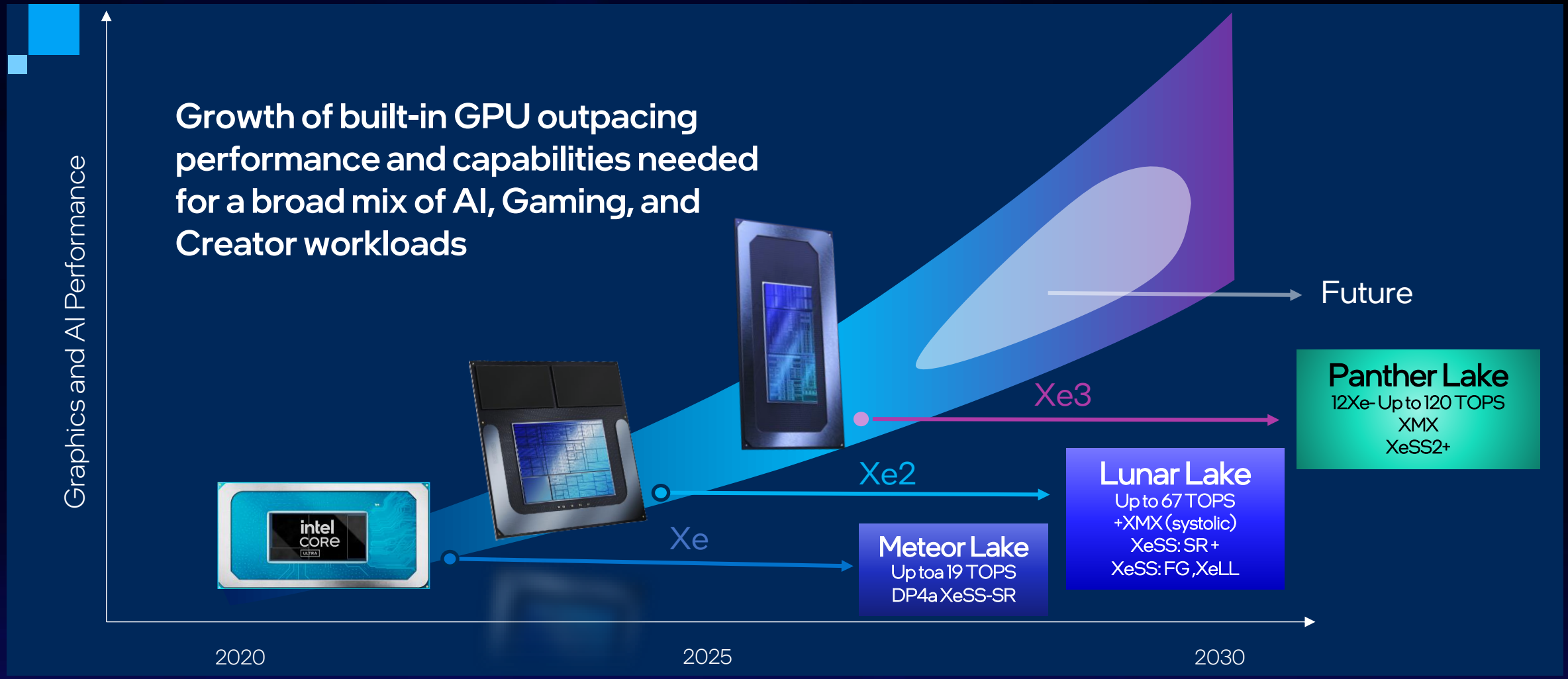
High-performance GPU IP



Smart rendering

Platform tuning

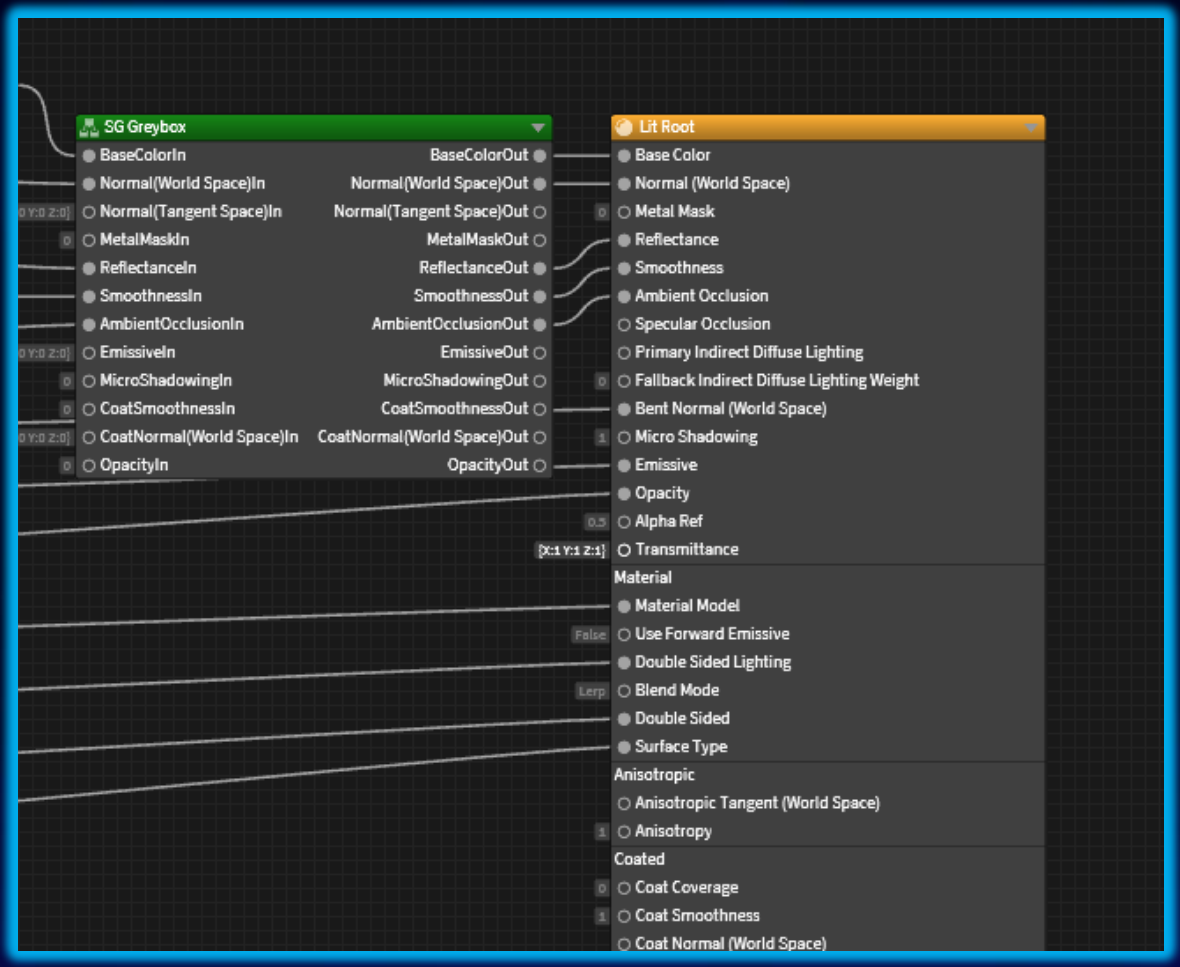
Evolution of Mobile PC Gaming



All product plans and roadmaps are subject to change without notice.

The Pipeline

Expression Shaders



- New data-driven shader graph system.
- New workflows and runtime.
- More in Simon Taylor's **Flexible and Extensible Shader Authoring in Frostbite with Serac** from Advances in Real-Time Rendering 2024

Expression Shaders - Instancing

```
//---------------------------------------------------------------------
component SvbInput
{
    [Domain("PerFrame")]
    extern StructuredBuffer<float4> Buf;

    [Domain("PerInstance")]
    extern uint Offset;
```

Parameters can be marked PerInstance.

All PerInstance parameters in a shared InstanceBuffer

```
var_93 = system_InstanceBufferSrv0[system_InstanceBufferOffset + vsOut.interp_3].ExpressionShaderParameter_common_
var_94 = system_InstanceBufferSrv0[system_InstanceBufferOffset + vsOut.interp_3].ExpressionShaderParameter_common_
var_95 = system_InstanceBufferSrv0[system_InstanceBufferOffset + vsOut.interp_3].ExpressionShaderParameter_common_
var_96 = system_InstanceBufferSrv0[system_InstanceBufferOffset + vsOut.interp_3].ExpressionShaderParameter_common
```

Expression Shaders - GPU Driven Batching

Multiple instanced draws packed inside ExecuteIndirect

```
Execute Indirect:
- DrawInstanced(1024, 5, startIndex, baseVertex, 0)
- DrawInstanced(2048, 1, startIndex, baseVertex, 0)
- DrawInstanced(4096, 3, startIndex, baseVertex, 0)
```

GPU Culling

```
Execute Indirect:
// 3 instances culled by gpurp.
- DrawInstanced(1024, 2, startIndex, baseVertex, 0)
// entire draw culled and compacted
//
// No instanced culled
- DrawInstanced(4096, 3, startIndex, baseVertex, 2)
```

```
[Merged instances]
  ExecuteIndirect(obj#10453,7,obj#219,60260,obj#1383,4776)  {this->ID3D12GraphicsCommandList obj#10844}
    DrawIndexedInstanced(207,3,24356000,0,5578)  {this->ID3D12GraphicsCommandList obj#10844}
    DrawIndexedInstanced(684,3,143277236,0,5581)  {this->ID3D12GraphicsCommandList obj#10844}
    DrawIndexedInstanced(1092,3,122541748,0,5584)  {this->ID3D12GraphicsCommandList obj#10844}
    DrawIndexedInstanced(2457,3,181106146,0,5587)  {this->ID3D12GraphicsCommandList obj#10844}
```

Expression Shaders - Raster Pass

```
ExpressionShaderInstanceKey instanceKey = ExpressionShaderInstanceUtil::createVariationInstanceKey(subset->getEsInstanceKey()
ExpressionShaderInstanceBuilder shaderInstance = context.buildInstance(instanceKey, { inputLayout });
```

```
virtual void processInstance(const IPermutationProviderContext& context, InstanceEditor& instance) override
{
    // Only add to material bake instances
    if (!instance.getParameters().doesContain(isMaterialBakeParam()))
        return;

    LightingModeType lightingType = instance.getStaticOutputs().getStatic(lightingModeStaticOutput);
    if (lightingType == LightingModeType_Lit)
    {
        instance.addPass({ getMaterialBakePass() });
    }
}
```

```
FB_IMPLEMENT_EXPSHADER_RASTER_PASS_EX(
    MaterialBake,
    Serac_MaterialBake,
    getRenderWithCbr(),
    getOpaqueRenderState(),
    getDefaultSortMethod(),
    FB_EXPSHADER_TARGET_FORMATS_NO_DEPTH(ResourceFormat_R8G8B8A8_UNORM, ResourceFormat_R8G8B8A8_UNORM),
    PrimitiveTopologyType_Triangle);
}
```

Expression Shaders – Shader Stuttering

Expression Shaders – PSO

Shader Programs

HLSL usually authored by engineers

```
[numthreads(THREADS_PER_WAVEFRONT, 1, 1)]
void main(uint2 groupID : SV_GroupID, uint threadID :
SV_GroupThreadID)
{
    // groupId.x - copyCommand index
    // groupId.y - dispatch group index
    const CopyCommand command = CopyCommands[groupID.x];
    const uint offset = groupID.y * THREADS_PER_WAVEFRONT + threadID;
    if (offset < command.m_dimension)
        FastCopyTarget[offset + command.m_dstOffset] =
            FastCopySource[offset + command.m_srcOffset];
}
```

Permutations are generated by pipeline code

```
ShaderProgramDef program;
program.shaderProgram = ShaderProgram_BatchCopyBufferCs;
program.source =
FB_RENDER_SYSTEM_DATA_PATH("PostProcess/BatchCopyBufferCs.hlsl");
program.csName = "main";
outPrograms.push_back(program);
```

