



VRS Tier 1 with DirectX 12 From Theory To Practice

Marissa du Bois - Intel, John Gibson - Tripwire



@IntelSoftware @IntelGraphics



Legal Notices and Disclaimers

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at [intel.com].

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.

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

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

Intel, Core and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

Notices & Disclaimers

- Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade. Results have been estimated or simulated.
- Your costs and results may vary.
- Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.
- © Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

© Intel Corporation.

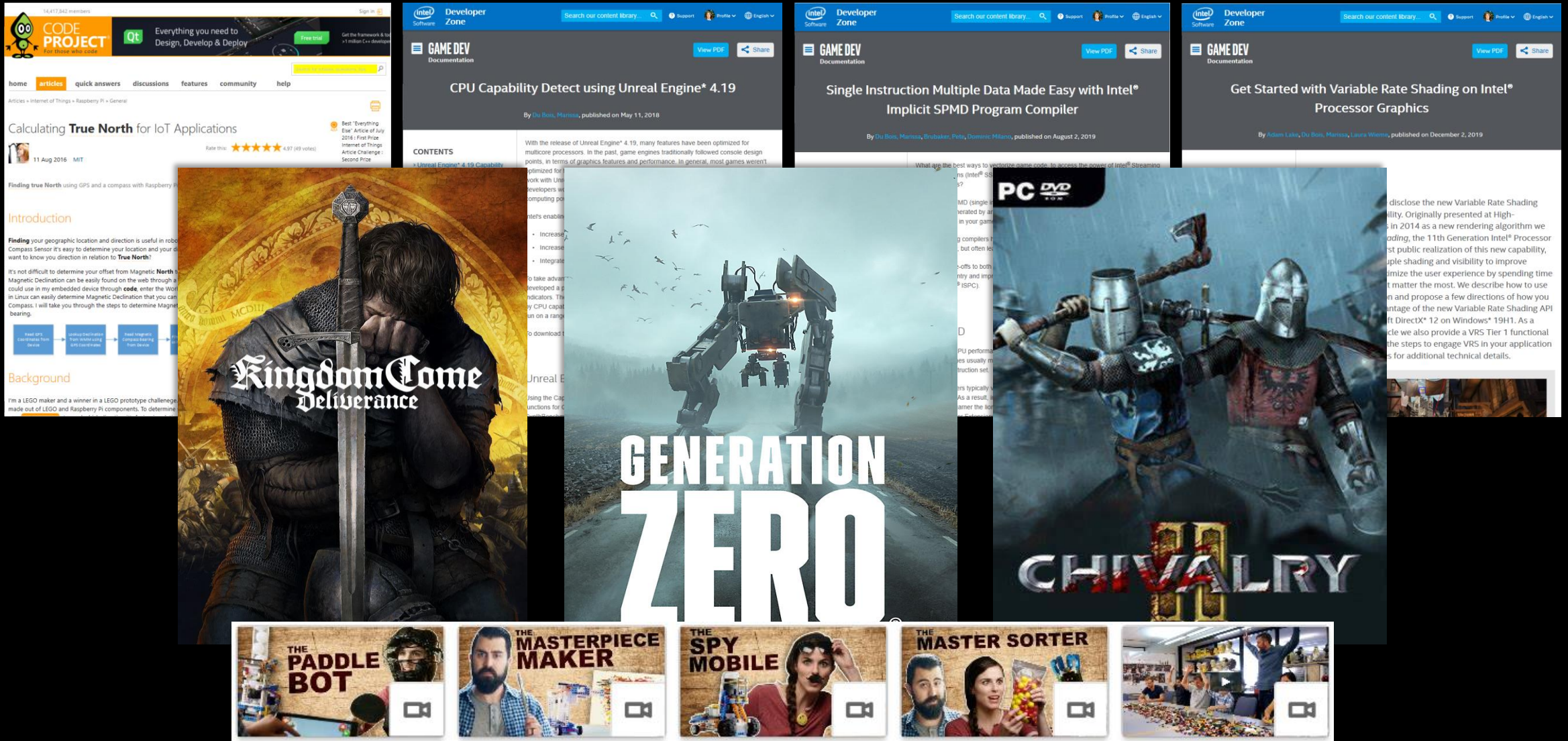


Today's Agenda

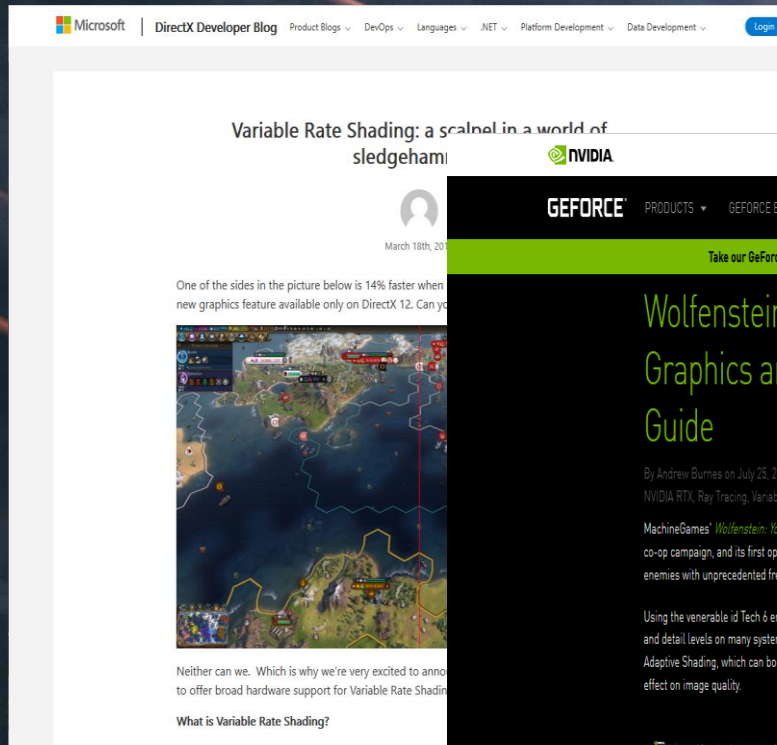
- Intro to Variable Rate Shading Tier 1
- VRS Tier 1 in Unreal Engine 4
- VRS In Practice with Chivalry II
- Question & Answer



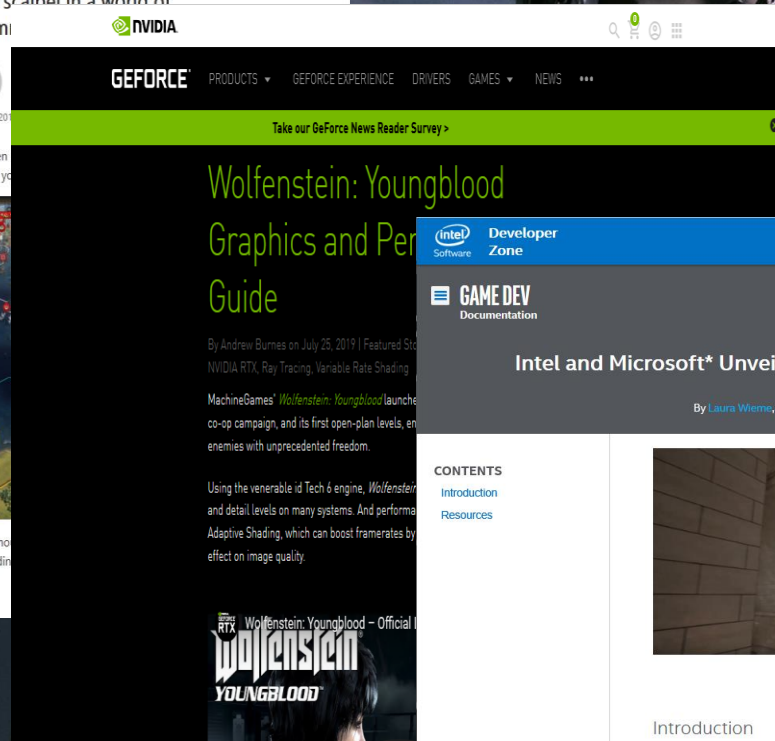
Marissa du Bois – Intel Graphics Software Engineer



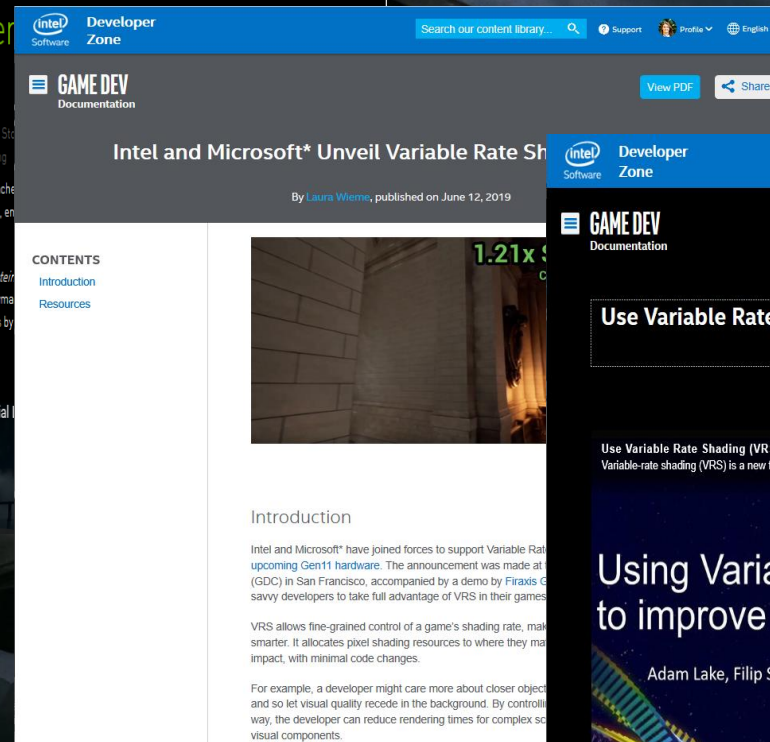
Prior Art



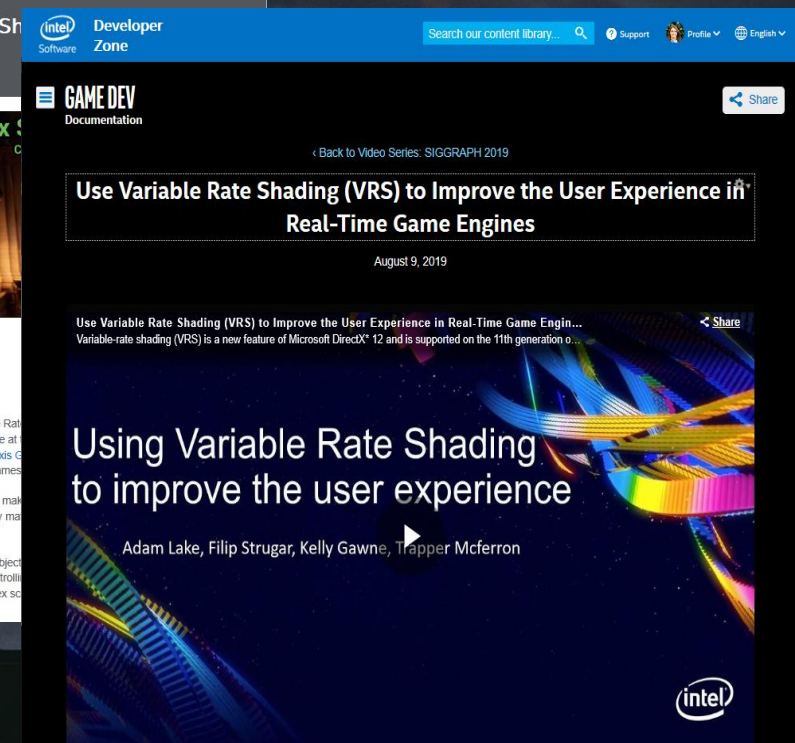
[Van Ryhn 2019]



[Burnes 2019]



[Wieme (Reznikov), 2019]



[Lake et al. 2019]





Introduction to Variable Rate Shading Tier 1

Marissa du Bois – Intel Graphics Software Engineer

Thanks to Adam Lake, Filip Strugar, Kelly Gawne, Laura Reznikov, Adam Kunka, and many others.



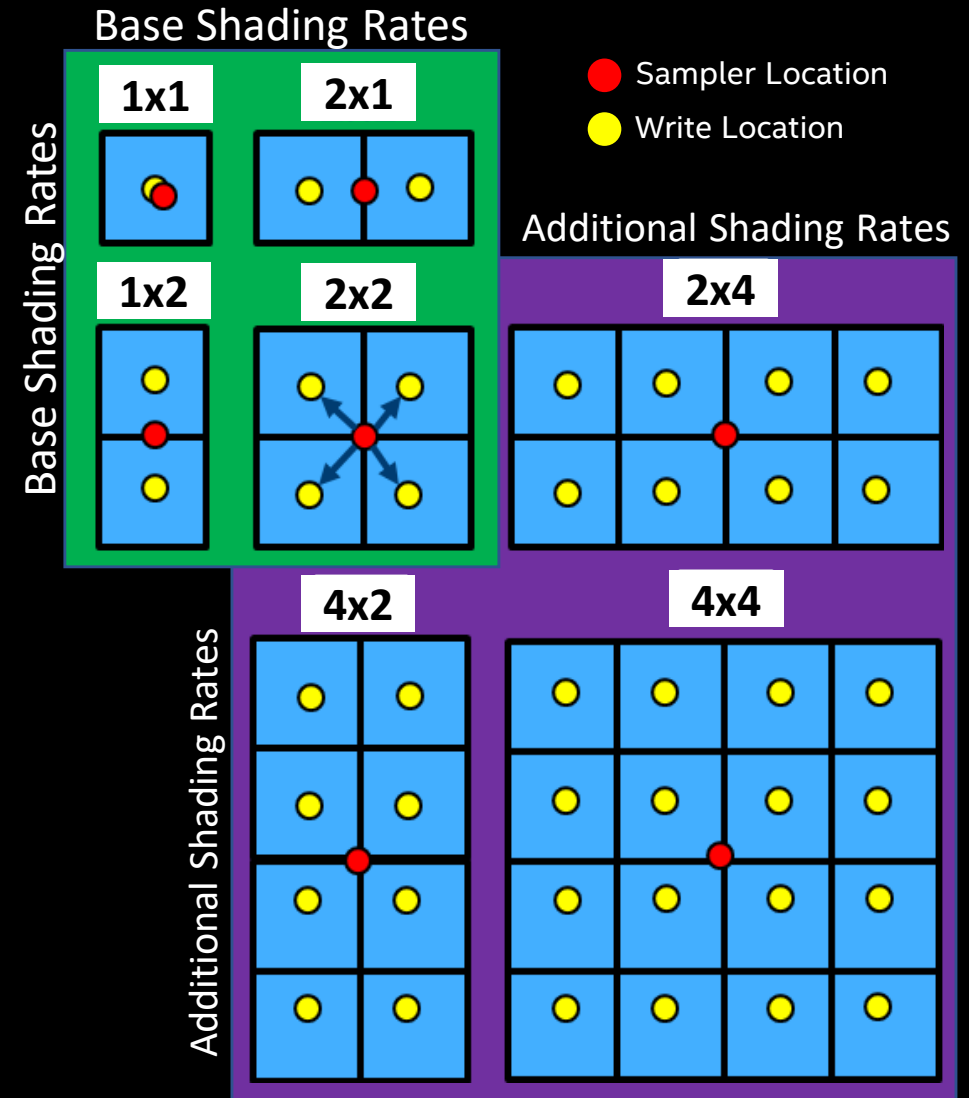
@IntelSoftware @IntelGraphics



VRS Basics

Smartly control rasterization by varying shading rates.

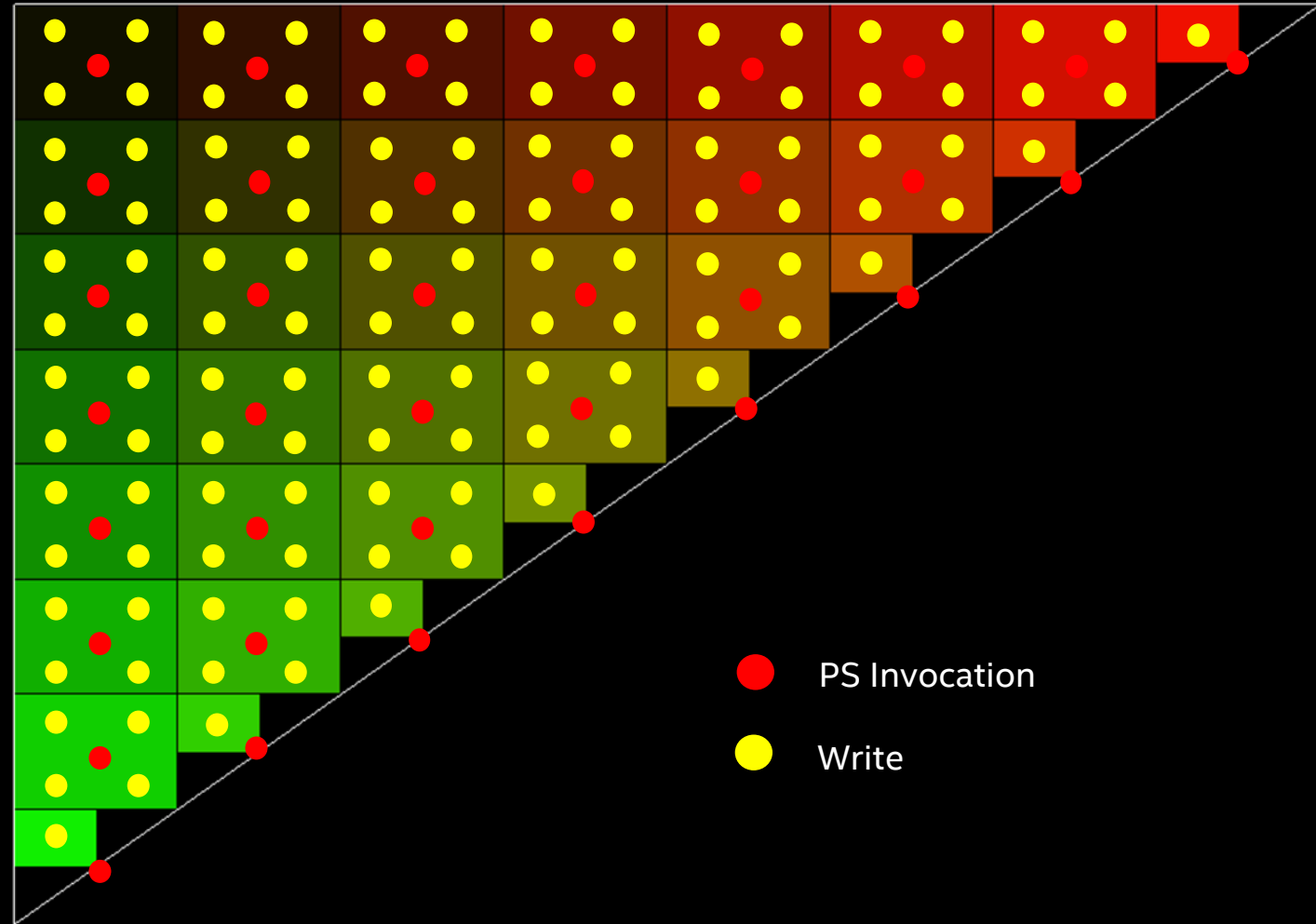
- *Variable Rate Shading Tier 1* is a graphics feature introduced with Intel Ice-Lake Architecture and launched in 2019.
- *Shading Rates* can be set through *Microsoft's* DirectX* 12 API* and are supported by Intel's Graphics driver.
- *1x1, 1x2, 2x1, 2x2* are the minimum supported shading rates in the VRS Tier 1 specification.
- Intel Ice-Lake Architecture also supports *2x4, 4x2, and 4x4* which considered additional shading rates and require a conditional test to determine support.