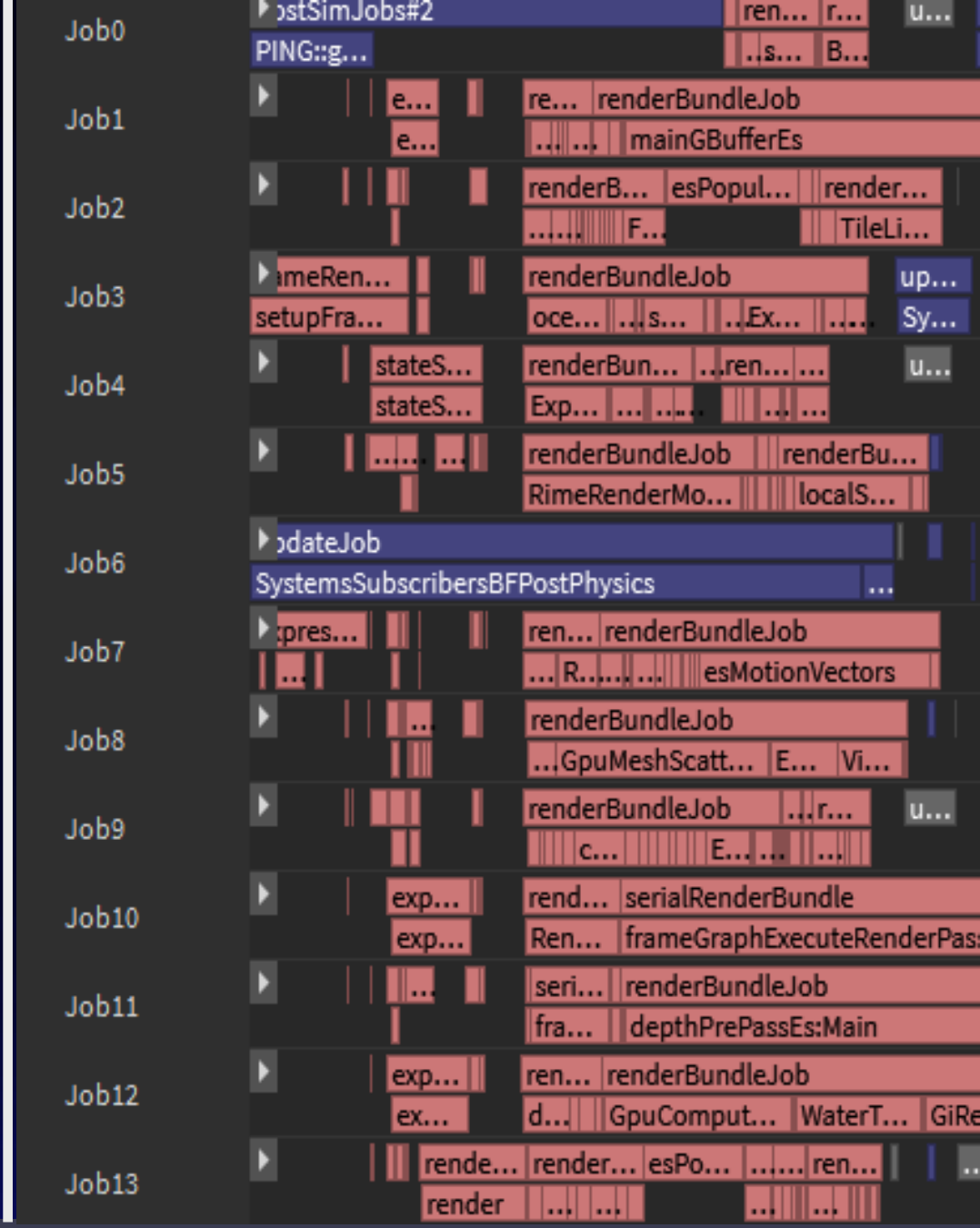
Shader Programs – PSO

Journal Encountered PSO

Replay the journal on boot

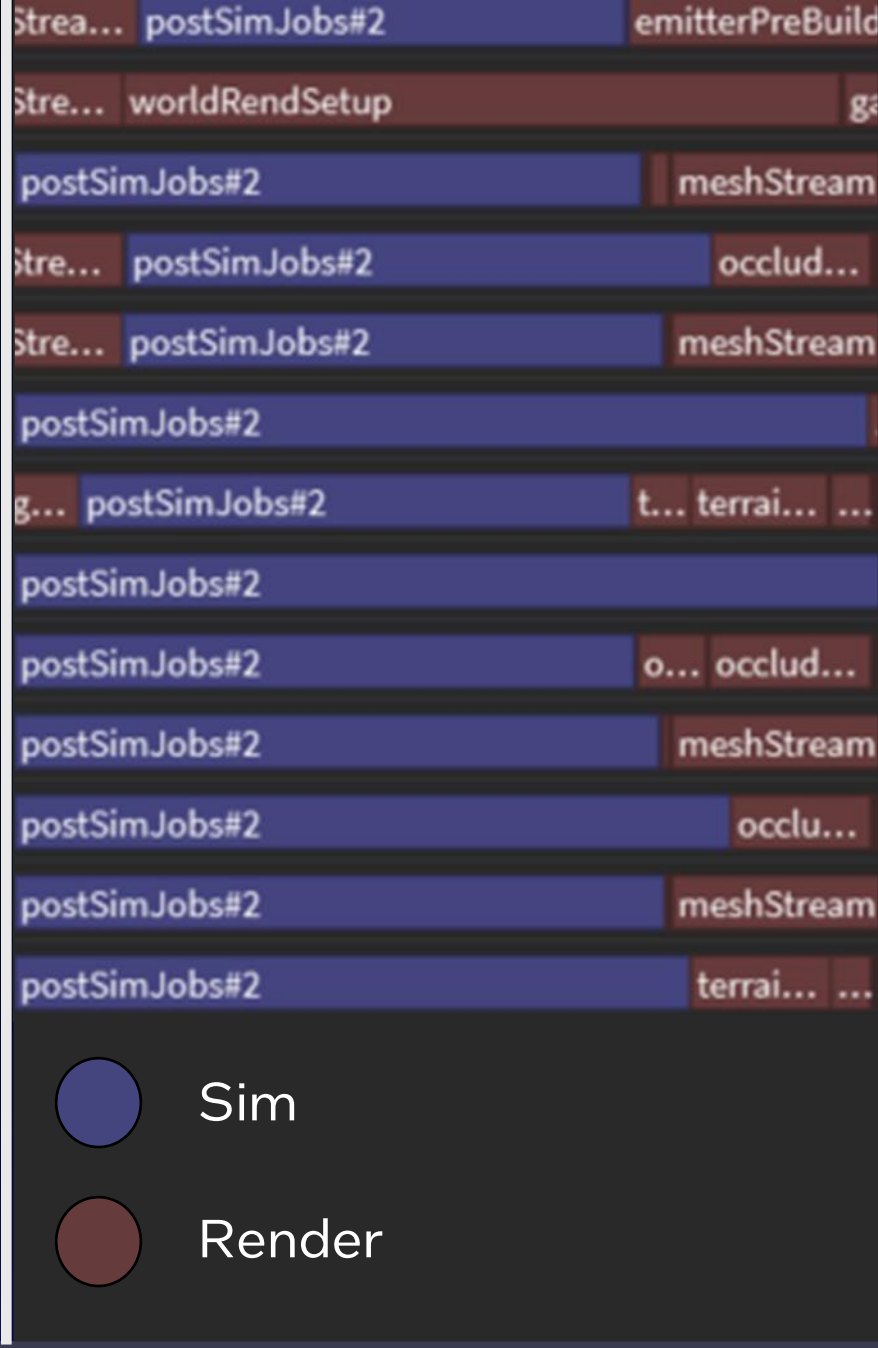
Multithreading

- Frame is a big job graph.
- Number of job threads scales with hardware threads.
- Job threads are not pinned to hardware threads on PC.
- Optimize for all hardware.
- Resource loading doesn’t block render or sim loop.
- Sim and Render are decoupled.

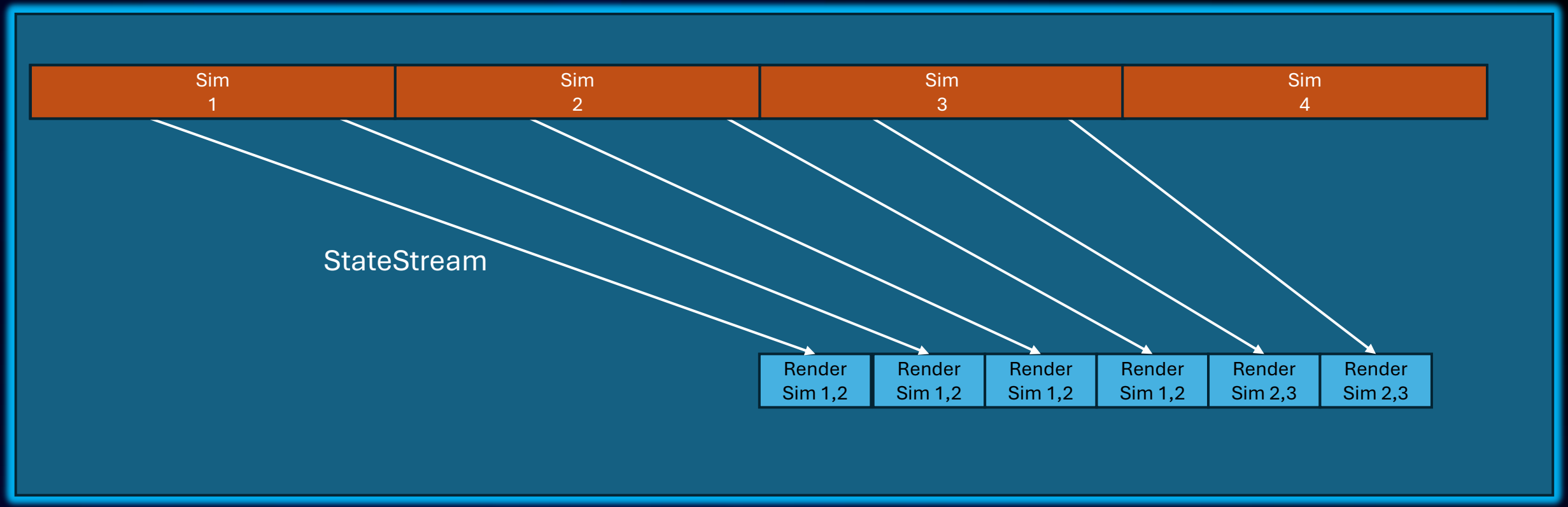


Sim and Render

- Sim and Render can run at the same time and at different frequency.
- Sim and Render run interleaved on the same threads on PC.
- Bias job threads towards Sim or Render jobs to avoid starvation.
- Latency critical actions bypass Sim.

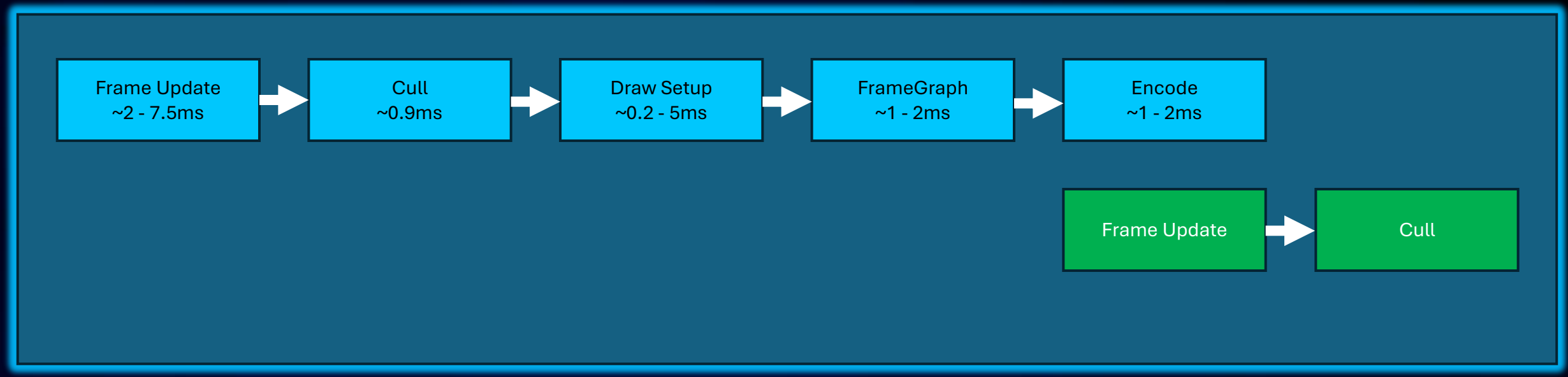


Decoupled Sim and Render



Walking Through a Frame

CPU Frame – Anatomy of a Render Frame



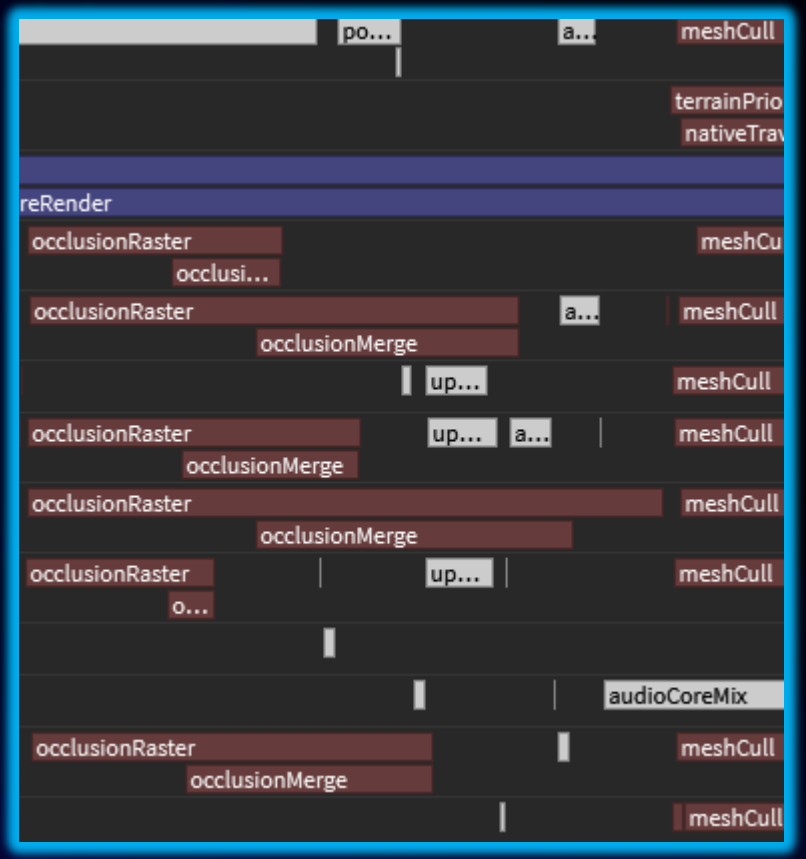
Timings for illustrative purposes only

CPU Frame – Update

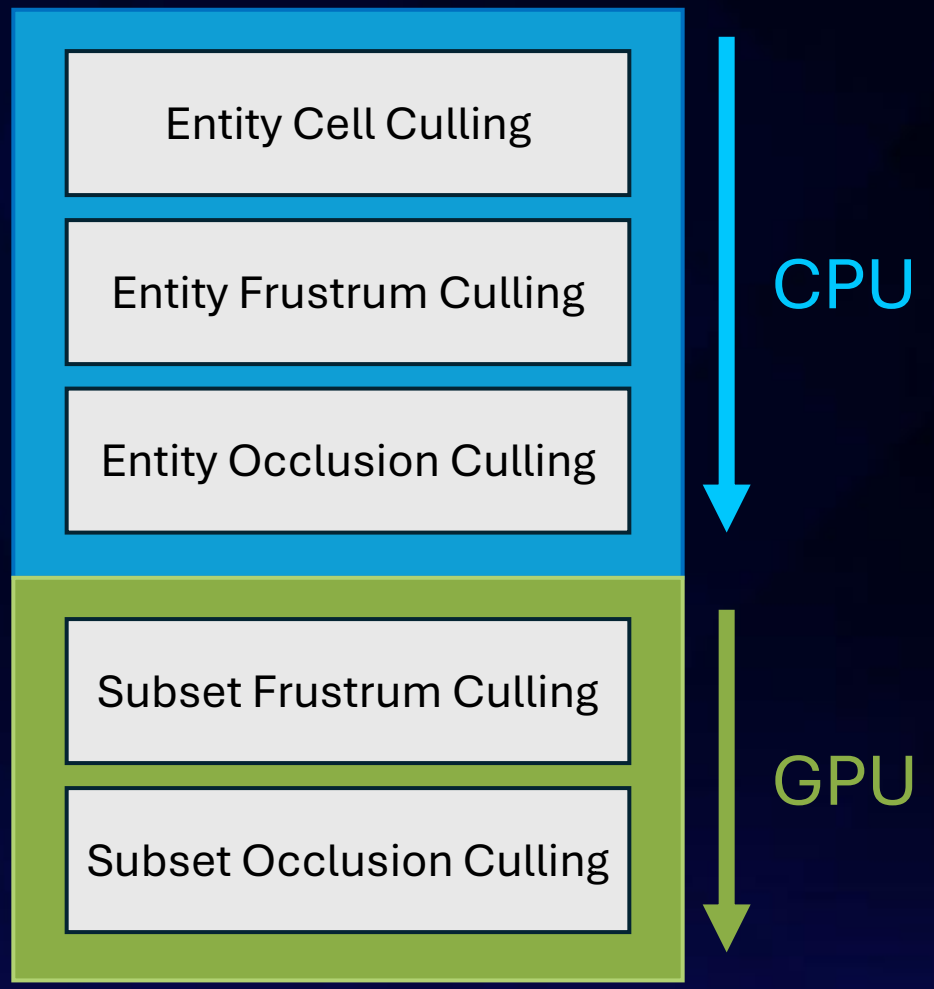
- Interpolate sim changes to render (StateStream).
- Overlapped with previous frame encoding on PC.
- Set up views and job graph.
- Process resource load/unload.
- Recycle memory and rendering objects.