Pixel Shader Invocations

PS Invocations Explained

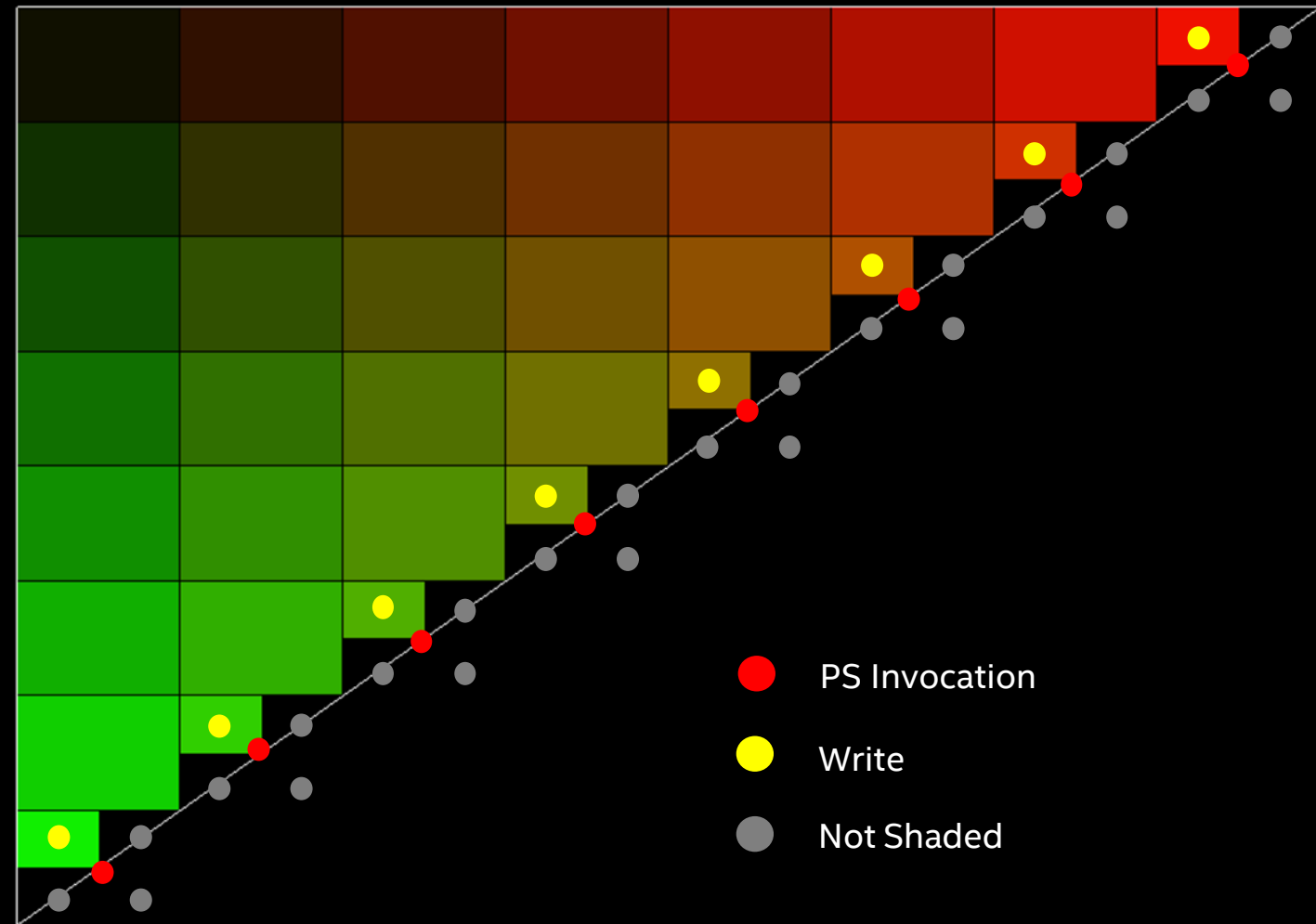
- 16x16 Triangle Rendered at 2x2
- VRS Reduces Pixel Shader Invocations and write a single value to multiple pixels.
- Pixel Shader reduction is *Approximately Proportional* to the VRS shading rate.
- VRS 1x1 - 120 Pixels Covered with 120 PSInvocations
- *Versus*
- VRS 2x2 - 120 Pixels Covered with 36 PSInvocations
- *Versus*
- VRS 4x4 - 120 Pixels Covered with 10 PSInvocations



VRS Edge Preservation

Edge Preservation Explained

- 16x16 Triangle Rendered at 2x2
- 28 2x2 coarse pixels + 8 1x1 pixels = 120 Pixels
- Edge Pixels are sampled at center of coarse pixel
- Pixels outside of coverage area are not shaded
- Edge Preservation is one *advantage* over Resolution Scaling



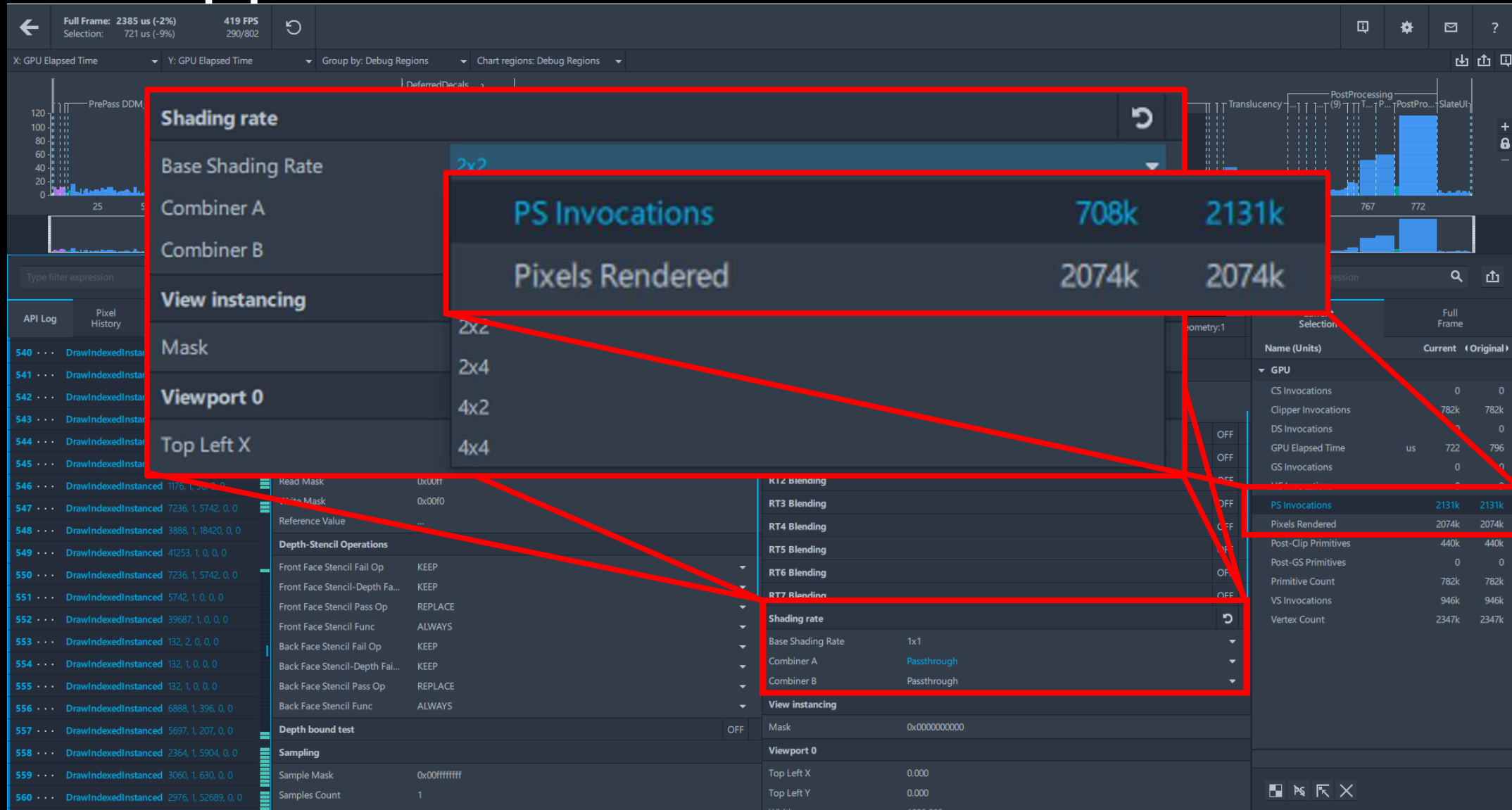
Tier 1 Limitations

Things to watch out for...

- If **SV_Coverage** is declared as a shader input or output for VRS Tier 1 then the shading rate is reduced to 1x1.
- **SampleMask** is *required* to be a *full mask*. If **SampleMask** is configured to be something else, the shading rate will be reduced to 1x1.
- **EvaluateAttributeAt[Centroid/Sample/Snapped]** are not compatible with tier 1 VRS. If these intrinsic functions are used the shading rate is reduced to 1x1.
- HLSL **sample** keyword referenced in shader will reduce shading rate to 1x1.



VRS Support in GPA





VRS Tier 1 in Unreal Engine 4

Marissa du Bois – Intel Graphics Software Engineer

Thanks to Adam Lake, Kelly Gawne, Kai Wang, Jessica Chen, Filip Strugar, and Jeff Rous.



@IntelSoftware @IntelGraphics



Unreal Engine 4

Integration for Deferred and Forwarding Shading.

- *DirectX 12* RHI Integration.
- *RHICheckVRSSupport* determines hardware support and is called at initialization time.
- *RHISetVRSValues* takes a control structure to pass shading rate values into the Command List using the `D3D12CommandList5`
- *Console Variable* implementation for *Mesh-Pass* shading rates.
- *Editor Integration* for *Per-Material* shading rates.



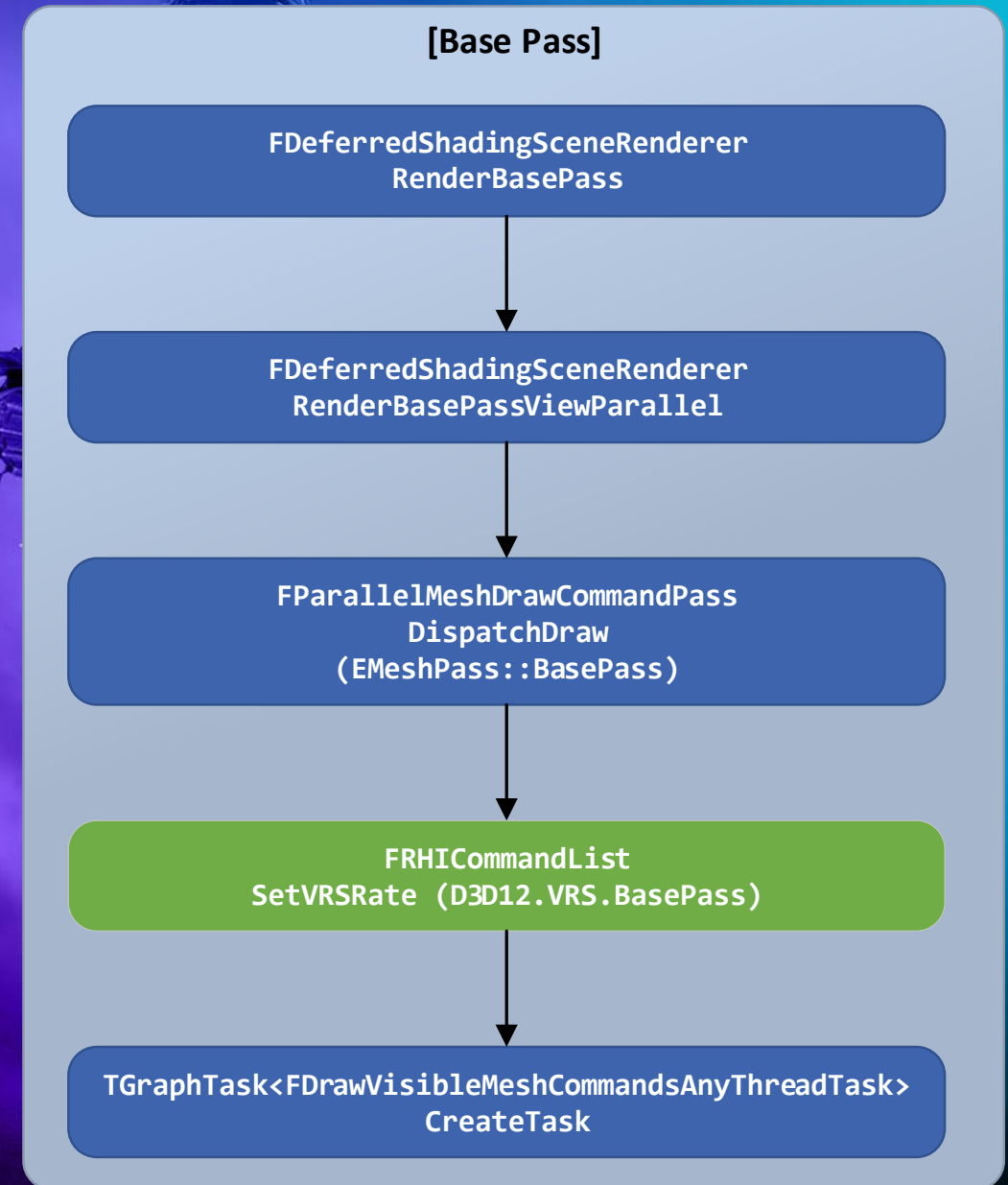
VRS Mesh Passes (Cont.)

Controlling Shading Rates during G-Buffer laydown.

- Multiple mesh passes defined by *EMeshPass* enumeration for g-buffer laydown.

BasePass
TranslucencyStandard
TranslucencyAfterDOF
TranslucencyAll

- Offers *performance boost* for *pixel-bound workloads*.
- BasePass* and *Translucency Passes* were the most artifact free passes. Experiments with other passes did not achieve acceptable quality.
- Mesh Passes have higher visual quality than equivalent render scale due to edge preservation but performance will be content dependent.



VRS Mesh-Passes

Doesn't Resolution Scaling give a similar benefit?

- *Resolution Scaling* is a common technique used to smooth framerate on lower power graphics parts.
- Resolution Scaling impacts the quality of the whole frame during *multiple pipeline stages*.
- Controlling Mesh Pass Shading Rates gives you *more control* over where you're trading quality for performance.
- Resolution Scaling can give higher performance but Mesh-Pass Shading Rates *preserves triangle-edges* while Resolution Scaling does not.
- Mesh Pass shading rates can be used with DRR for *added performance boost*.



SSR + VRS Experiment

- Applying *VRS* to *SSR* causes visual artifacts. But can we improve visual quality?
- Diagonal artifacts noted top-right flicker when *TAA* is enabled.
- Setting *r.SSR.Temporal* to 1 and scaling down *r.TemporalAAFilterSize* by Shading Rate X / Shading Rate Y can reduce flicker but increases sharpness.
- In *ScreenSpaceReflections.usf* scaling the *StepOffset* value in *ScreenSpaceReflectionsPS* by the shading rate can mitigate visual corruption but edge artifacts still remain.

```
// Adjust offset by VRS rate 1x1 = 1.0, 1x2/1x2 = 2, 2x2 = 4, 2x4/4x2 = 8, 4x4 = 16  
  
// Compute Shading Rate in HLSL by taking the derivative of SVPosition.x and SVPosition.y  
float2 shadingRate = float2(ddx(SvPosition.x), ddy(SvPosition.y));  
  
// Scale Ray-March Offset based on Shading Rate x and y  
StepOffset /= shadingRate.x * shadingRate.y;
```

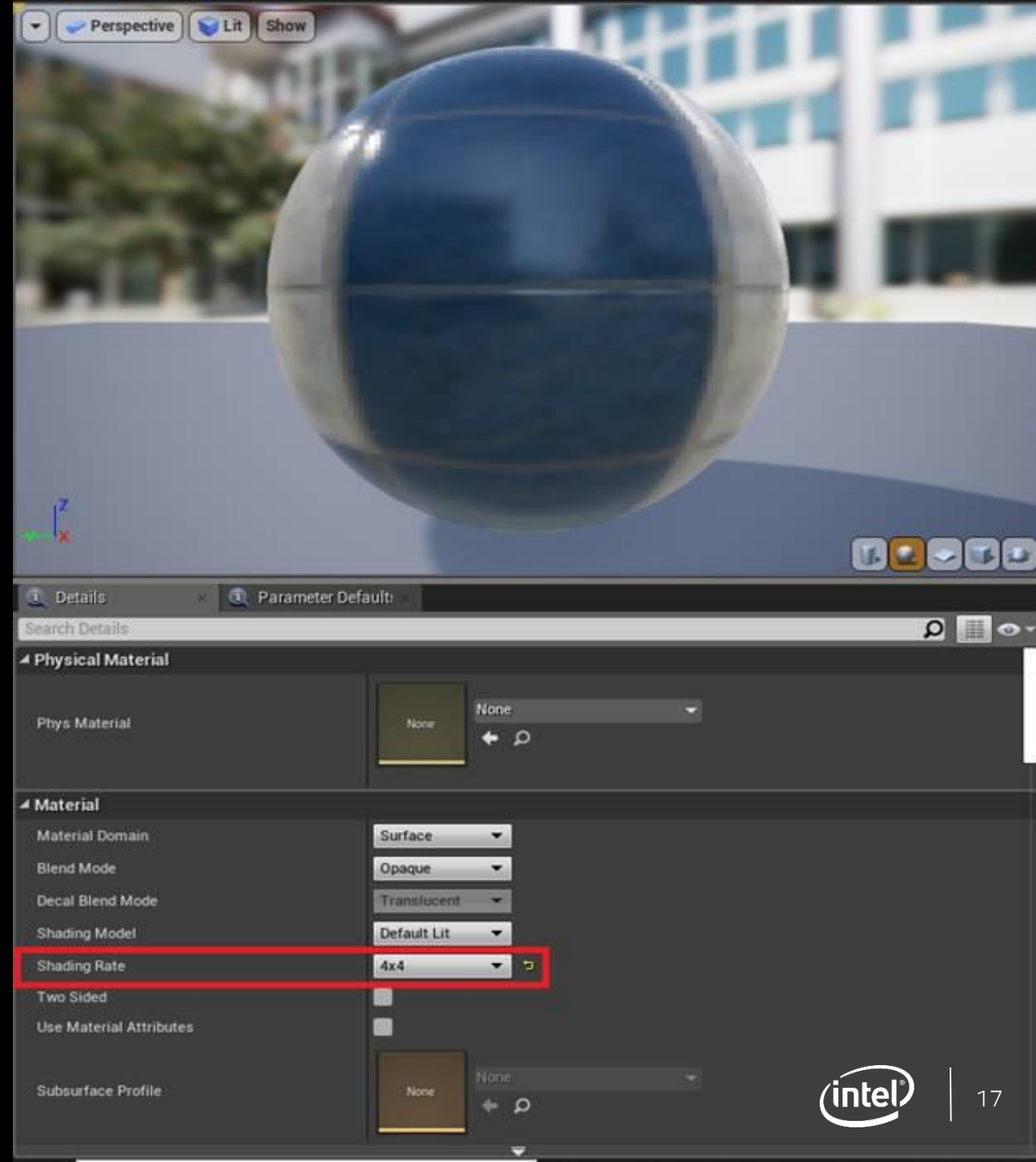
- I wouldn't recommend this for a final product but it points to a future where *SSR + VRS* could be feasible.



VRS Materials

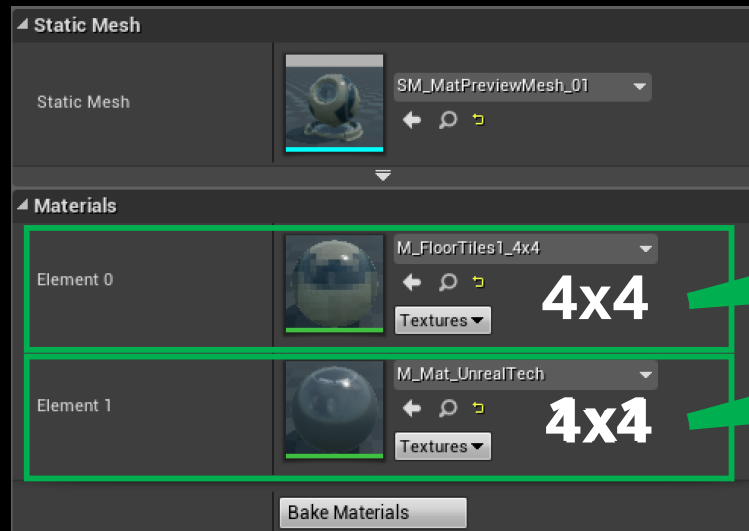
Sub-Pass Material System

- *New Material Property* - Shading Rate.
- *Material Instance* support with *Instance Overrides*.
- *Live Updates* in Material Viewport.
- Apply Shading Rates to multiple assets that share materials.
- Create Instances to smartly control shading rates for a subset of meshes.
 - Use highest Shading Rate 1x1 to preserve high-quality assets.
 - Use a Lower Shading Rate like 2x2 or 4x4 for less important assets.
- **Mix & Match** material shading rates to selectively retain quality on a single mesh.

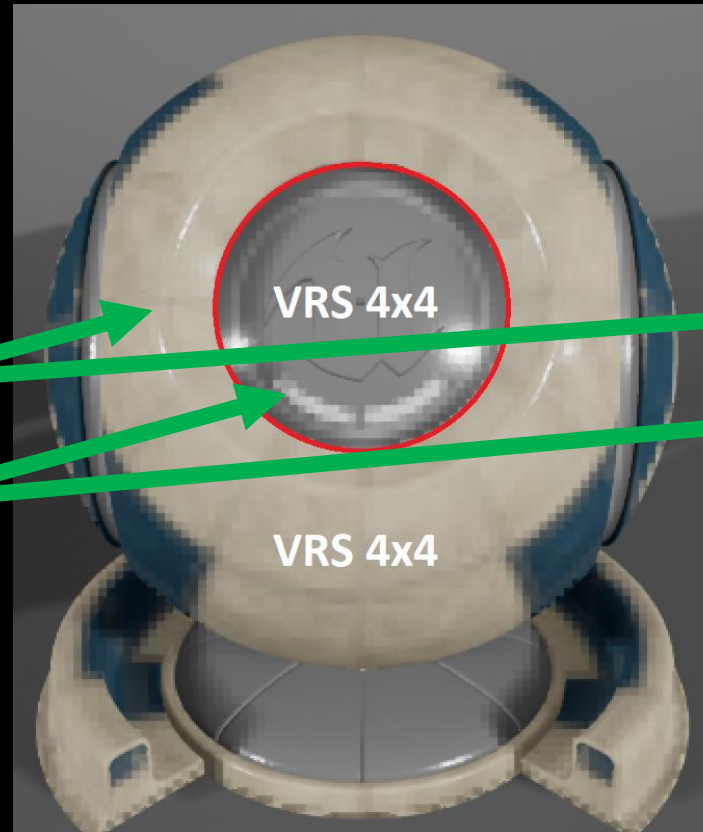


Mix & Match Material Shading Rates

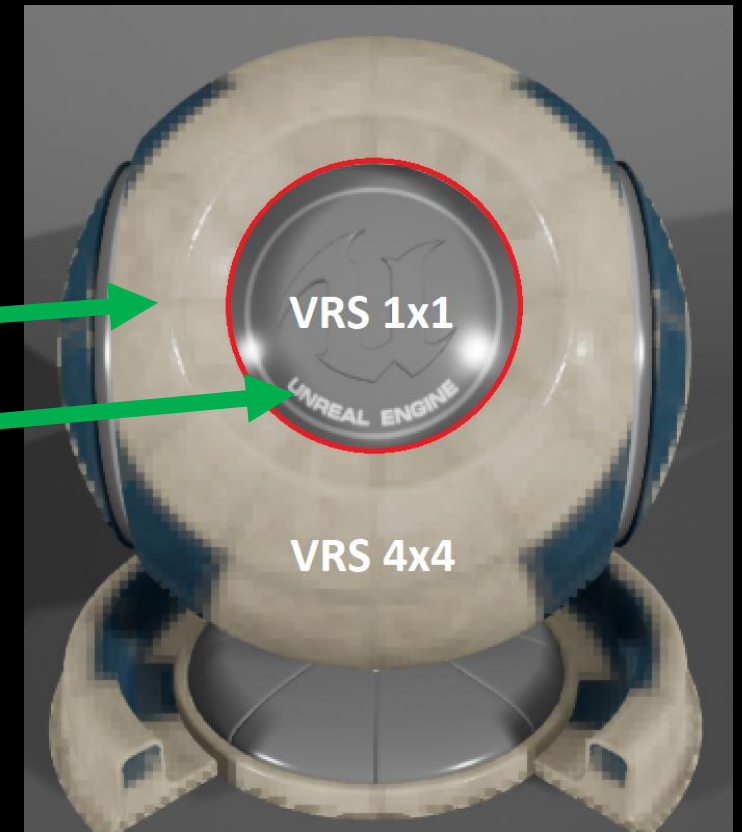
Use two Materials to selectively Vary Shading Rates on a Per-Model basis.