CPU Frame – Culling

- Decreases cost of the rest of the frame.
- Benefits both cpu and gpu.
- Distance, frustum, size cull 85% of objects.
- Occlusion culling increases that to 95%.
- Optimized with SIMD and multithreading.
- For details see Daniel Collin's **Culling the Battlefield Data Oriented Design in Practice** GDC 2011



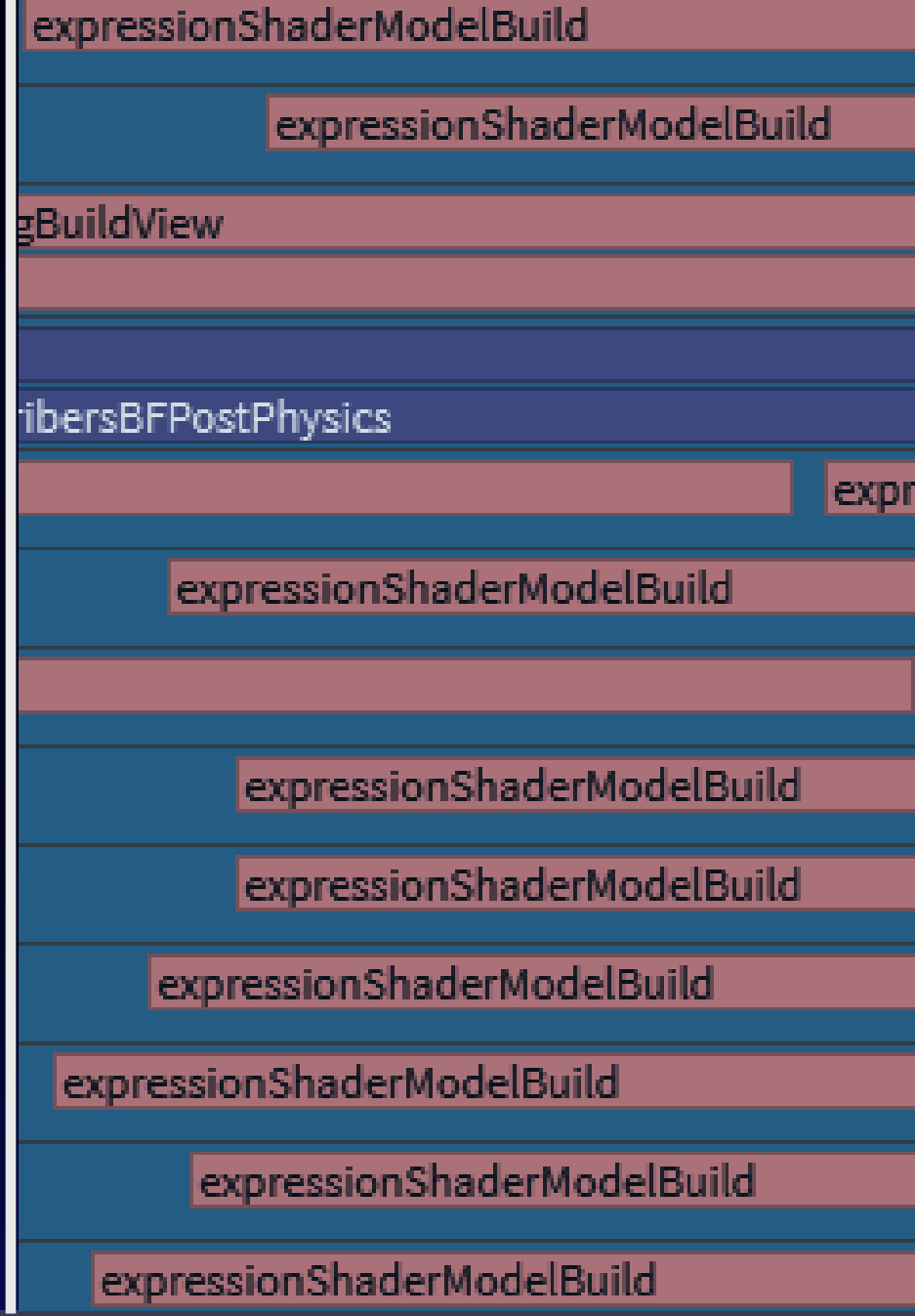
CPU Frame – Mesh Culling Pipeline



CPU Frame - Draw Setup

Model renderer takes longest:

- Expand entities to subsets.
- Resolve permutation and parameters of newly visible or dirty objects.
- Price of dynamic renderer features.
- Significant effort into reducing thread contention.



CPU Frame – FrameGraph Setup

```
auto& paramIO = blackboard.get<WorldRendererParamIO>();

data.colorTexture = builder.read(colorTexture);

// We must clear the target (to zero), as the entire target is not written to.
data.output = createIntermediateRenderTarget(builder, blackboard, "SmaaEdgeTexture", true);

if (paramIO.rootView->viewSettings->getSmaaPredicatedThresholdingEnable())
{
    data.depthTexture = readDepthTexture(builder, blackboard);
}
```

- Build dependency graph. Set up memory aliasing.
- Resource creation very expensive on PC, especially CreateCommittedResource.
- Use Placed Resources or reuse from previous frames.
- Single thread, but usually not on the critical path.

More on FrameGraph in Yuriy O'Donnell's **FrameGraph: Extensible Rendering Architecture in Frostbite** GDC 2017

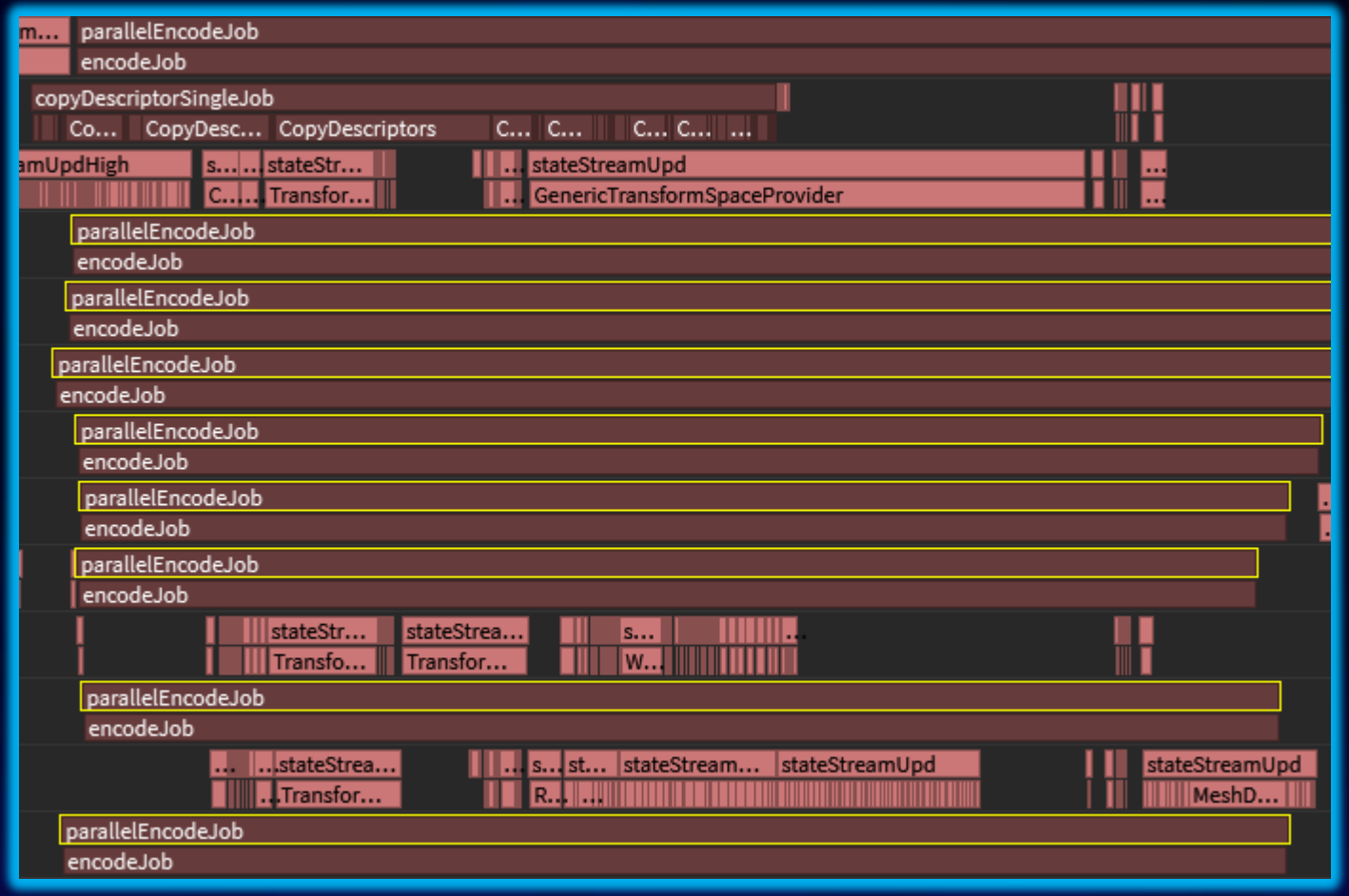
CPU Frame – FrameGraph Execute

```
commandRecorder.beginRenderPass(renderTarget.asRenderPassDesc("Screen
commandRecorder.setPipelineRasterizerState(rasterizerStateDesc);
commandRecorder.setPipelineDepthStencilState(depthStencilStateDesc);
commandRecorder.setPipelineBlendState(blendDesc);
commandRecorder.setPipeline(info.pipeline);
```

- On the critical path.
- Parallel by default.
- Rc2CommandRecorder – fake command list.

More on Rc2 in Jason Bright's Bringing Frostbite to New Rendering Tech and Platforms (While Nobody Noticed) GDC 2024

CPU Frame – Encode



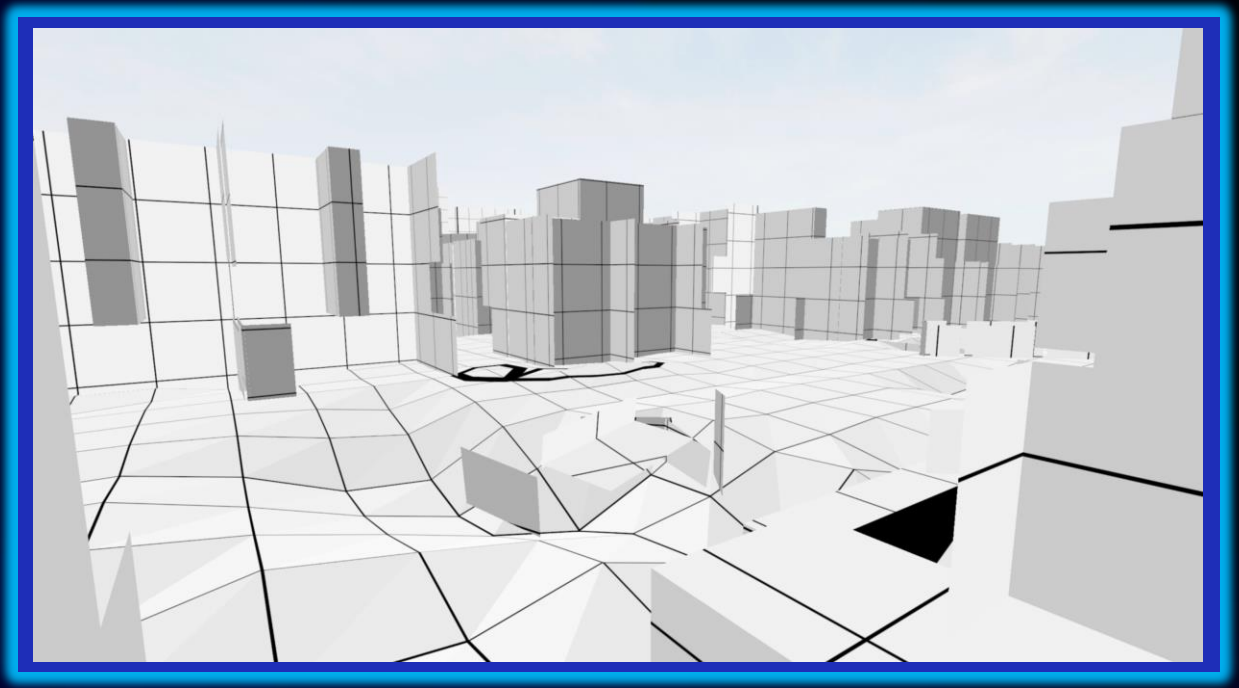
Next frame

Current frame

GPU Culling

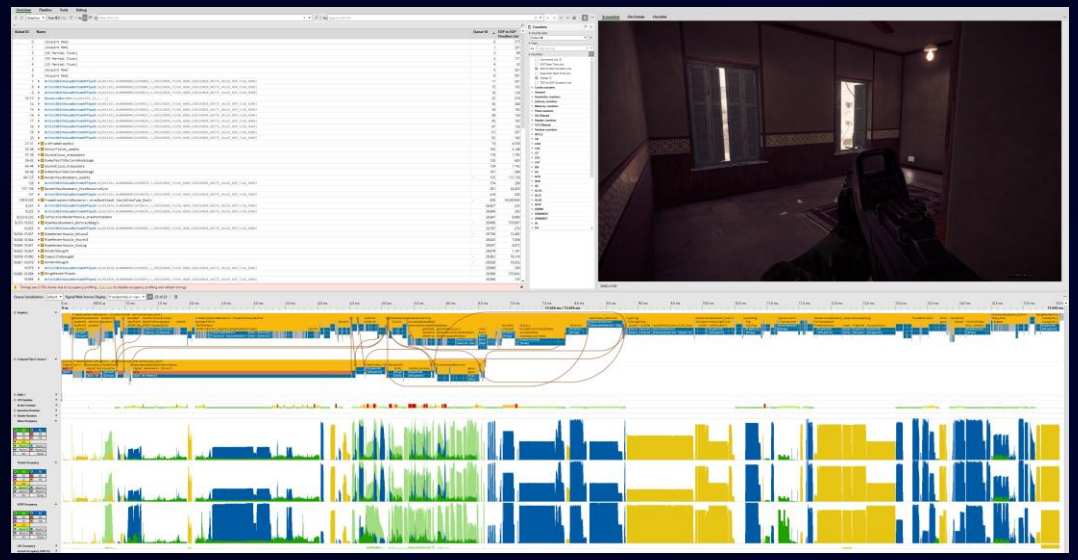
GPU Culling Improvements

- Helps when CPU occluders can’t (i.e. destructible mesh)
- Reducing VS work in typical gameplay scenarios
- Helps scalability with upscaling and dynamic resolution scaling
- Still faster in a worst-case scenario (compared to 2042)



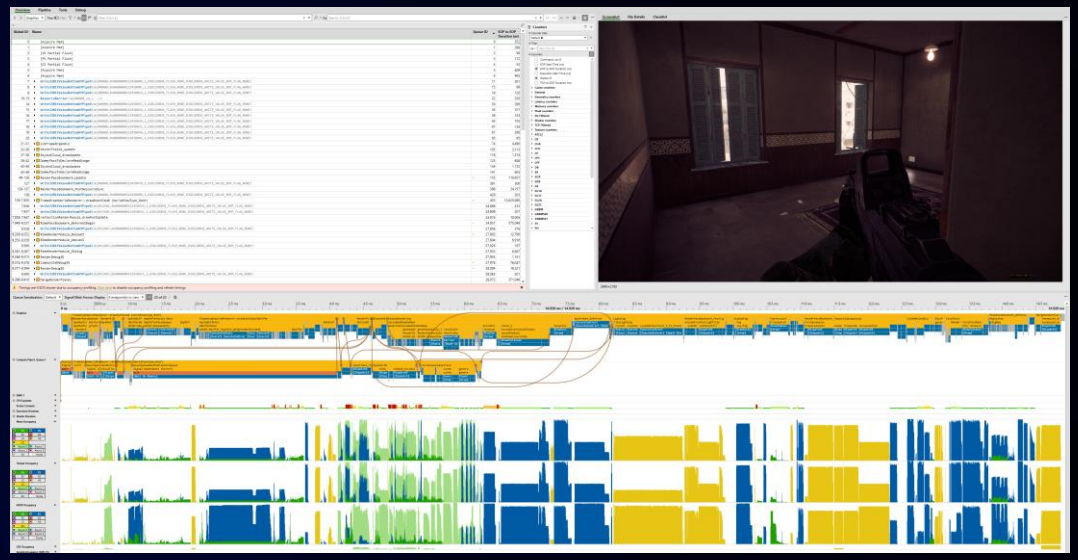
GPU Culling Improvements

Frustum + Occlusion Culling: Disabled



GBuffer: 1.94ms GPU time

Frustum + Occlusion Culling: Enabled

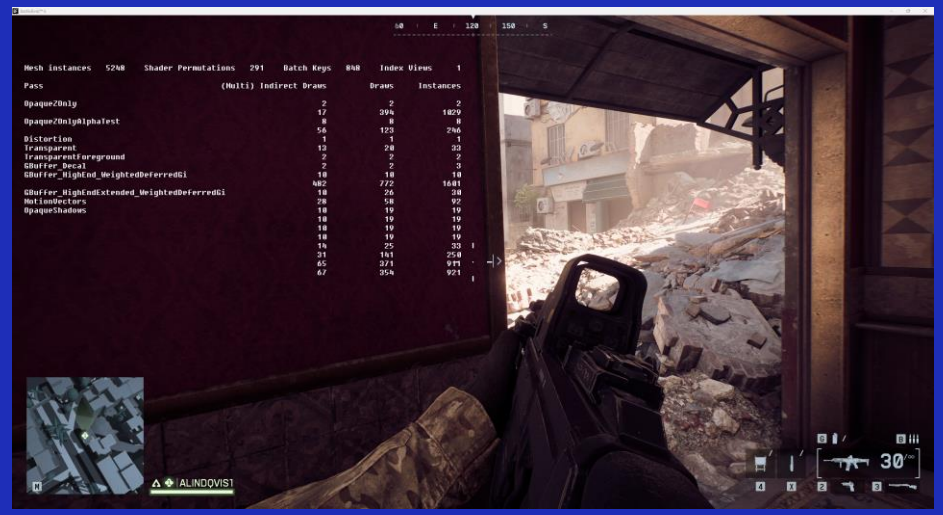


GBuffer: 1.56ms GPU time

Visualization for illustrative purposes only. Timing data provided by EA Dice on development hardware, performance may vary based on hardware configuration.

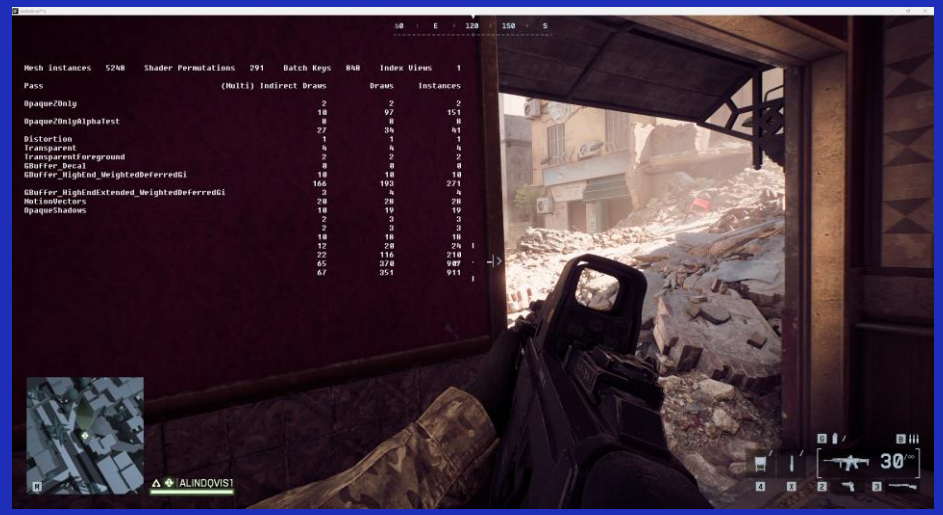
GPU Culling Improvements

Frustum + Occlusion Culling: Disabled



Mesh instances 5248	Shader Permutations 291	Batch Keys 848	Index Views 1
Pass	(Multi) Indirect Draws	Draws	Instances
OpaqueZOnly	2	2	2
	17	394	1029
OpaqueZOnlyAlphaTest	8	8	8
	56	123	246
Distortion	1	1	1
Transparent	13	20	33
TransparentForeground	2	2	2
GBuffer_Decal	2	2	3
GBuffer_HighEnd_WeightedDeferredGi	10	10	10
	482	772	1601
GBuffer_HighEndExtended_WeightedDeferredGi	10	26	30
MotionVectors	28	58	92
OpaqueShadows	10	19	19
	10	19	19
	10	19	19
	10	19	19
	14	25	33
	31	141	250
	65	371	911
	67	354	921

Frustum + Occlusion Culling: Enabled



Mesh instances 5248	Shader Permutations 291	Batch Keys 848	Index Views 1
Pass	(Multi) Indirect Draws	Draws	Instances
OpaqueZOnly	2	2	2
	10	97	151
OpaqueZOnlyAlphaTest	8	8	8
	27	34	41
Distortion	1	1	1
Transparent	4	4	4
TransparentForeground	2	2	2
GBuffer_Decal	0	0	0
GBuffer_HighEnd_WeightedDeferredGi	10	10	10
	166	193	271
GBuffer_HighEndExtended_WeightedDeferredGi	3	4	4
MotionVectors	20	28	28
OpaqueShadows	10	19	19
	2	3	3
	2	3	3
	10	18	18
	12	20	24
	22	116	210
	65	370	907
	67	351	911

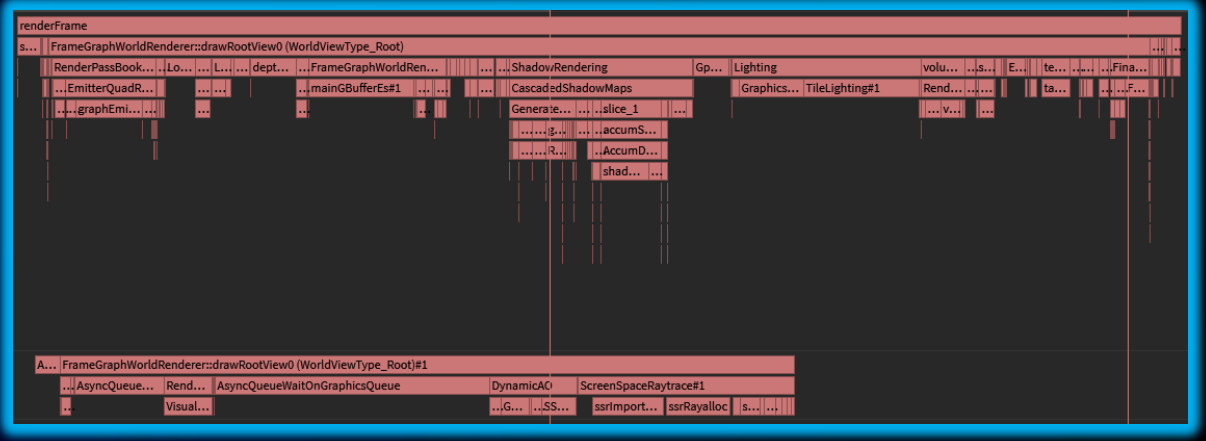
GPU Culling - Limitations

- No meshlet culling -> More coarse culling
- No bindless textures -> Less batching when we need to read material information
- No GPU persistent scene -> We draw depth prepass without GPU occlusion culling

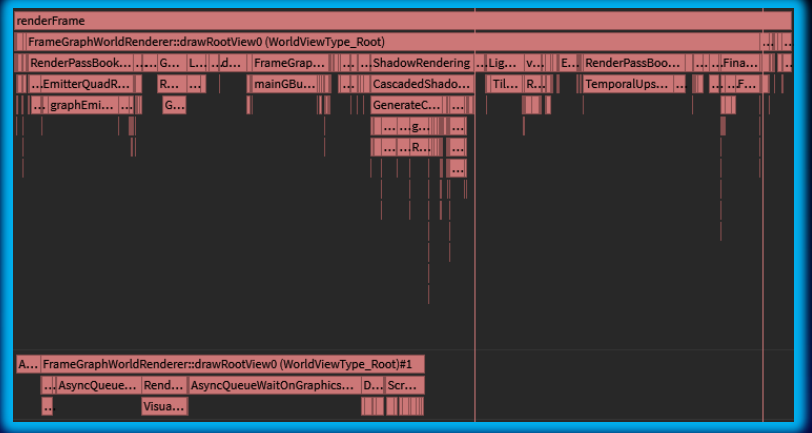
Upscaling - Performance

- Hardware accelerated
- Total Frame GPU cost
  - 13.9ms -> 9.0ms (54% uplift)
- Pre-Upscaling GPU cost
  - 12.2ms -> 6.5ms (88% uplift)
- 1440p High Settings on a B580

Native TAA



XeSS Balanced - 50% Scaling



Visualization for illustrative purposes only. Timing captured on Intel® Arc™ B580 @ 1440p High Preset

Frame Generation Integration

- We render UI in a separate buffer always.
  - Works well for FG quality.
- Also provide Hudless target for FG
- Use only one FG mode for each GPU for the sake of simplicity in code.
- Create FG swapchain using the same desc as the native engine swapchain.

Color Input

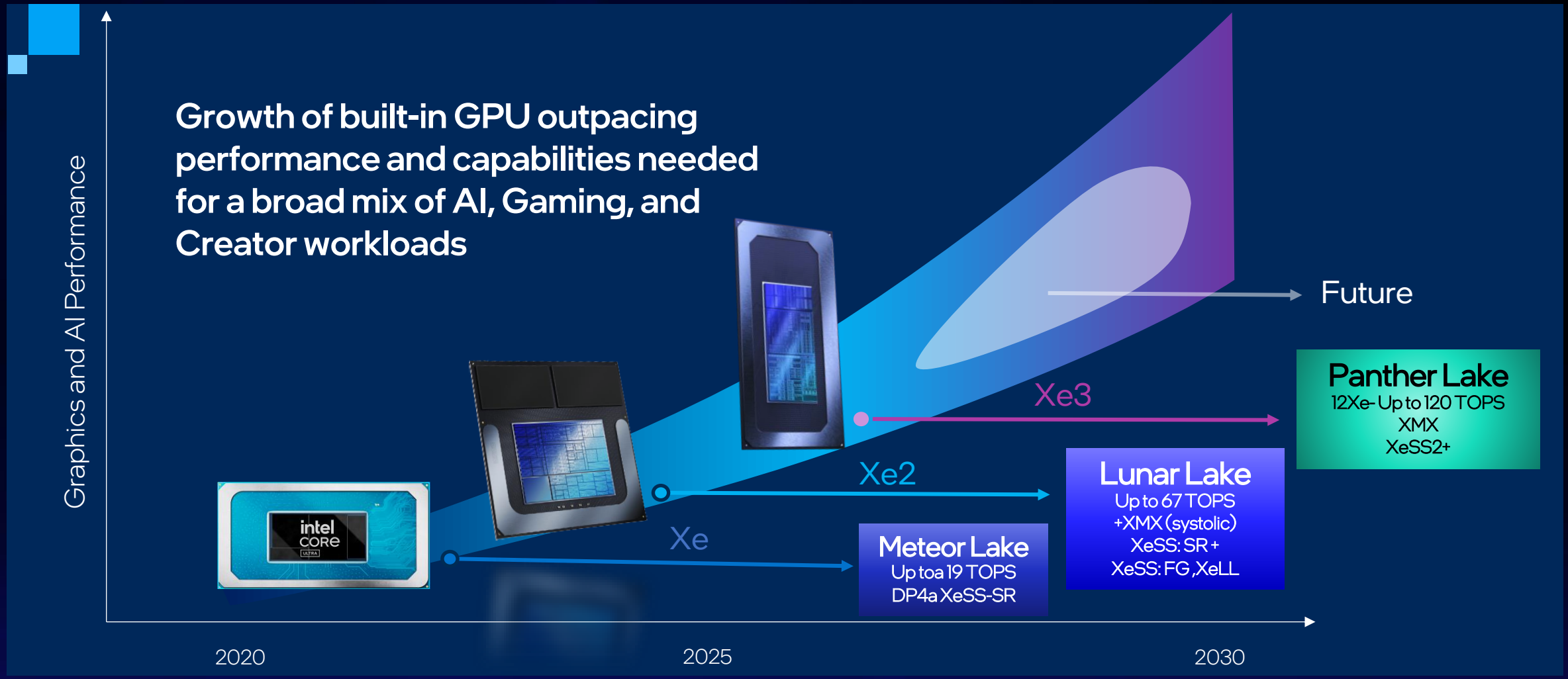
UI Input

Merged Output

Hudless Output

Why Battlefield 6 Performs well on Intel Built-In GPUs

Evolution of Mobile PC Gaming

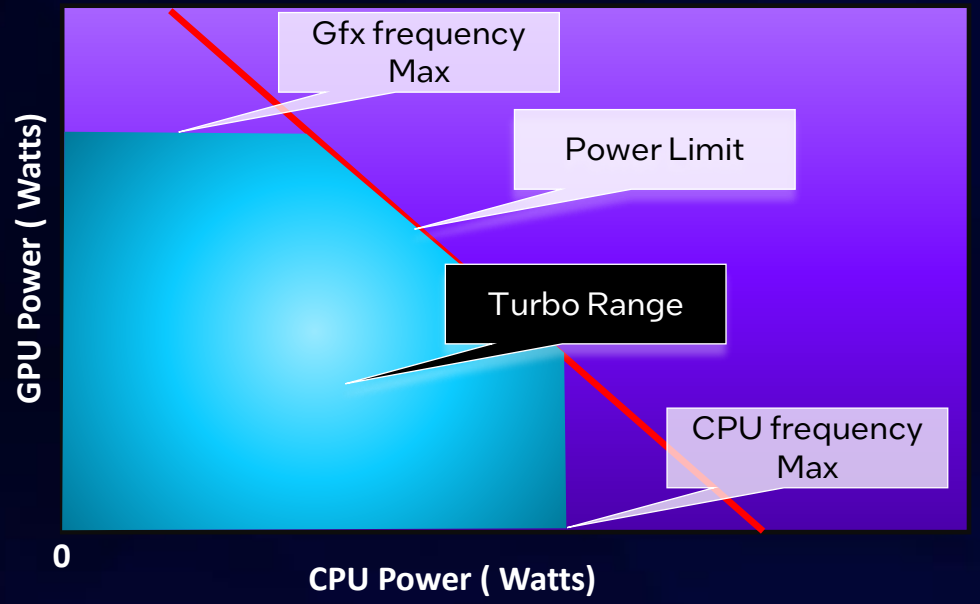


All product plans and roadmaps are subject to change without notice.

Panther Lake

Shared Thermal Design Point (TDP)

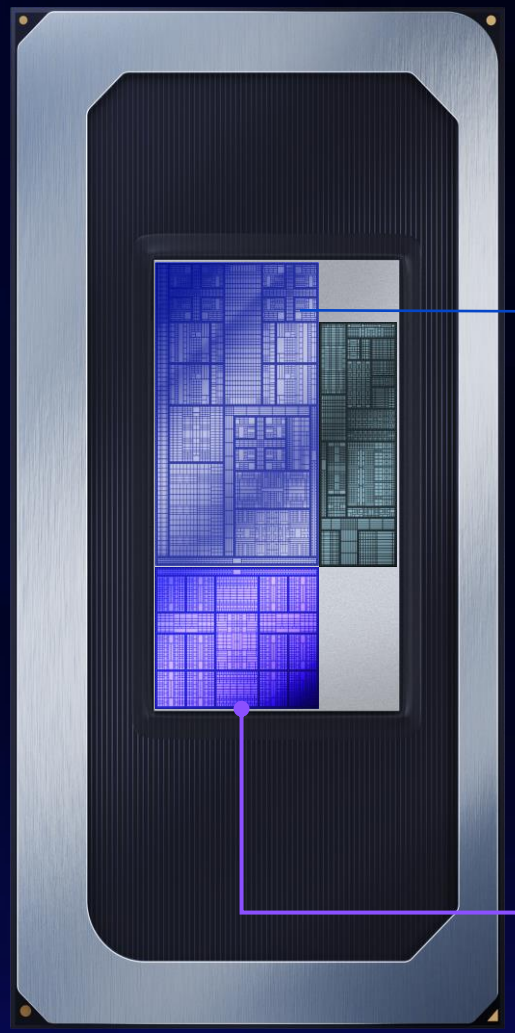
- CPU: 16 Cores
- GPU 12 Xe



Power ∝ Frequency²

Visualization for illustrative purposes only
See details from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex

Panther Lake
16 core 12Xᵉ



Compute tile — Built on intel 18A

CPU

up to 16 cores	P-cores 4 Cougar Cove	E-cores 8 Darkmont	LP E-cores 4 Darkmont

xPU

IPU 7.5 Image signal processor	NPU 5	Xᵉ Media & display engines

Memory

Up to 9600 MT/s LPDDR5x	8MB Memory-side cache

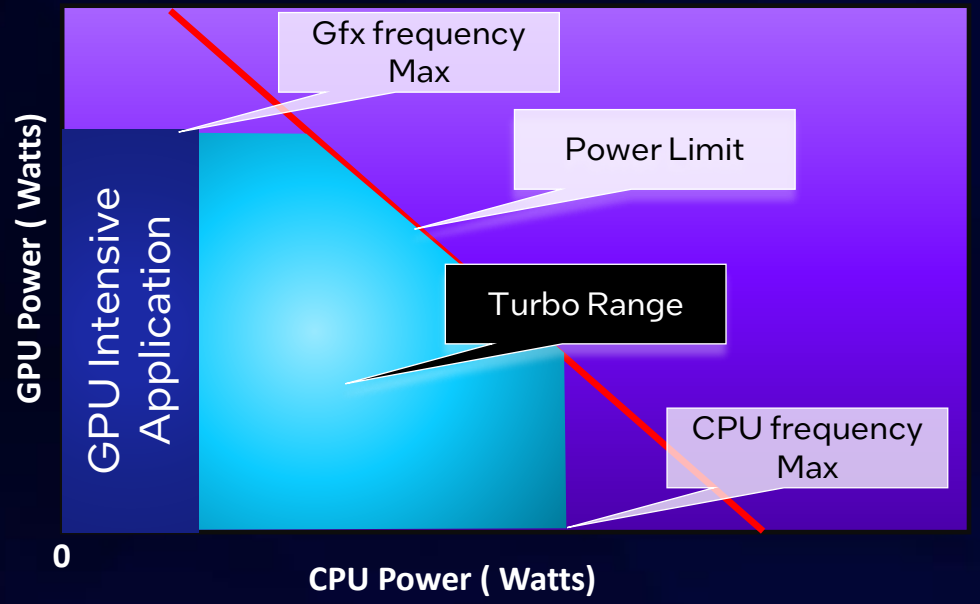
GPU tile — Built on External

Xᵉ3 Architecture	12 Xe-cores	12 Ray tracing units

Panther Lake

Shared Thermal Design Point (TDP)

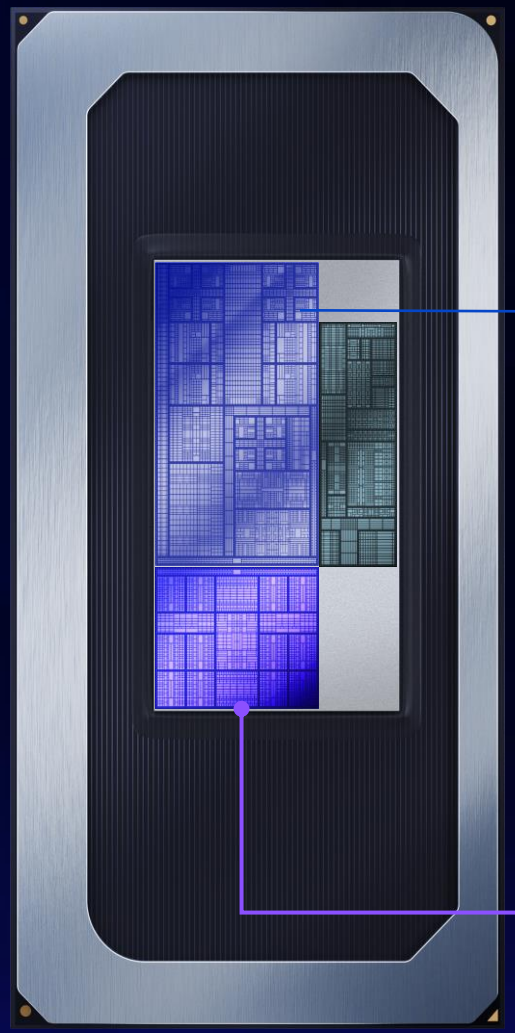
- CPU: 16 Cores
- GPU 12 Xe



High GPU Frequency
Limited CPU power available

Visualization for illustrative purposes only
See details from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex

Panther Lake
16 core 12Xe



Compute tile — Built on intel 18A

CPU

up to 16 cores	P-cores 4 Cougar Cove	E-cores 8 Darkmont	LP E-cores 4 Darkmont

xPU

IPU 7.5 Image signal processor	NPU 5	Xe Media & display engines

Memory

Up to 9600 MT/s LPDDR5x	8MB Memory-side cache

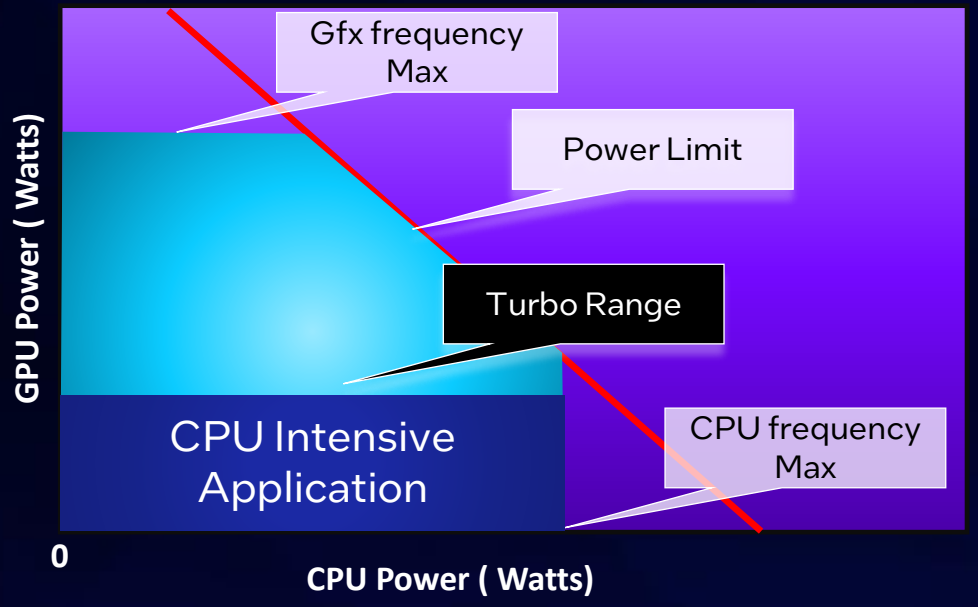
GPU tile — Built on External

Xe3 Architecture	12 Xe-cores	12 Ray tracing units

Panther Lake

Shared Thermal Design Point (TDP)

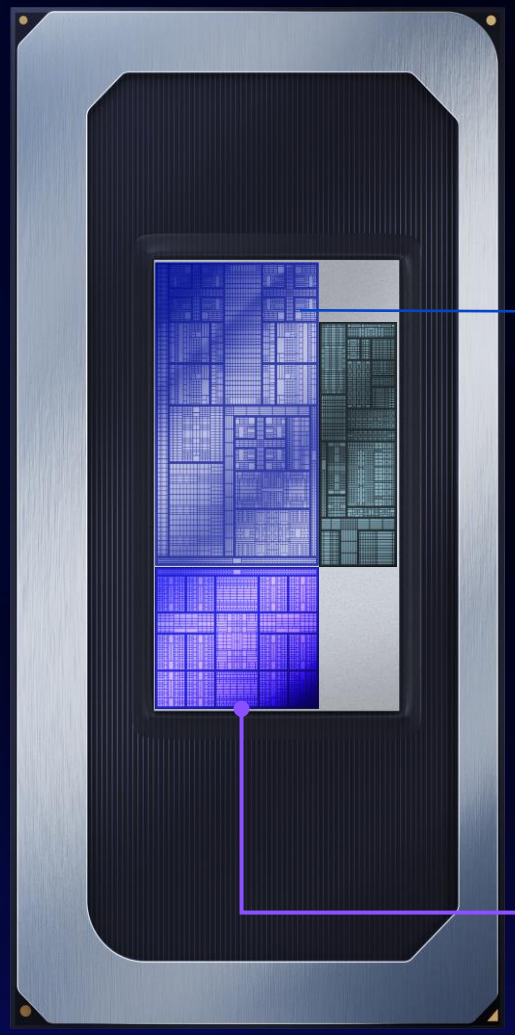
- CPU: 16 Cores
- GPU 12 Xe



High CPU Frequency
Limited GPU power available

Visualization for illustrative purposes only
See details from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex

Panther Lake
16 core 12Xe



Compute tile — Built on intel 18A

CPU

up to 16 cores	P-cores 4 Cougar Cove	E-cores 8 Darkmont	LP E-cores 4 Darkmont

xPU

IPU 7.5 Image signal processor	NPU 5	Xe Media & display engines

Memory

Up to 9600 MT/s LPDDR5x	8MB Memory-side cache

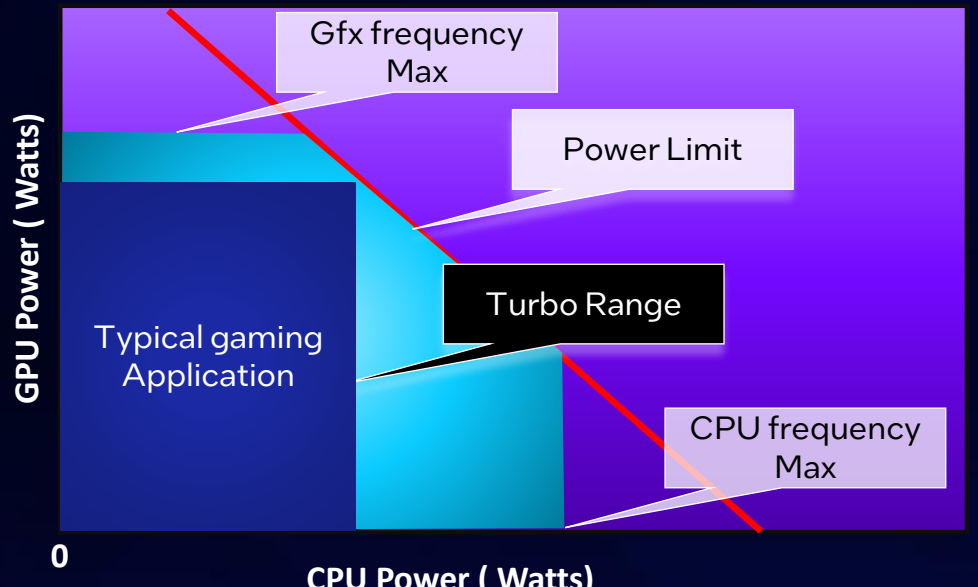
GPU tile — Built on External

Xe3 Architecture	12 Xe-cores	12 Ray tracing units

Panther Lake

Shared Thermal Design Point (TDP)

- CPU: 16 Cores
- GPU 12 Xe

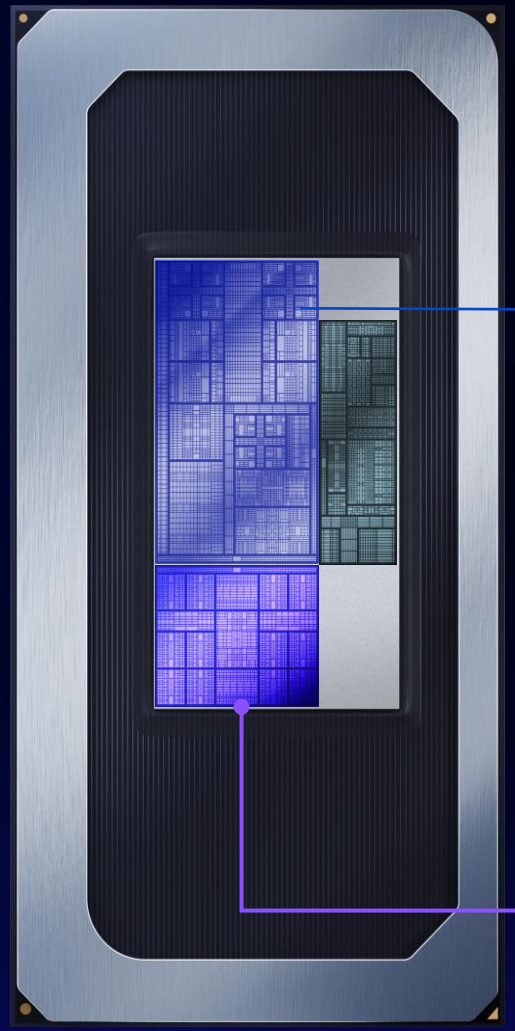


Push Me

Pull You

Visualization for illustrative purposes only
See details from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex

Panther Lake
16 core 12Xe



Compute tile — Built on intel 18A

CPU

up to 16 cores	P-cores 4 Cougar Cove	E-cores 8 Darkmont	LP E-cores 4 Darkmont

xPU

IPU 7.5 Image signal processor	NPU 5	Xe Media & display engines

Memory

Up to 9600 MT/s LPDDR5x	8MB Memory-side cache

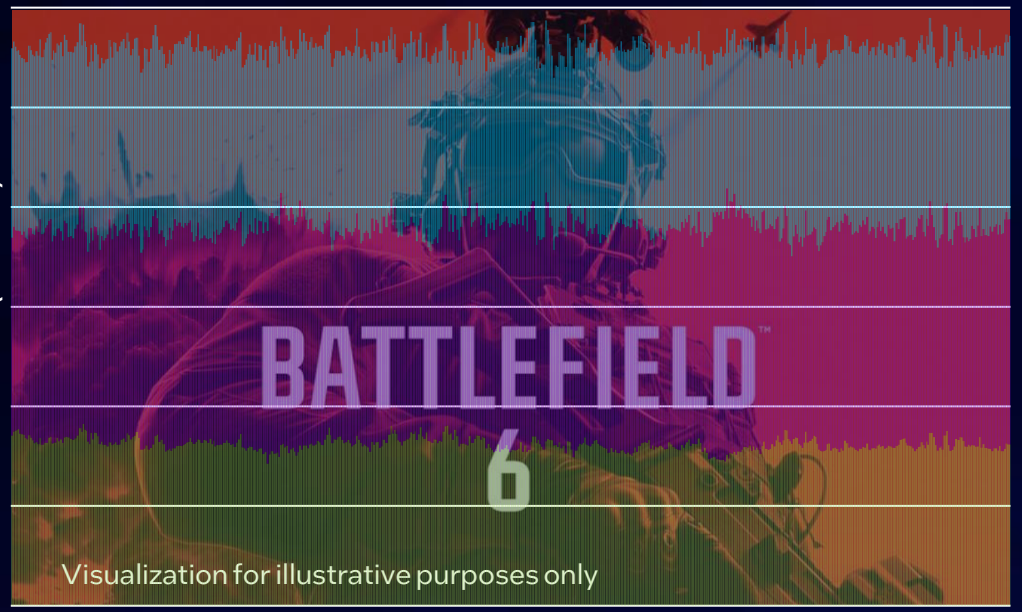
GPU tile — Built on External

Xe3 Architecture	12 Xe-cores	12 Ray tracing units

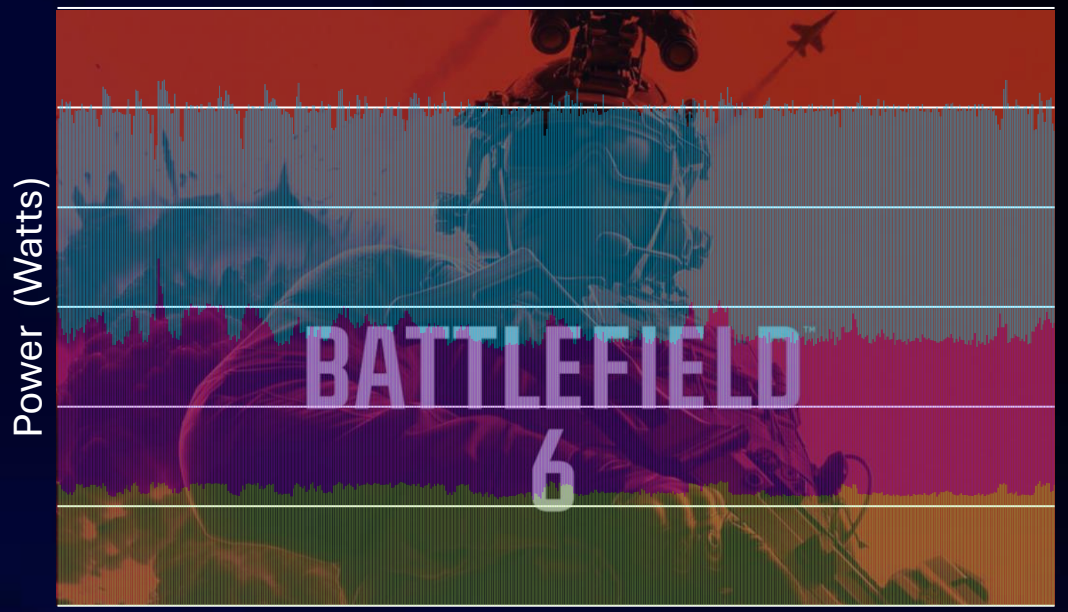
Optimized Power Delivery with Platform Tuning