Using 2x2 or 4x4 in some cases may result in poor visual quality.



By Mixing Shading Rates we can preserve high-fidelity content.

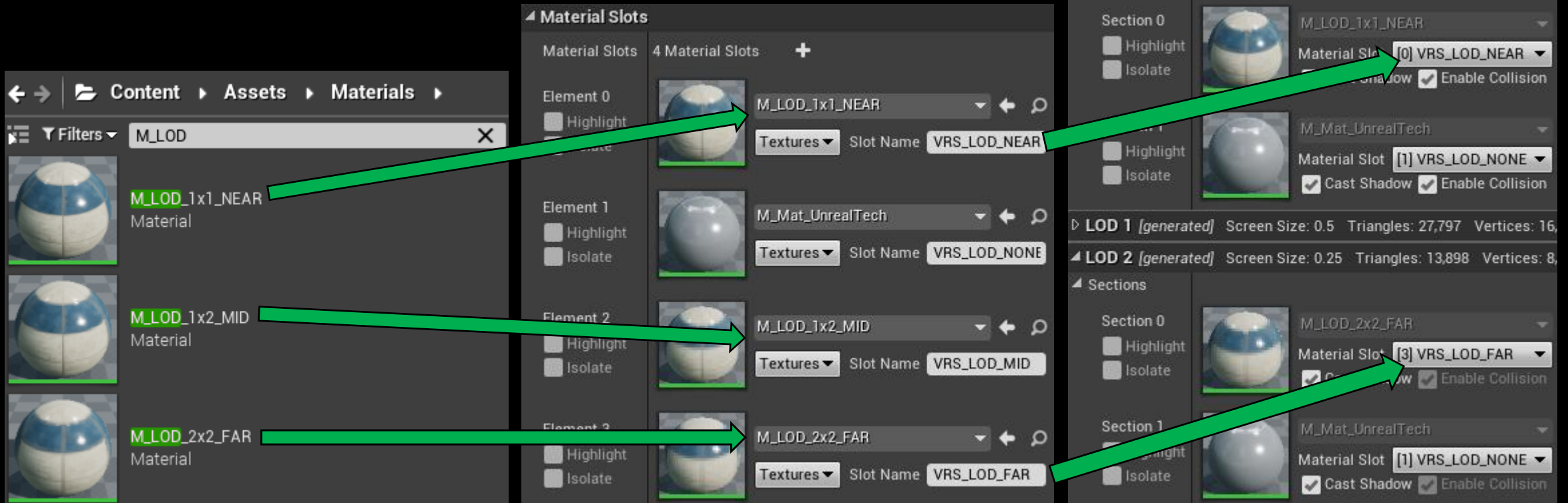


VRS Level of Detail

Create VRS materials for each LOD level, we define NEAR (1x1), MID (1x2), FAR (2x2).

In the material editor insert the LOD materials into material slots.

Use the material slots to apply VRS materials to custom LODs.

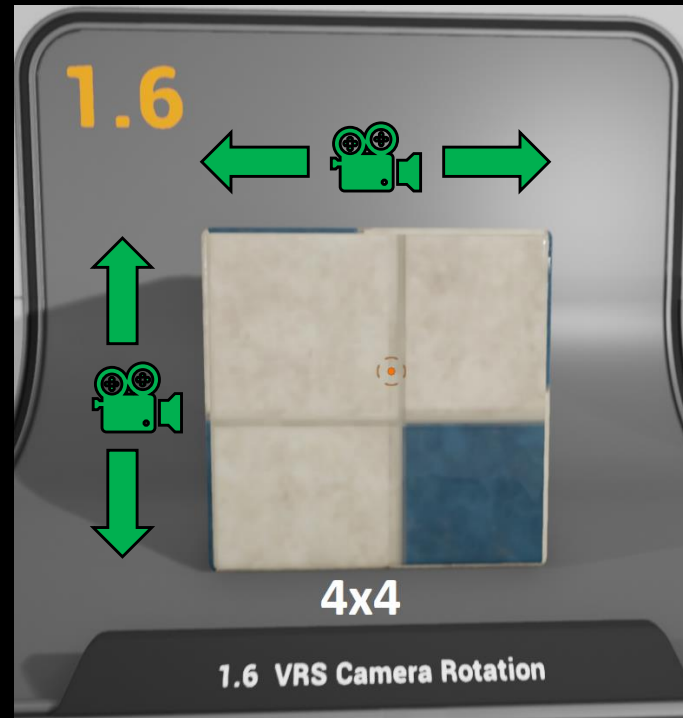


VRS Velocity & Camera Rotation

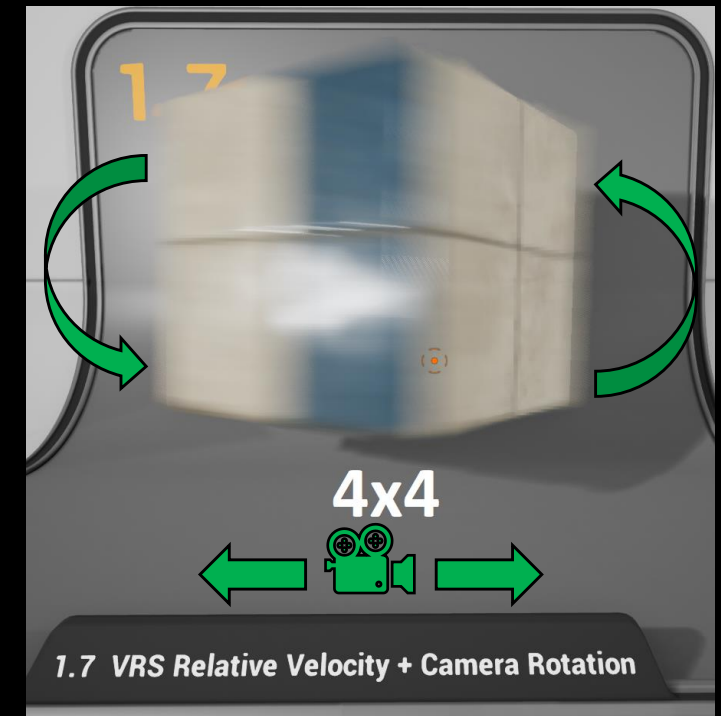
When an object is under motion we can apply a material with a lower shading rate.



Similarly, when the camera is under rotation we can alter the shading rate of the material.



We can combine both motion & rotation to dynamically apply material shading rates.



VRS Volumes & Particles

Use bounding volumes to set VRS materials when models overlap.



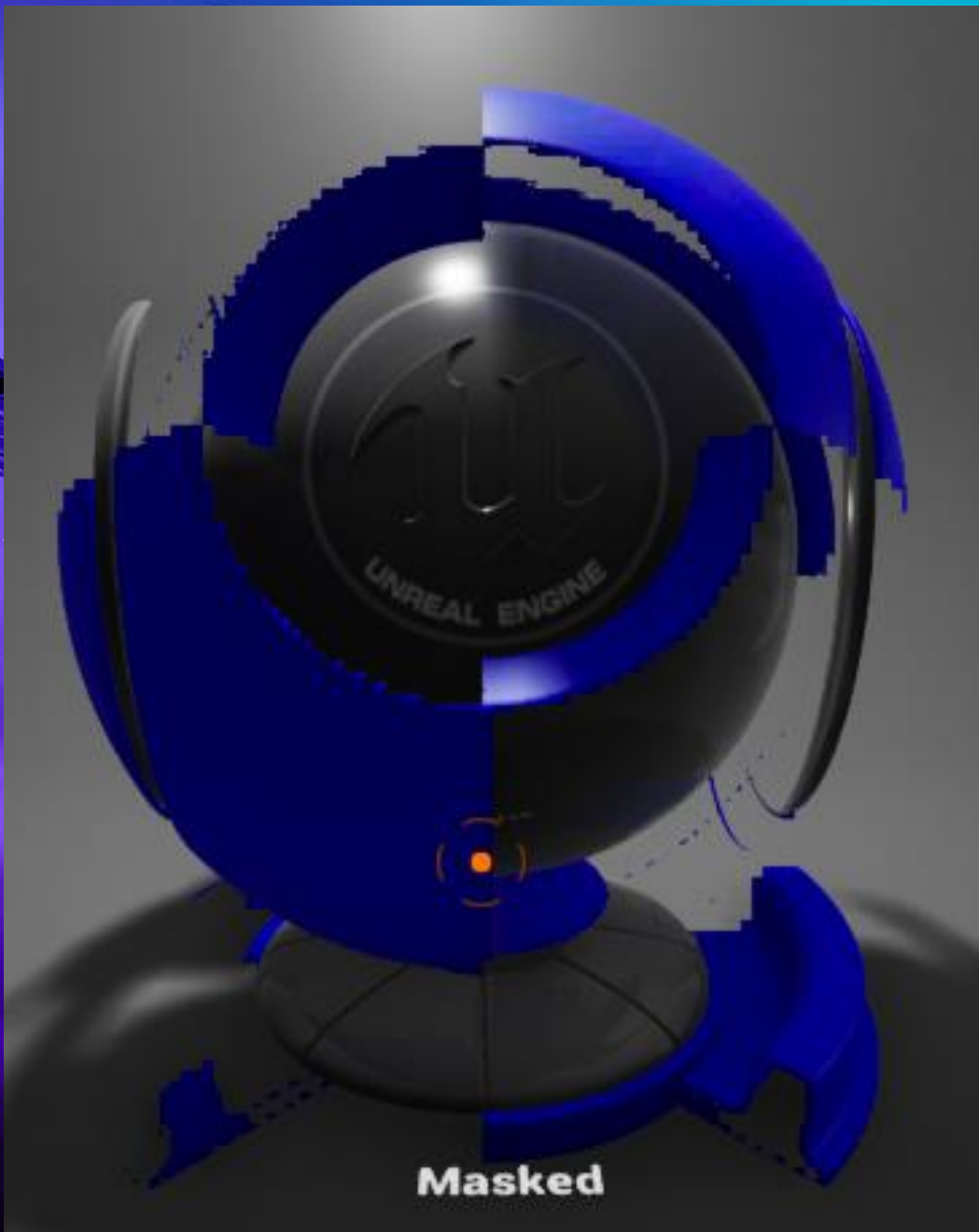
We can add a particle system to the bounding volume to obscure shading rate changes.