Battlefield™ 6 Medium XeSS - SR Perf - 1080p - DX12

Previous Gen



Panther Lake



vs. Lunar Lake & Arrow Lake
>50%
more performance
GPU

vs. Arrow Lake
>30%
lower power at similar MT performance
CPU

vs. Lunar Lake
Up to
10%
lower power
SoC

Above projections based on Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex

Gives more headroom for larger GPU configurations

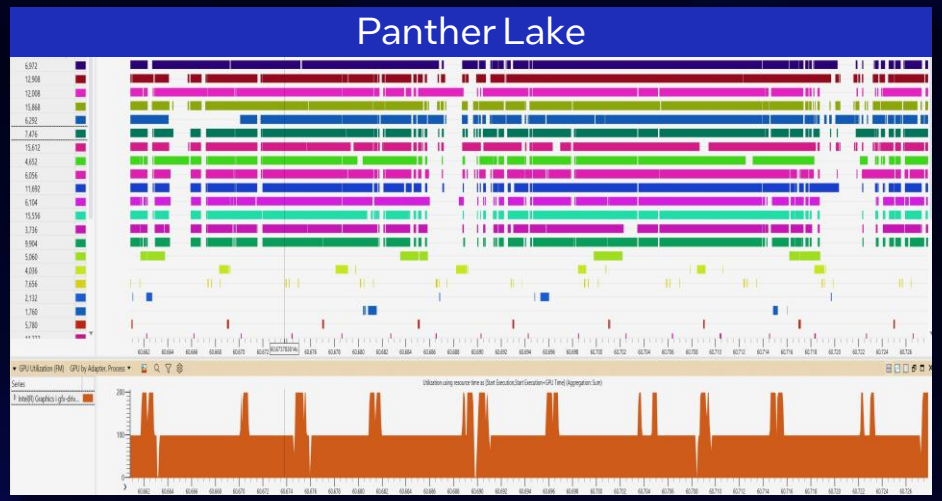
Platform Changes Relative to Lunar Lake tested during BF6 Playtest – See Appendix for details

Power Benefits of Multi-threading

Lunar Lake

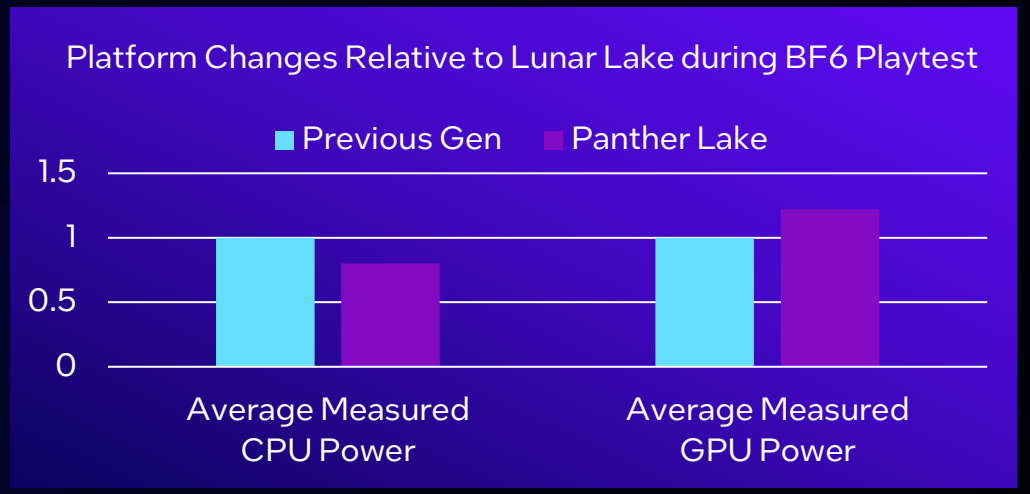
6 Threads running at up to 2.0-3.5GHz

Panther Lake

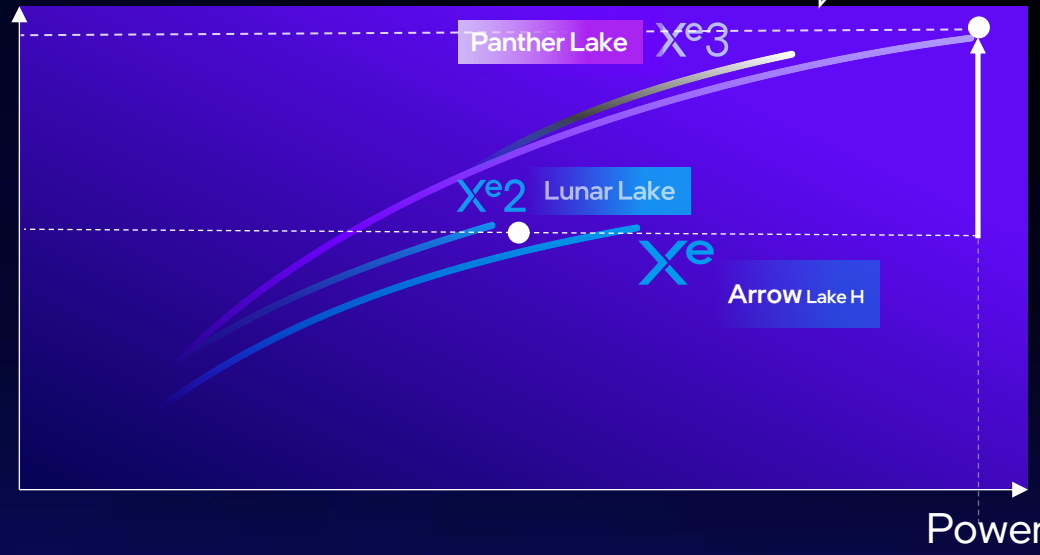


14 Threads running up to 1.2-1.3GHz

$P \propto F^2$



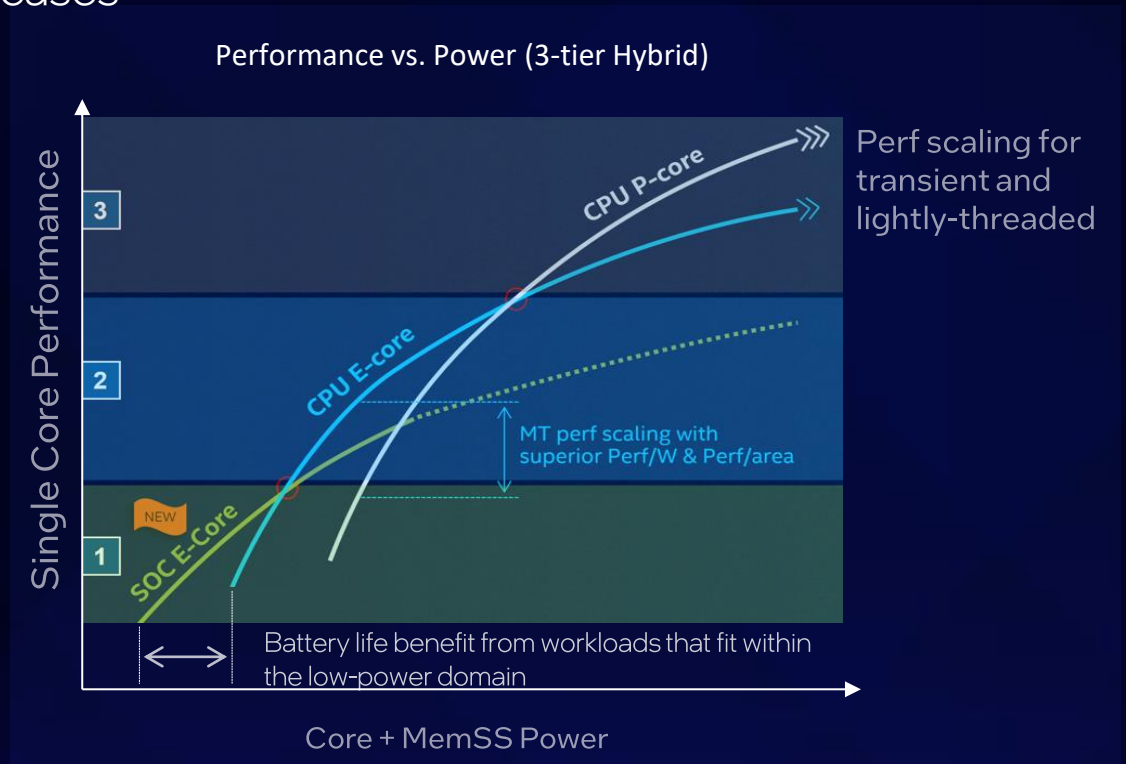
PTL- More Power to the GPU



See Appendix for testing details

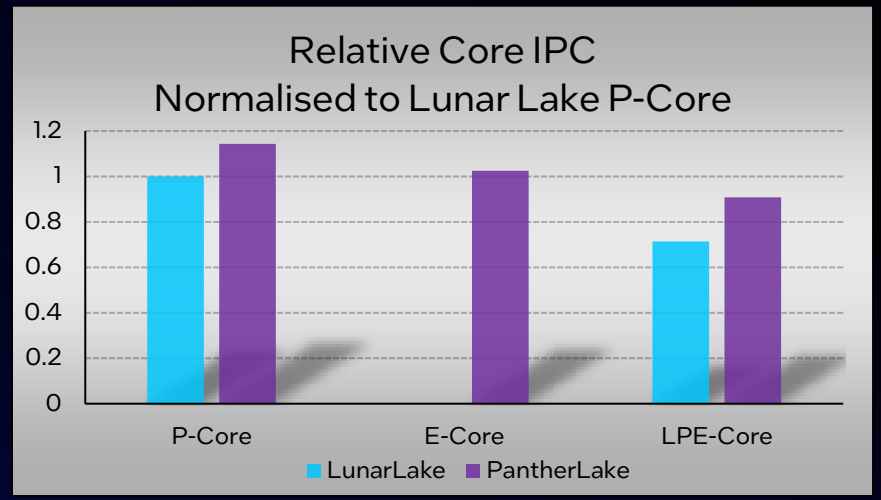
Panther Lake - Thread Director

Enhanced & optimized intelligent thread guidance across a wider set of use cases



Perf scaling for transient and lightly-threaded

Don't Pin Threads:
Best core can vary by power and workload



OS containment zones

Efficiency | Hybrid / compute | Zoneless

Relative Core IPC tested during BF6 Playtest – See Appendix for details

Intel® Core Processor Windows Scheduling/Parking

Windows Core Parking Engine: Makes global scalability decisions about the workload and determines the optimum set of compute cores for execution.

Power Management Parking Settings

Varies by power plan:

- CPMinCores
- CPMaxCores
- CPIncreaseTime
- CPDecreaseTime
- CPHeadroom

High Concurrency with Low utilization

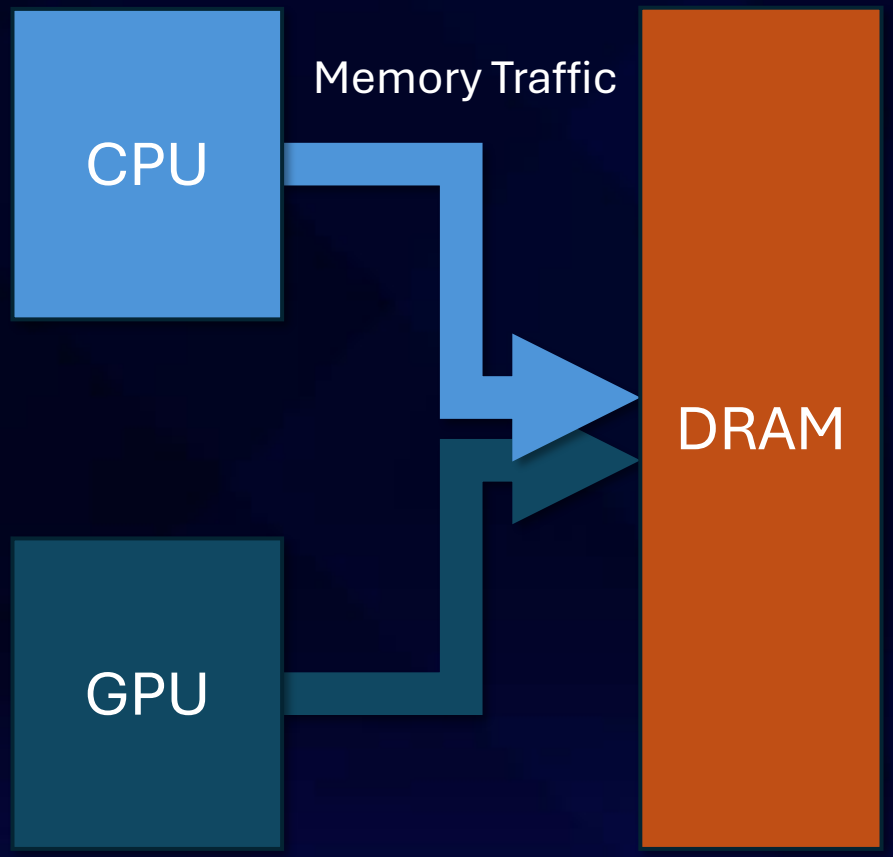
OS will see this and start parking cores

High Concurrency with High utilization

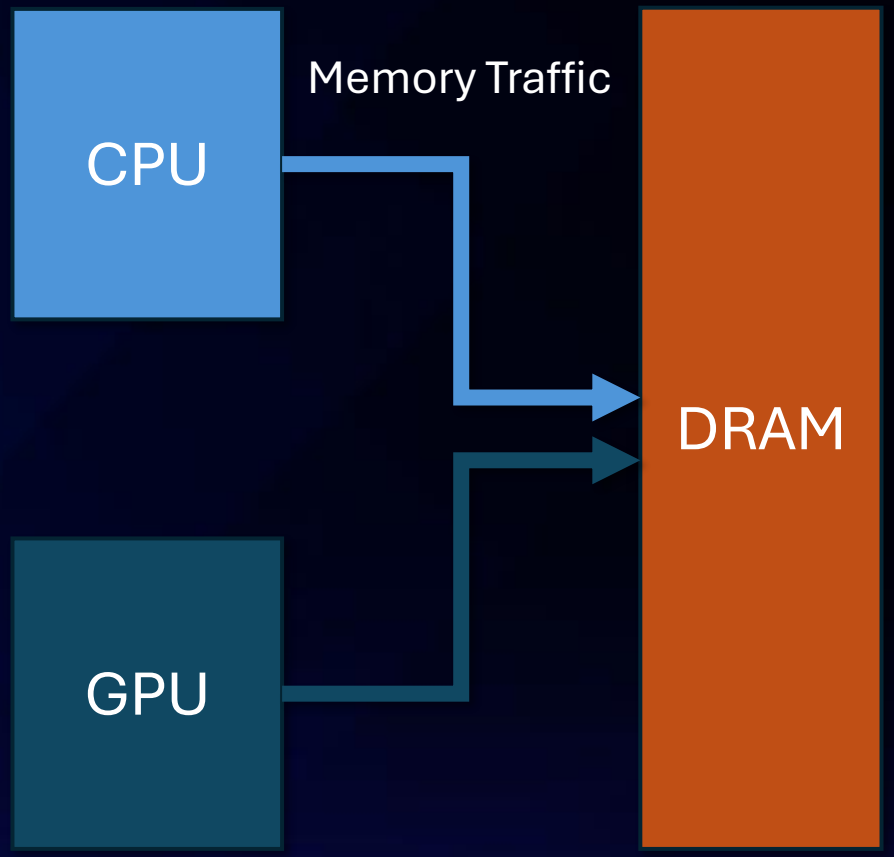
OS will keep consistent number of cores unparked

Taking Performance Further with Intel XeSS 2

Shared Memory = Shared Bottleneck



Native

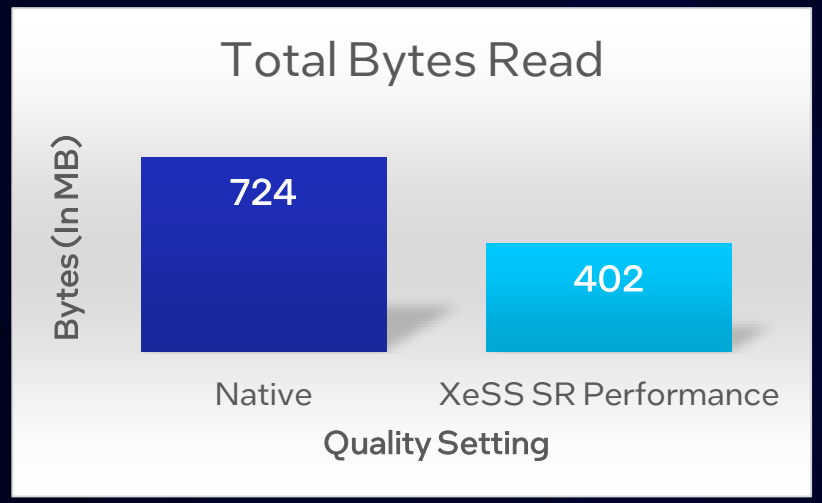


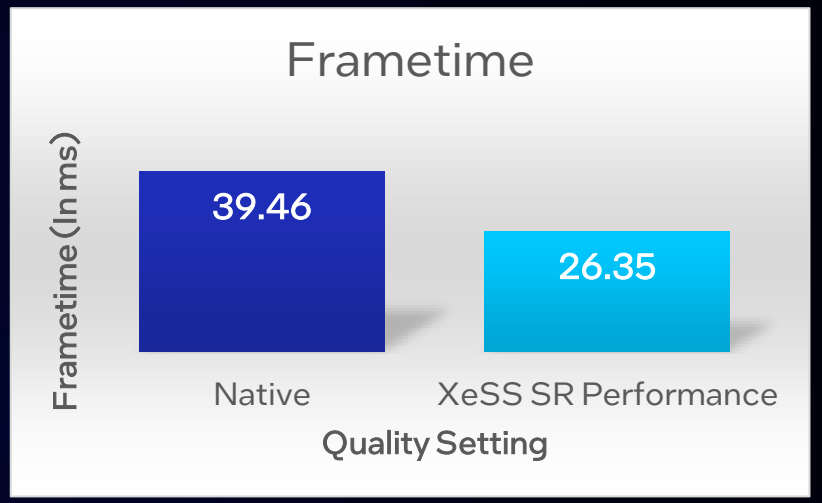
With XeSS SR

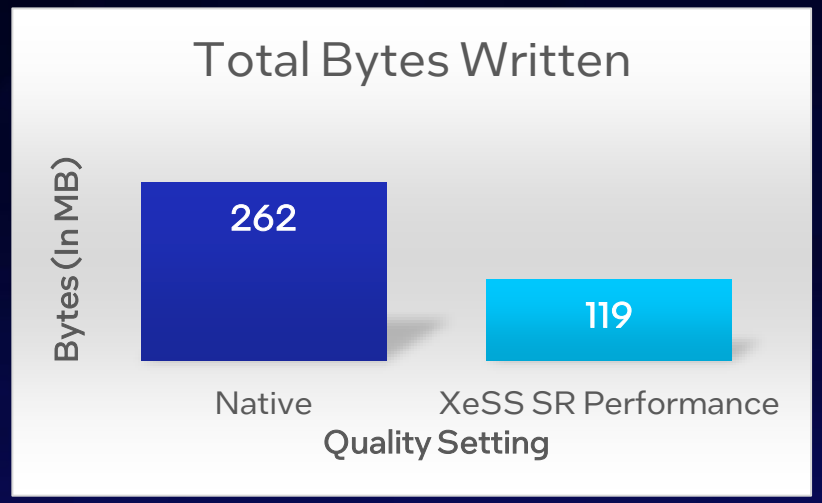
Visualization for illustrative purposes only

XeSS Super Resolution – Memory

Analysis of XeSS SR impact on Memory in a Battlefield 6 Frame







- At same output resolution, SR:
  - Reduced Memory Traffic by 47%
  - Decreased frame time by 33%

*PIX Captures taken on Lunar Lake MSI Claw 8 AI+ at High Preset, 1080p, Performance Power Plan on DC Power, Test Date: 20/Oct/25

XeSS Super Resolution

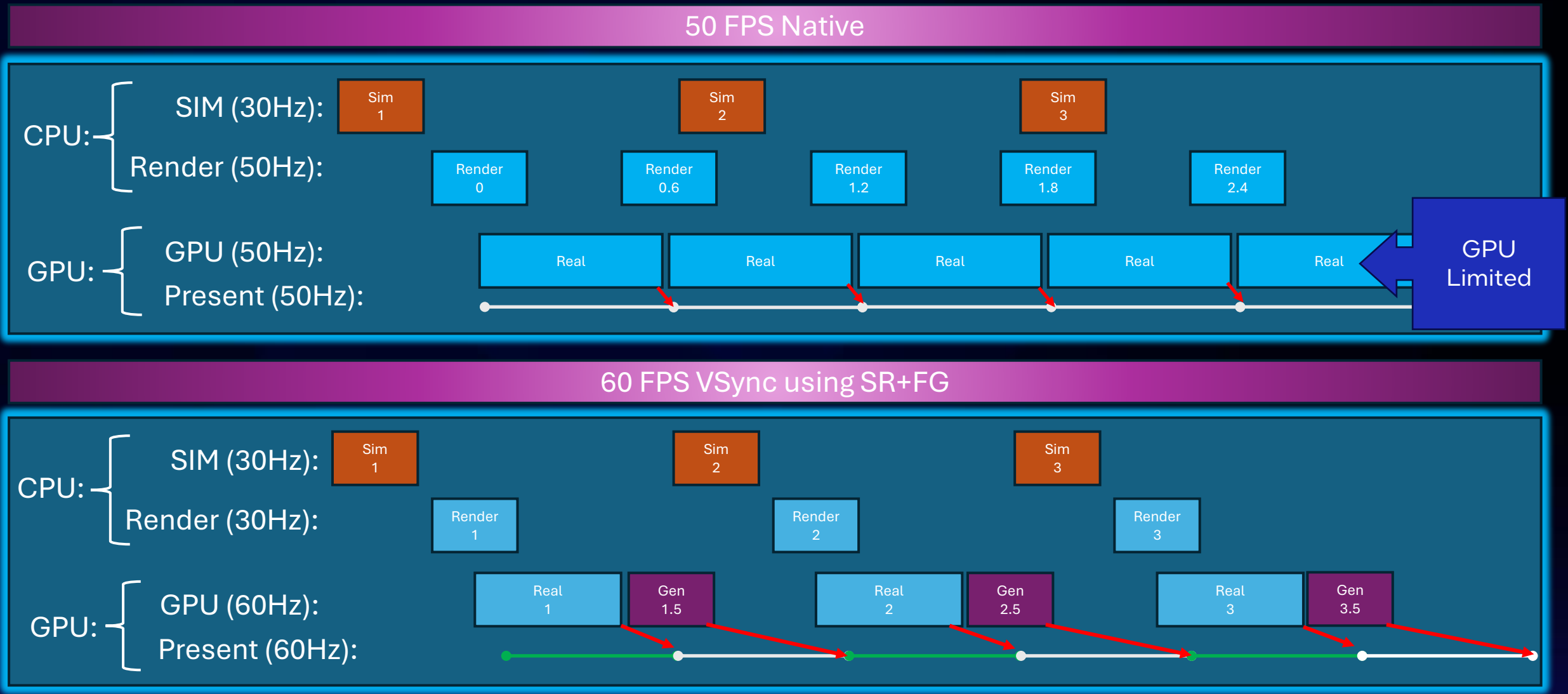
Native

XeSS SR Performance

XeSS SR Ultra Performance

*Captures taken on Lunar Lake MSI Claw 8 AI+ at High Preset, 1080p, with Performance Power Plan on DC Power

Battlefield 6 + XeSS SR+FG



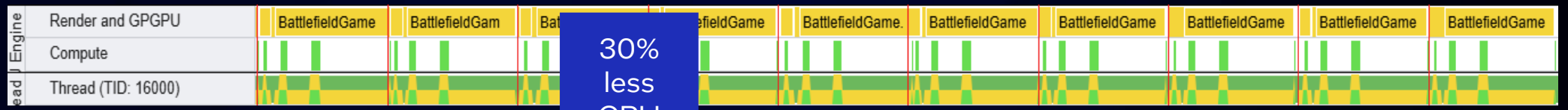
Visualization for illustrative purposes only

XeSS FG – CPU Workload

Total Instructions Retired over 10 V-Sync intervals at 60 Hz (10 displayed frames) — FG vs No-FG

FG Disabled

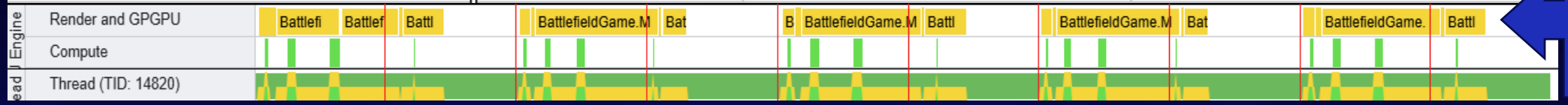
Core Type / Thread / Function / Call Stack	INST RETIRED.ANY	CPU CLK UNHALTED.THREAD	CPU CLK UNHALTED.REF TSC ▼
P-Core	2,083,562,000	1,307,592,000	1,152,398,000
E-Core	858,520,000	614,172,000	723,138,000



30% less CPU cycles

FG Enabled

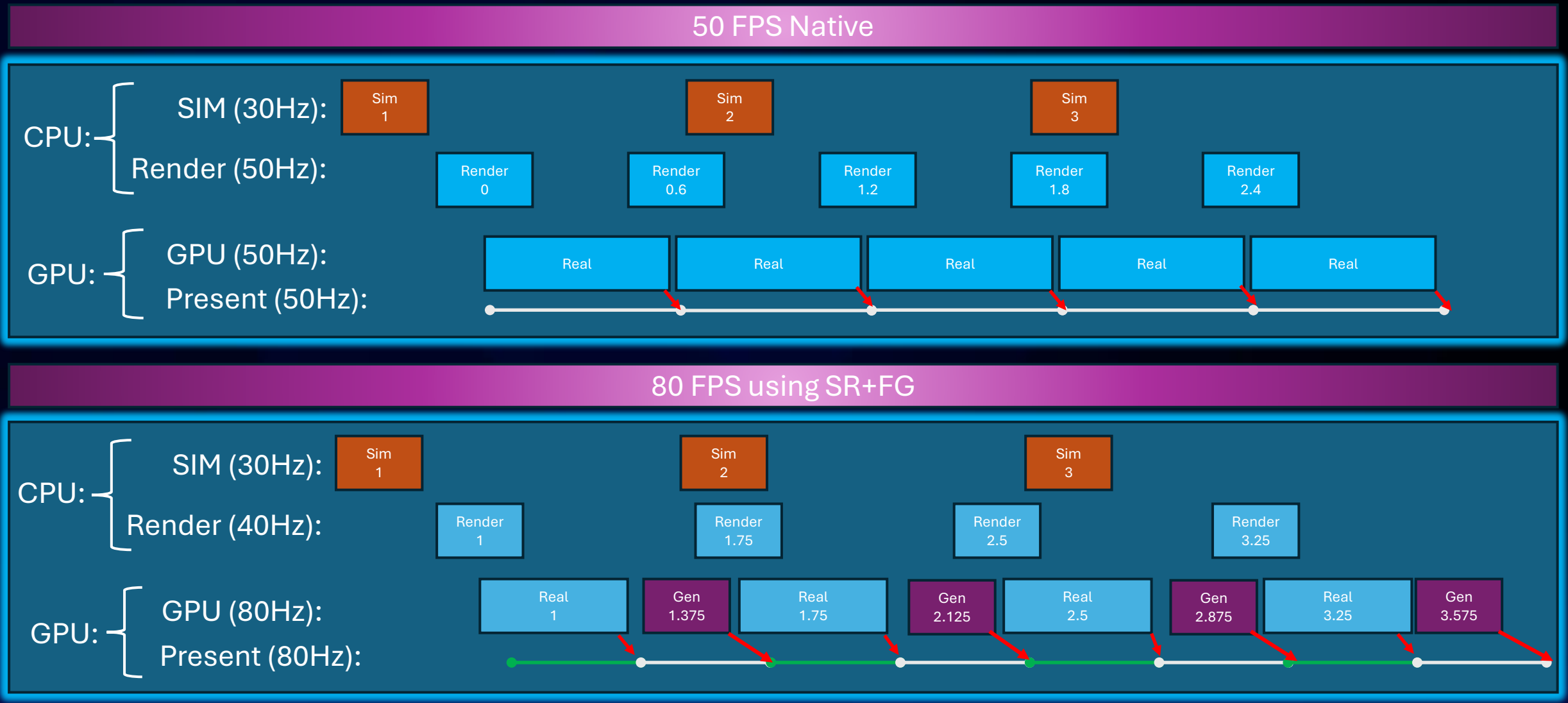
Core Type / Thread / Function / Call Stack	INST RETIRED.ANY	CPU CLK UNHALTED.THREAD	CPU CLK UNHALTED.REF TSC ▼
P-Core	1,162,304,000	871,728,000	759,460,000
E-Core	468,884,000	406,146,000	442,468,000



Increased GPU headroom

VTune Captures taken on Lunar Lake MSI Claw 8 AI+ at Medium Preset with XeSS SR Performance, Performance Power Plan on DC Power, Test Date: 10/22/25