Material Limitations

Everything in Moderation...

- *Overusing VRS materials can result in poor performance.*
- *Minimize shading rate changes to avoid API performance penalty and **partial pipeline flushes** on Ice-Lake.*
- *Duplicate Shading Rates are pruned in the driver but **switching rates still has overhead**.*
- *We can **statically cache shading rates** in the renderer to reduce overhead; but too many material permutations can result in poor performance.*
- *Materials with **Opacity Masks** do not persevere edges in transparent regions; Consider Translucencies instead.*



CHIVALRY II

VRS In Practice with Chivalry II

John Gibson – Tripwire Interactive





TRI+IRE

TRI+IRE
PRESENTS



- Tripwire Interactive presents *Chivalry 2*, developed by Torn Banner Studios!

With OVER 7 MILLION units sold of the original *Chivalry*, we are excited to grow the franchise with the launch of *Chivalry 2*.



Chivalry 2:

An Unparalleled Medieval Melee FPS Combat

- Use swords, axes, bows, hammers and more to take the battle to your enemies
- Over **30 weapons** to master
- Unleash your inner Knight with completely revamped, accessible combat, refined for a broad audience



Chivalry 2

Epic Multiplayer Battles

- Dominate massive battlefields of **up to 64 players**
- Lay siege to castles, set fire to villages and use catapults to tear the earth apart in multi-phase multiplayer battles
- **Ride to war** and claim glory by your blade



Key Features

Humor and YouTube Fodder

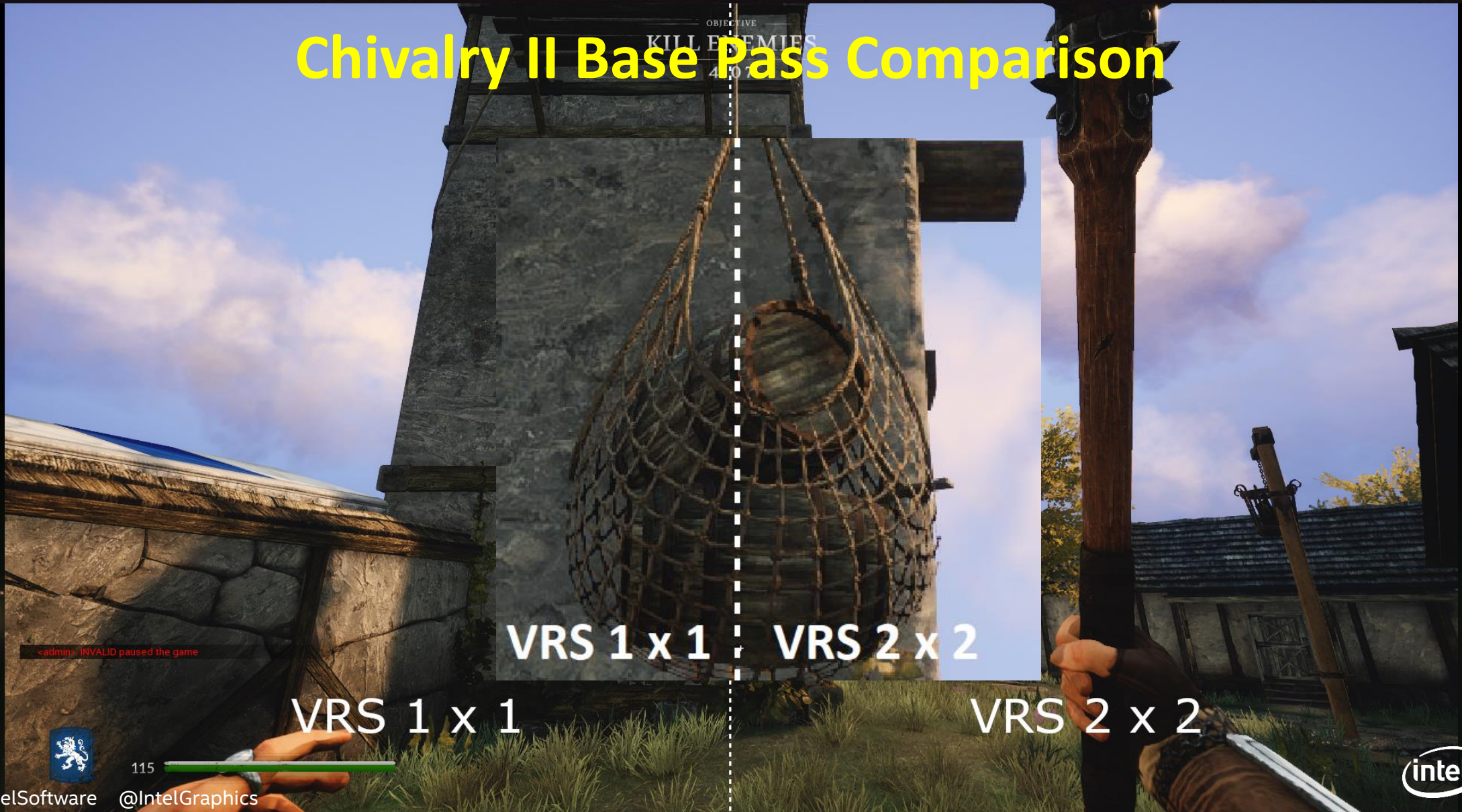
- Live out your “Monty Python” fantasies as you throw chickens, or anything else you can get your hands on.
- Humor filled VO and “pick up and fight with anything” battles make for great Youtube and Twitch moments.