Battlefield 6 + XeSS FG

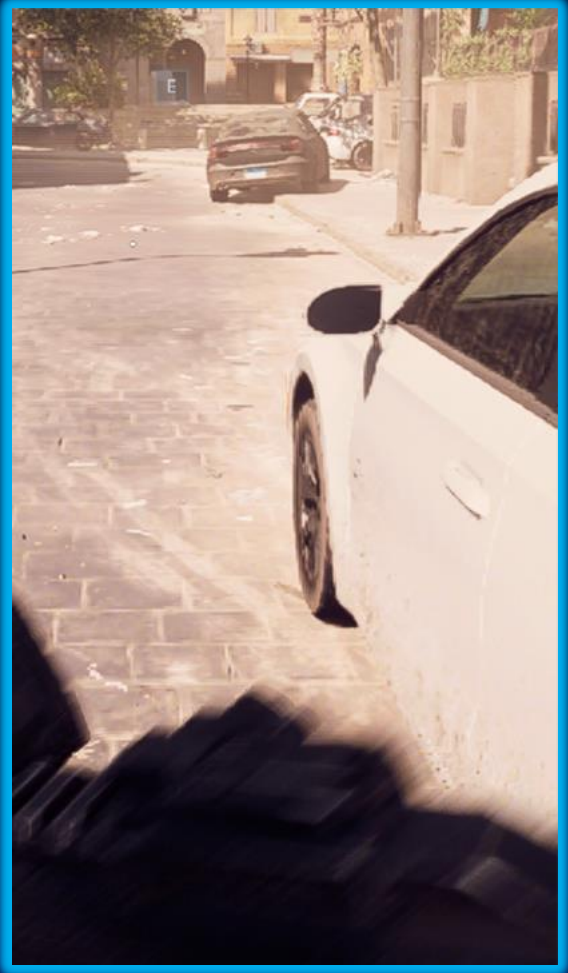


Visualization for illustrative purposes only

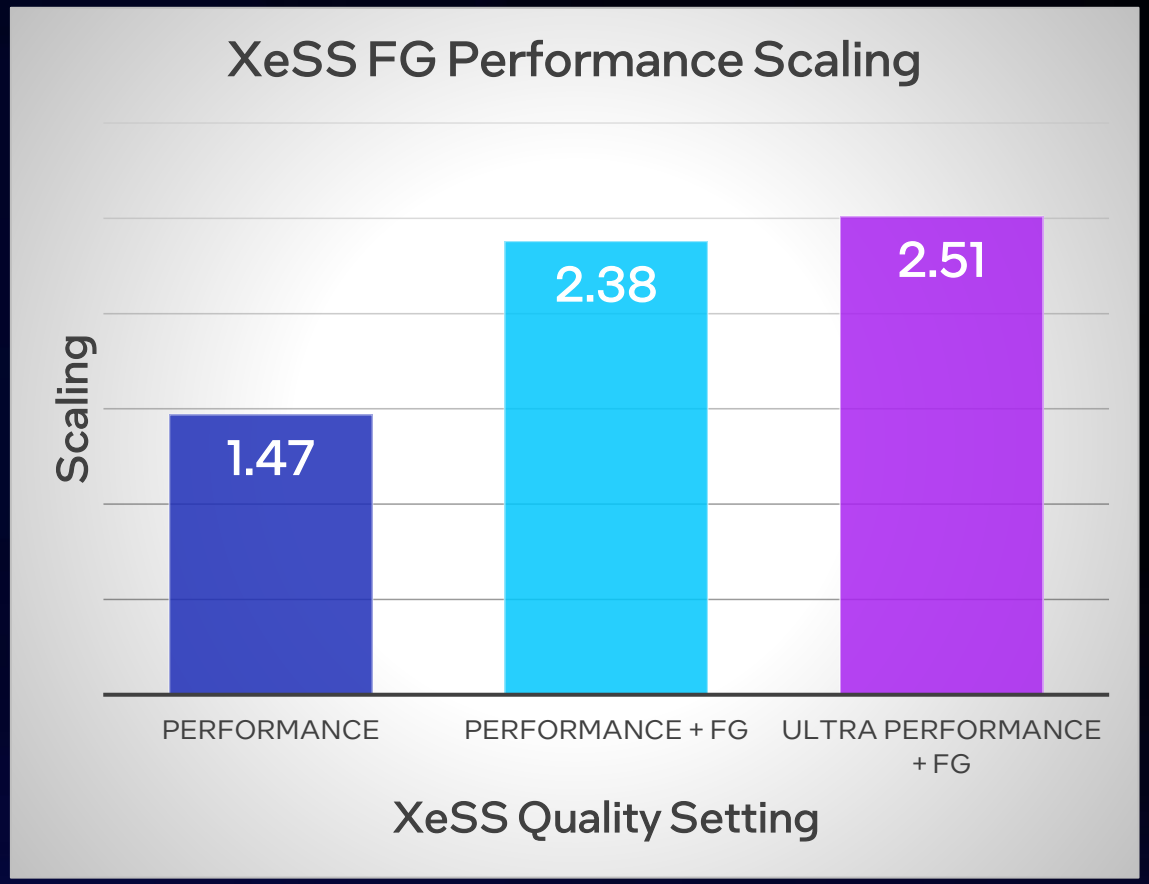
XeSS FG

XeSS SR Performance

SR + FG Generated Frame



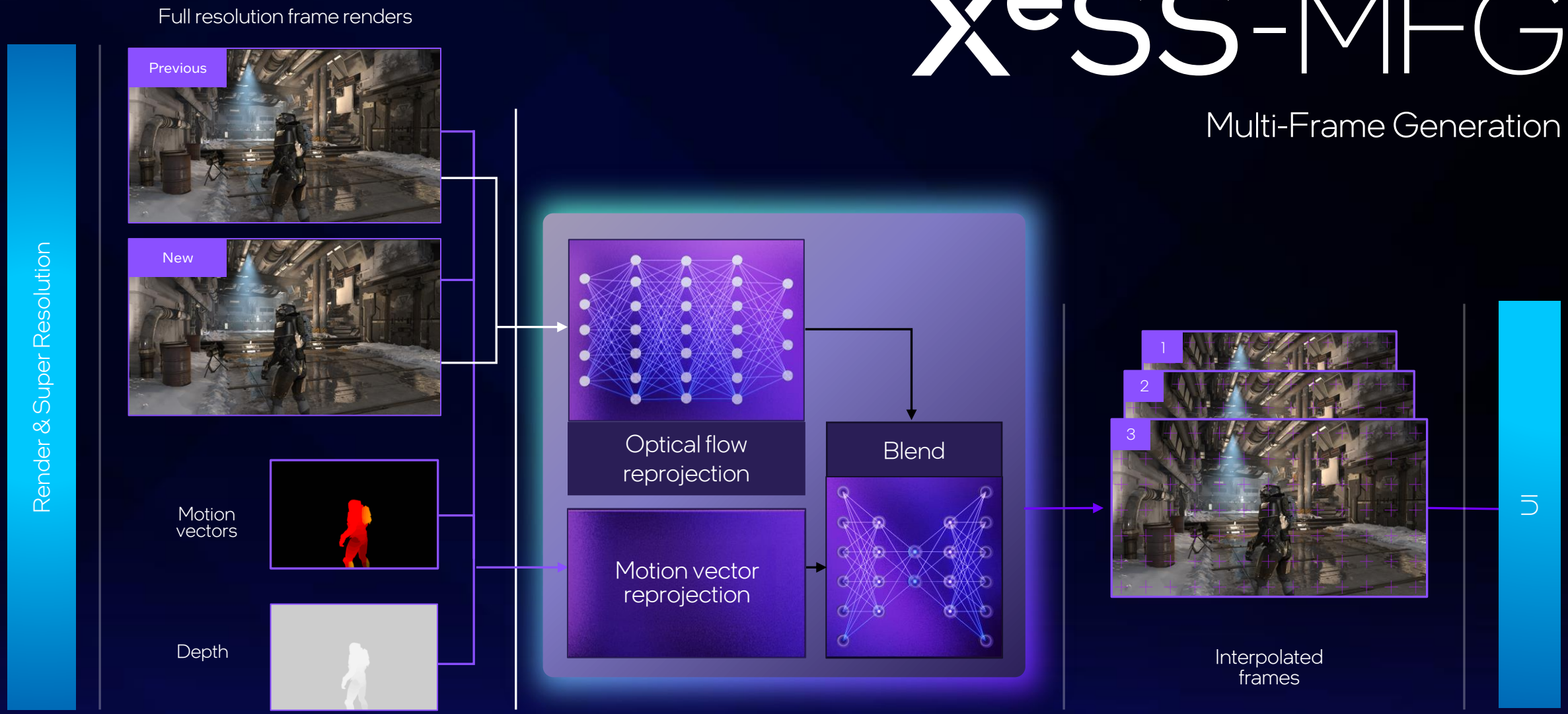
Captures taken on Lunar Lake MSI Claw 8 AI+ at High Preset with SR Performance to ensure 40 FPS minimum target for FG



PresentMon captures taken on Lunar Lake MSI Claw 8 AI+ at High Preset, XeSS SR Performance and Ultra Performance, with Performance Power Plan on DC Power, Test Date: 22/Oct/25

XeSS-MFG

Multi-Frame Generation



Visualization for illustrative purposes only

Conclusion

Key takeaways for developers

1	Run automated tests with gameplay context on target hardware.
2	On PC, even distribution of threaded work is key to good OS scheduling but minimize thread contention and **DON'T** pin threads.
3	Use CPU and GPU culling to remove redundant work from your frame.
4	Integrate Upscaling and Frame Gen early, AI hardware is increasing in quality and performance.

Questions

Why no Raytracing?

- Performance across all hardware tiers for all players
- Early adopter of new renderer, focused on content authoring and performance
- RT works against aggressive culling early in the frame

Workloads and Configurations – Xe IP Improvements

CLAIM	SYSTEM CONFIGURATION	MEASUREMENT	MEASUREMENT PERIOD
Xe2 Hardware support for ExecuteIndirect shows ~40% improvement in performance compared to Xe XMX cores improve performance on XeSS SR pass by ~38% compared to DP4a	MSI Claw 8 AI+ A2VM Motherboard: Micro-Star International Co., Ltd. MS-1T52 REV:1.0 BIOS Version: MSI_NB - 1072009 CPU: Intel(R) Core(TM) Ultra 7 258V VBS: Enabled Resizable BAR: NA Memory Model: Micron Technology Storage: 1x M.2 2230 SSD Win 11 Version: Microsoft Windows 11 Home Build 26100.6899 Defender: Enabled Vt-d: Enabled Power Plan: Balanced GPU: Intel(R) Arc(TM) 140V GPU (16GB) GFX Driver: 32.0.101.8136	All measurements and screenshots taken in Egypt Multiplayer map at Capture Point C PIX GPU Frame captures taken from same position and view Results are the median of 4 captures, measuring the total frametime spent running ExecuteIndirect and ExecuteIndirect draws in a single frame Quality Settings: XeSS SR Performance High Preset	Oct 20th 2025

Workloads and Configurations – XeSS SR Memory

CLAIM	SYSTEM CONFIGURATION	MEASUREMENT	MEASUREMENT PERIOD
XeSS SR Reduced Memory Traffic by ~47% and Decreased frame time by ~33%	MSI Claw 8 AI+ A2VM Motherboard: Micro-Star International Co., Ltd. MS-1T52 REV:1.0 BIOS Version: MSI_NB - 1072009 CPU: Intel(R) Core(TM) Ultra 7 258V VBS: Enabled Resizable BAR: NA Memory Model: Micron Technology Storage: 1x M.2 2230 SSD Win 11 Version: Microsoft Windows 11 Home Build 26100.6899 Defender: Enabled Vt-d: Enabled Power Plan: Balanced GPU: Intel(R) Arc(TM) 140V GPU (16GB) GFX Driver: 32.0.101.8136	All measurements taken in Egypt Multiplayer map at Capture Point C PIX GPU Frame captures taken from same position and view Results are the median of 4 captures, measuring the total Bytes Read and Bytes Write counters for the full frame Quality Settings: XeSS SR Performance and Native High Preset	Oct 20th 2025

Workloads and Configurations – XeSS FG CPU Workload

CLAIM	SYSTEM CONFIGURATION	MEASUREMENT	MEASUREMENT PERIOD
XeSS FG Reduced CPU workload	MSI Claw 8 AI+ A2VM Motherboard: Micro-Star International Co., Ltd. MS-1T52 REV:1.0 BIOS Version: MSI_NB - 1072009 CPU: Intel(R) Core(TM) Ultra 7 258V VBS: Enabled Resizable BAR: NA Memory Model: Micron Technology Storage: 1x M.2 2230 SSD Win 11 Version: Microsoft Windows 11 Home Build 26100.6899 Defender: Enabled Vt-d: Enabled Power Plan: Balanced GPU: Intel(R) Arc(TM) 140V GPU (16GB) GFX Driver: 32.0.101.8136	All measurements taken in Egypt Multiplayer map at Capture Point C VTune captures taken over 30 seconds at same position and viewpoint Results are the Instructions retired count filtered over a 10 frame interval with VSync enabled at 60hz Quality Settings: XeSS SR Performance Medium Preset XeSS FG Disabled vs Enabled	Oct 22th 2025

Workloads and Configurations – XeSS FG Performance

CLAIM	SYSTEM CONFIGURATION	MEASUREMENT	MEASUREMENT PERIOD
XeSS FG Performance scaling	MSI Claw 8 AI+ A2VM Motherboard: Micro-Star International Co., Ltd. MS-1T52 REV:1.0 BIOS Version: MSI_NB - 1072009 CPU: Intel(R) Core(TM) Ultra 7 258V VBS: Enabled Resizable BAR: NA Memory Model: Micron Technology Storage: 1x M.2 2230 SSD Win 11 Version: Microsoft Windows 11 Home Build 26100.6899 Defender: Enabled Vt-d: Enabled Power Plan: Balanced GPU: Intel(R) Arc(TM) 140V GPU (16GB) GFX Driver: 32.0.101.8136	All measurements taken in Egypt Multiplayer map at Capture Point C PresentMon captures taken over a 120 second period Results are the Median of 3 runs Quality Settings: XeSS SR Performance and Ultra Performance High Preset XeSS FG Disabled vs Enabled	Oct 22th 2025

Intel® Core Processor Windows Scheduling/Parking

Power Management Parking Settings

Varies by power plan:

CPMinCores: Specifies the minimum percentage of processors that can be unparked state at any given time.

CPMaxCores: Specifies the maximum percentage of processors that can be unparked state at any given time.

CPIncreaseTime: The minimum elapsed time before additional processors can be transitioned from the parked to unparked state.

CPDecreaseTime: The minimum elapsed before additional processors can be transitioned from unparked to parked state.

CPHeadroom: Specifies the additional utilization that would cause the core parking engine to unpark an additional processor

Appendix

Claim	Claim Details/Citation
IPU 7.5	IPU 7.5 is available on select systems. OEM enablement required. Check with OEM or retailer for system configuration. See details see from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex
>50% more performance vs. Lunar Lake & Arrow Lake	Testing by Intel as of Sept 2025. Data based on Panther Lake reference validation platform measurement vs Lunar Lake and Arrow Lake-H reference validation platforms as measured by 3Dmark Solar Bay, Cyberpunk 2077 and Borderlands 3. See details see from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex
>30% lower power at similar MT performance vs. Arrow Lake	Testing by Intel as of September 2025. Data based on Panther Lake reference validation platform measurement vs Lunar Lake & Arrow Lake reference validation platform(s) as estimated by SPECrate®2017_int_base (n-copy) for long-term expected steady state processor power consumption. . See details see from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex
Up to 10% lower power vs. Lunar Lake	Testing by Intel as of September 2025. Data based on Panther Lake reference validation platform measurement vs Lunar Lake & Arrow Lake reference validation platform(s) as estimated by SPECrate®2017_int_base (1-copy) for long-term expected steady state processor power consumption. See details see from Intel Technology Tour (ITT) 2025, available at intel.com/performanceindex
Gives more headroom for larger GPU configurations	Testing by Intel as of September 2025. Data based on Panther Lake reference validation platform measurement vs Lunar Lake & Arrow Lake reference validation platform(s) as estimated using pre-release build of Battlefield 6, setting used were Medium Presets at 1080P with performance upscaling preset. Power Metrics measured using Software tools.
Power Benefits of Multi-threading	Testing by Intel as of September 2025. Data based on Panther Lake reference validation platform measurement vs Lunar Lake & Arrow Lake reference validation platform(s) as estimated using pre-release build of Battlefield 6, setting used were Medium Presets at 1080P with performance upscaling preset. Power Metrics measured using Software tools.
Reduced average CPU frequency from up to 3.5Ghz to 1.3Ghz when running Battlefield 6	Testing by Intel as of September 2025. Data based on Panther Lake reference validation platform measurement vs Lunar Lake & Arrow Lake reference validation platform(s) as estimated using pre-release build of Battlefield 6, setting used were Medium Presets at 1080P with performance upscaling preset.
Relative Core IPC Normalised to Lunar Lake P-Core	Testing by Intel as of September 2025. Data based on Panther Lake reference validation platform measurement vs Lunar Lake & Arrow Lake reference validation platform(s) as estimated using pre-release build of Battlefield 6 , setting used were Medium Presets at 1080P with performance upscaling preset. IPC measured using Intel® VTune™ Profiler 2025.3.
XeSS Super Resolution – Memory	*PIX Captures taken on Lunar Lake MSI Claw 8 AI+ at High Preset, 1080p, Performance Power Plan on DC Power, Test Date: 20/Oct/25
XeSS FG – CPU Workload	VTune Captures taken on Lunar Lake MSI Claw 8 AI+ at Medium Preset with XeSS SR Performance, Performance Power Plan on DC Power, Test Date: 10/22/25
XeSS FG	PresentMon captures taken on Lunar Lake MSI Claw 8 AI+ at High Preset, XeSS SR Performance and Ultra Performance, with Performance Power Plan on DC Power, Test Date: 22/Oct/25