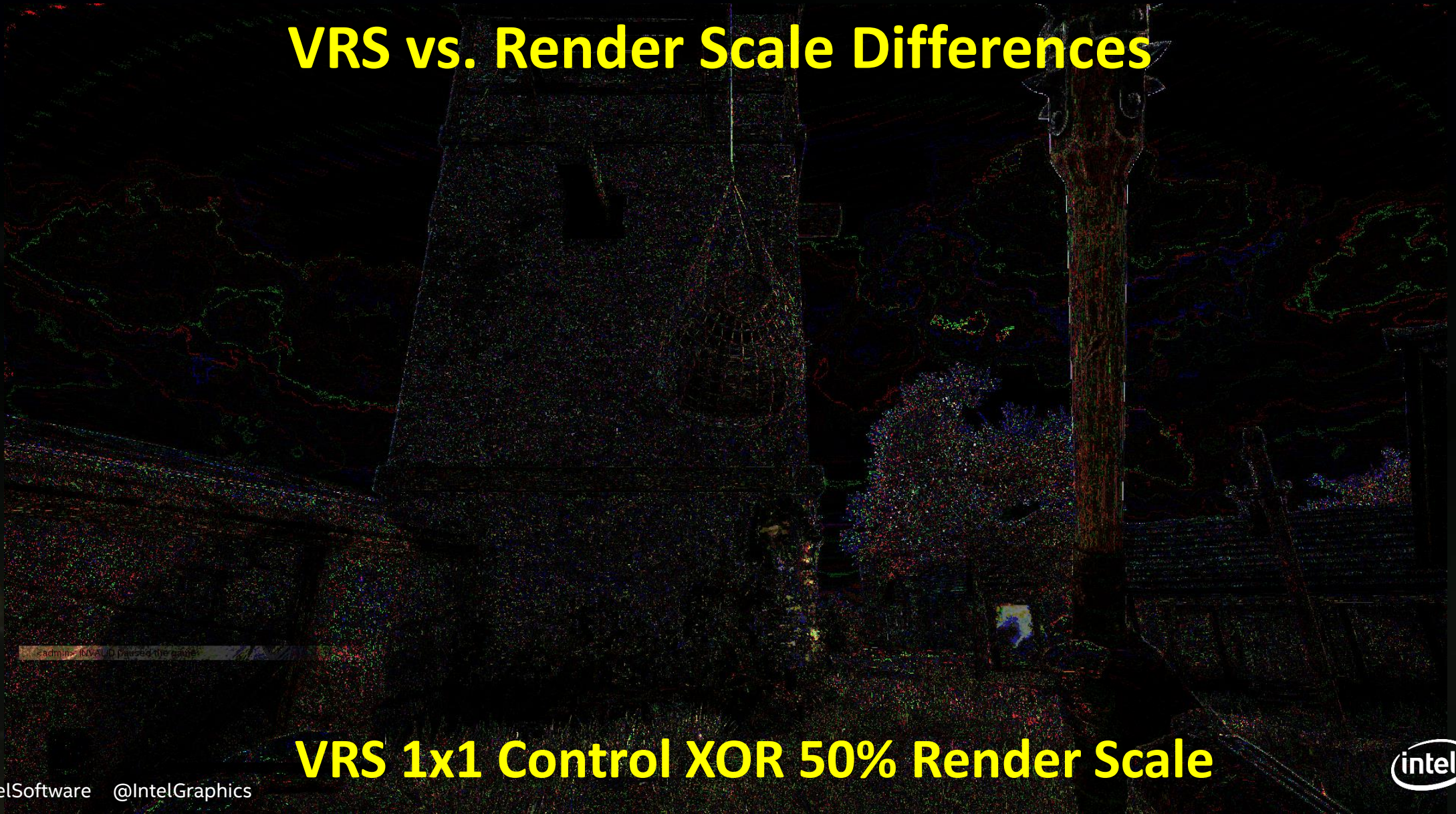
Chivalry II Base Pass Comparison



Chivalry II Base Pass Comparison



VRS vs. Render Scale Differences



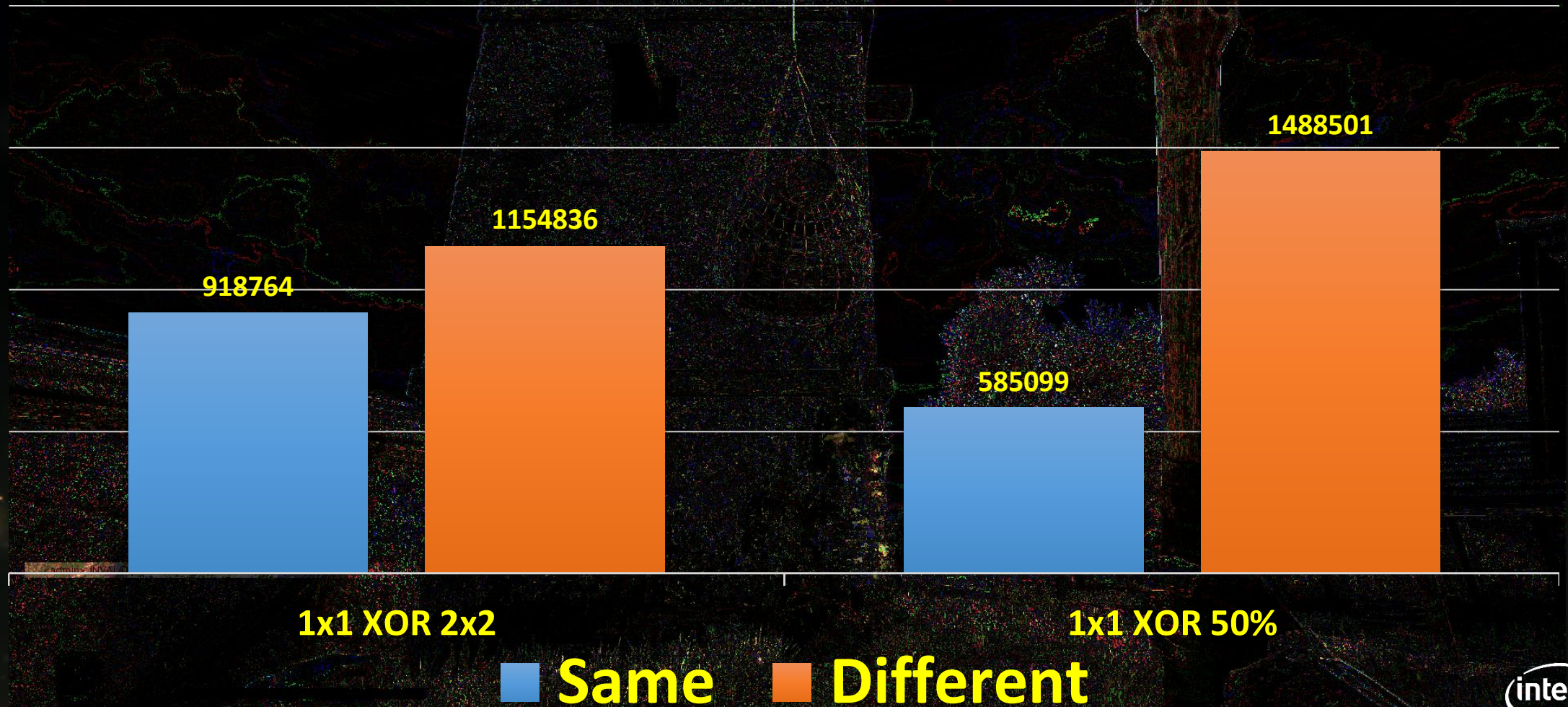
VRS 1x1 Control XOR 50% Render Scale

VRS vs. Render Scale Differences

VRS 1x1 XOR VRS 2x2



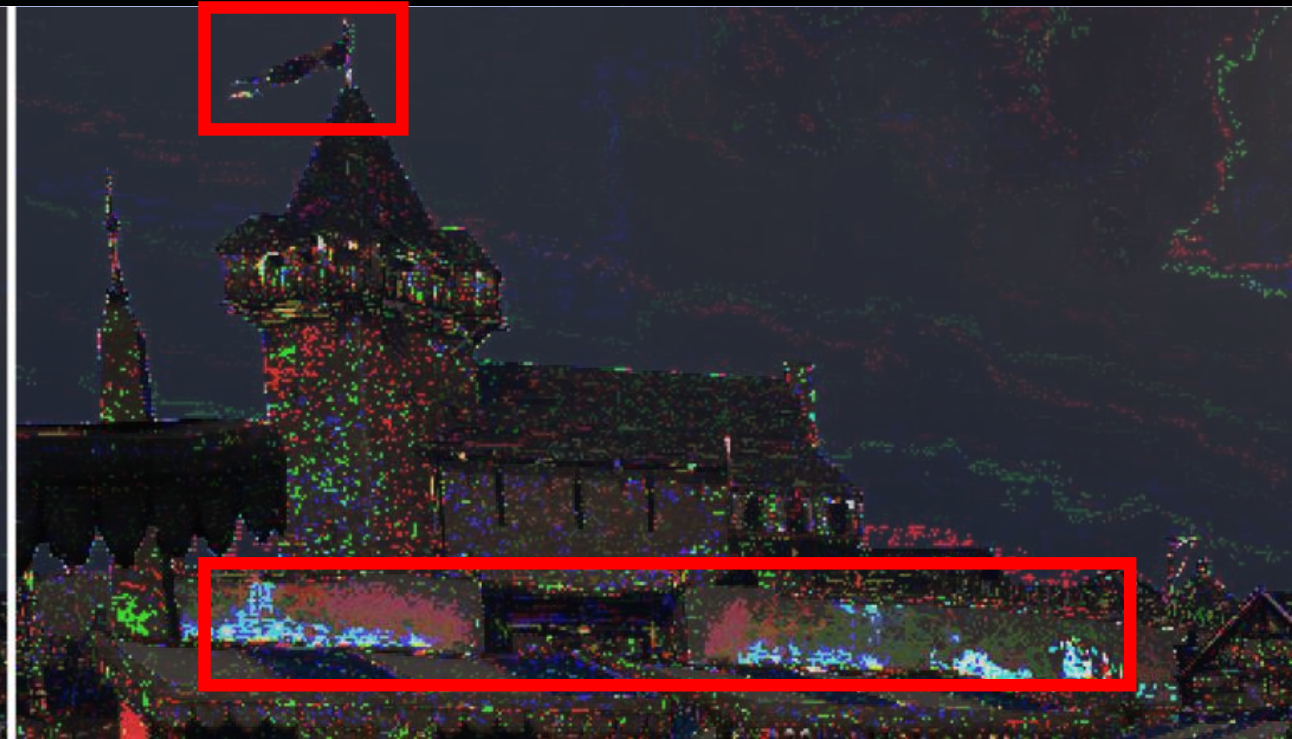
VRS vs. Render Scale Differences



Quality Differences Analyzed



VRS 2x2

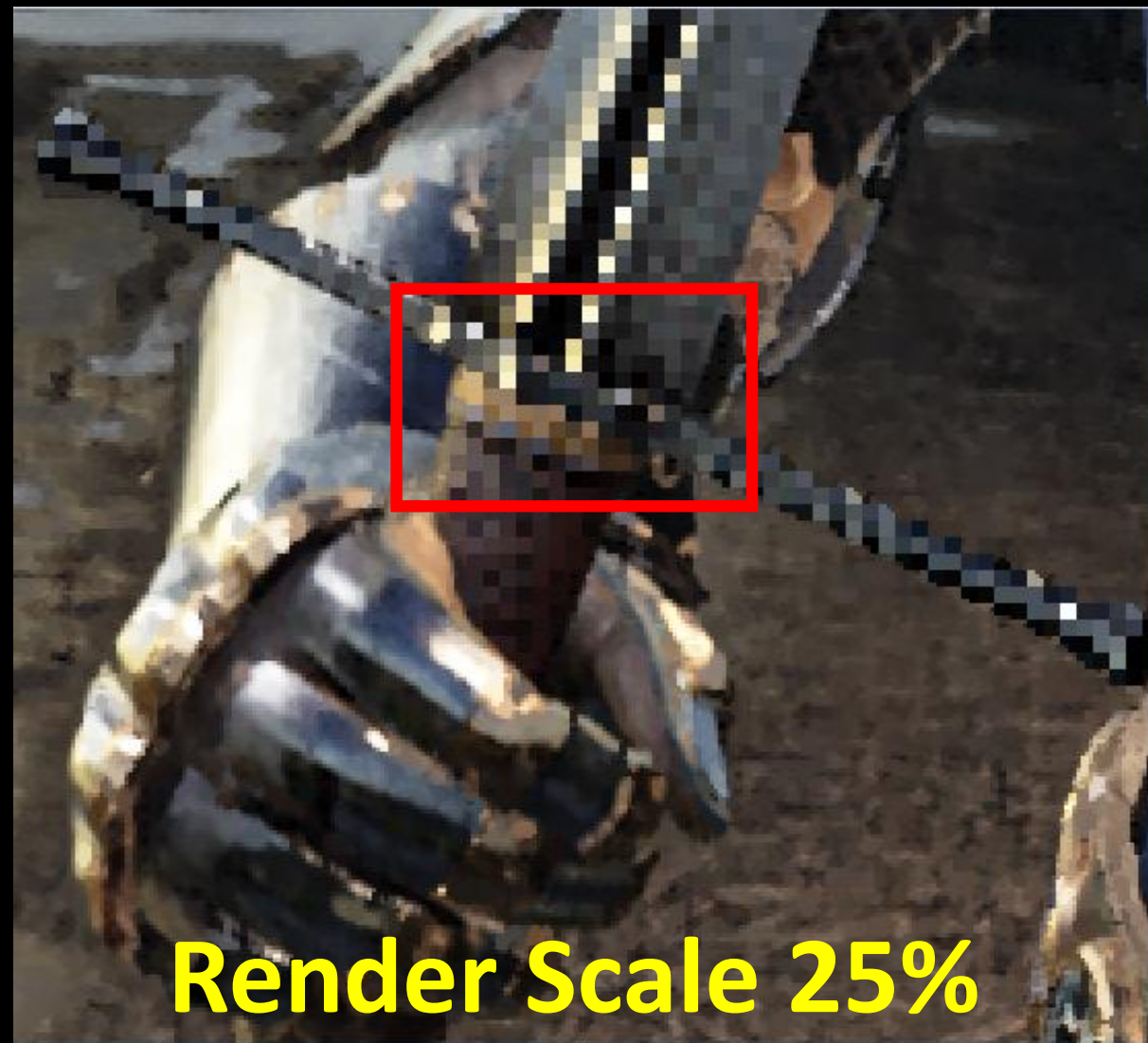


50% Screen Percent

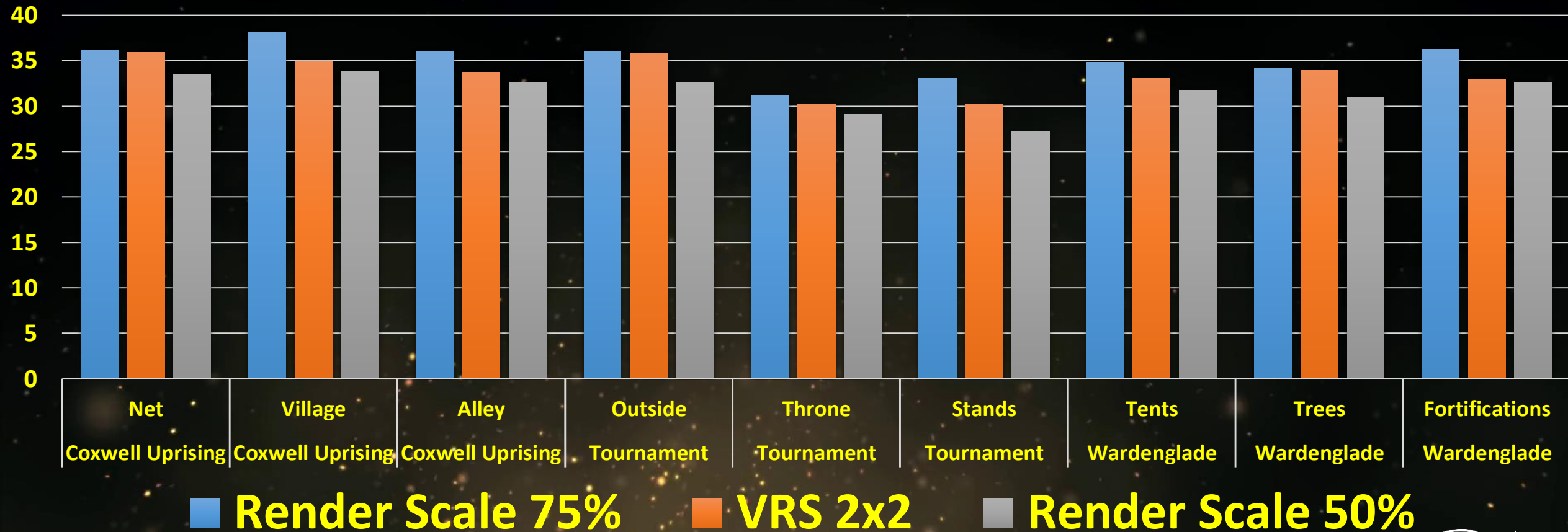
VRS Edge Preservation In Action



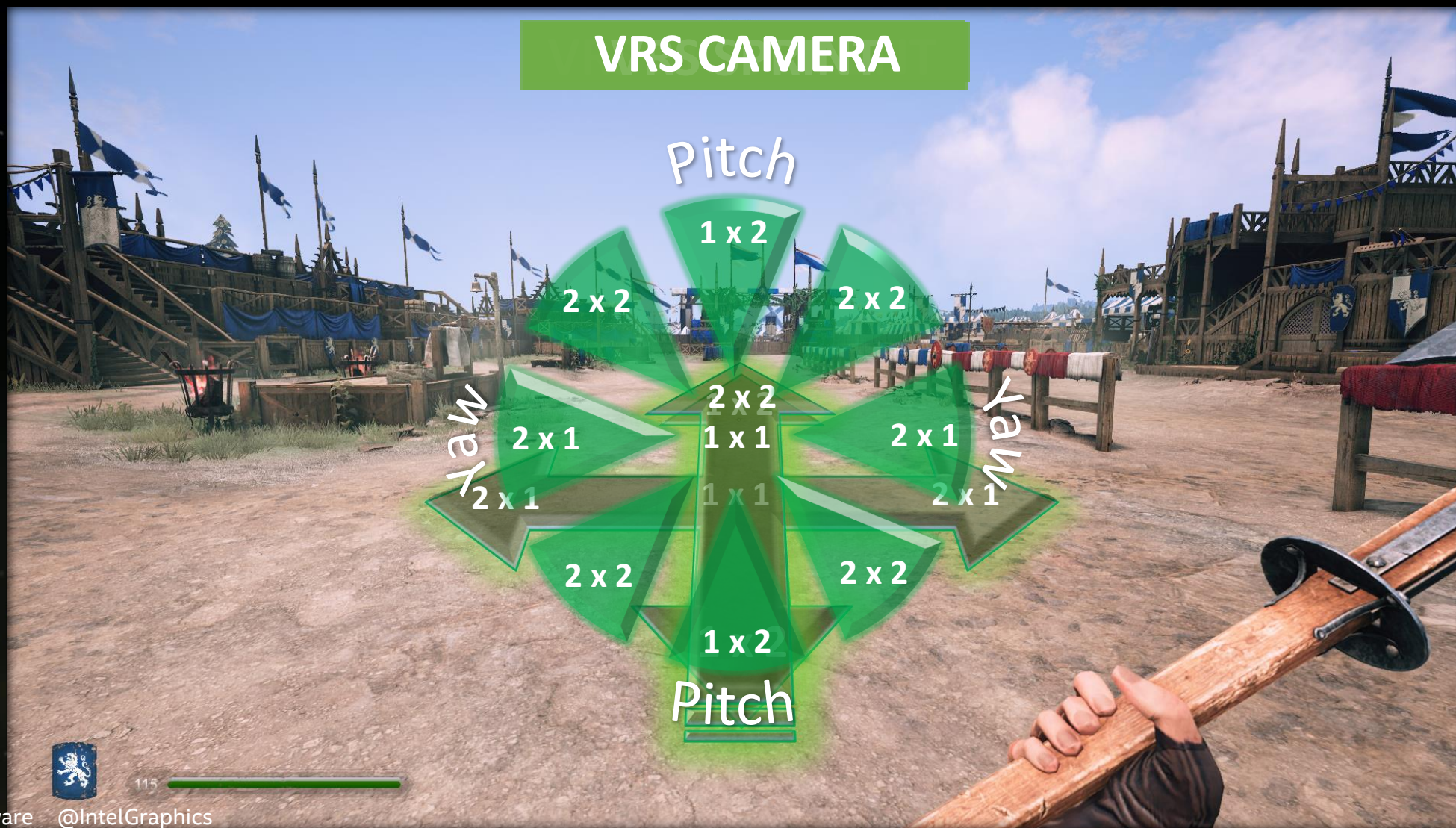
VRS Edge Preservation In Action



Chivalry II PSNR Comparisons (Higher is Better)

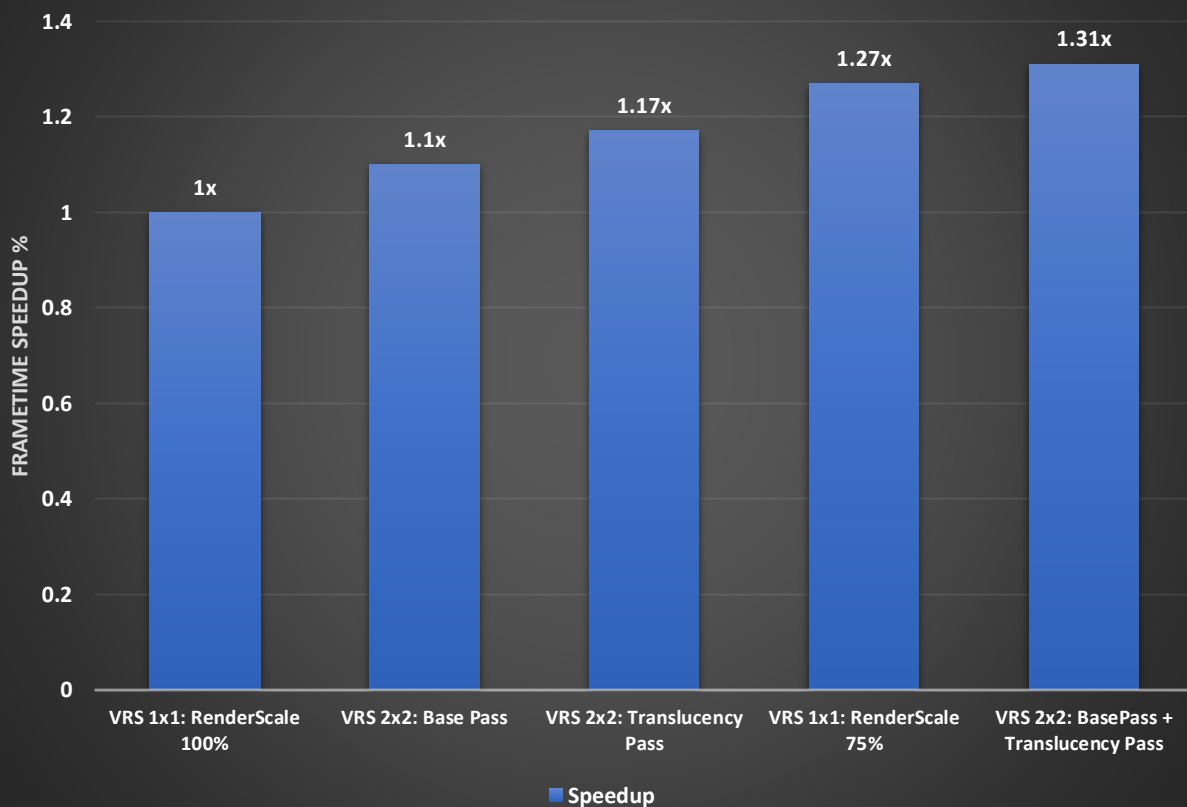


Relaxed Mode Plus in Chivalry II

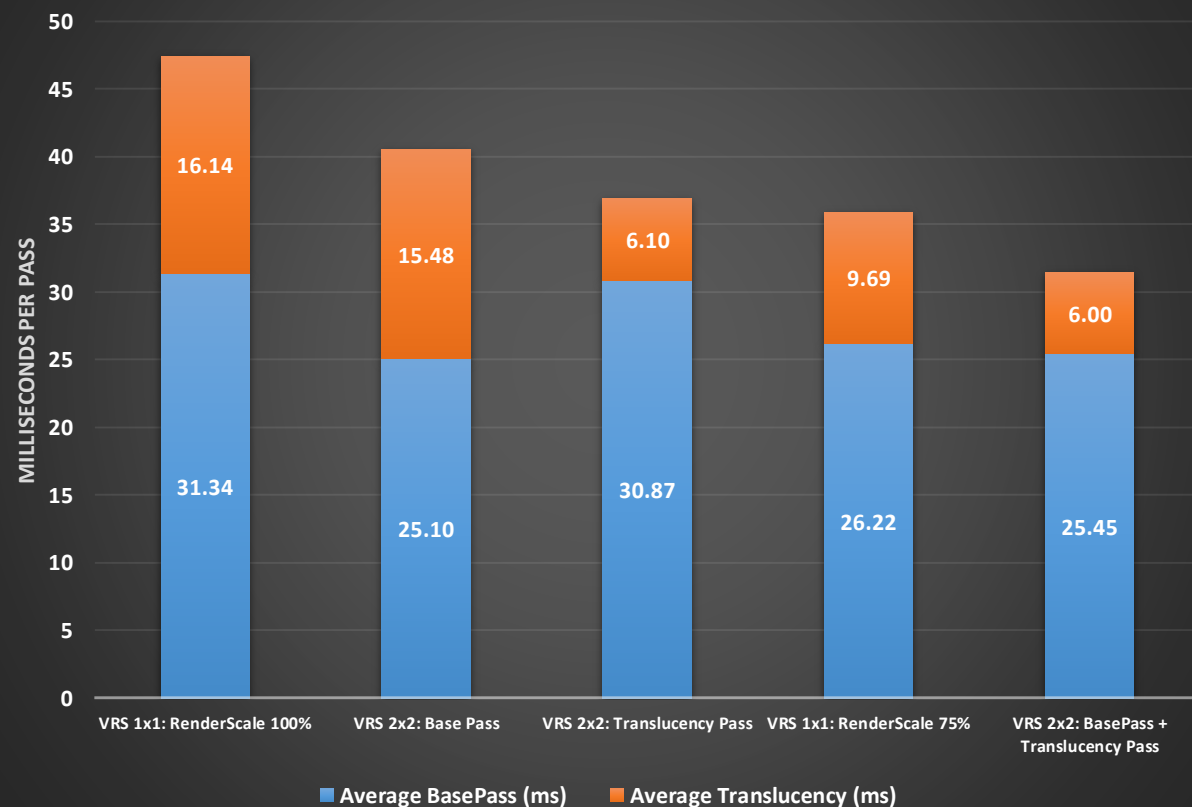


Chivalry II Scripted* Demo Performance Speedup

Scripted* Castle Siege Frametime Speedup
(Higher Is Better)

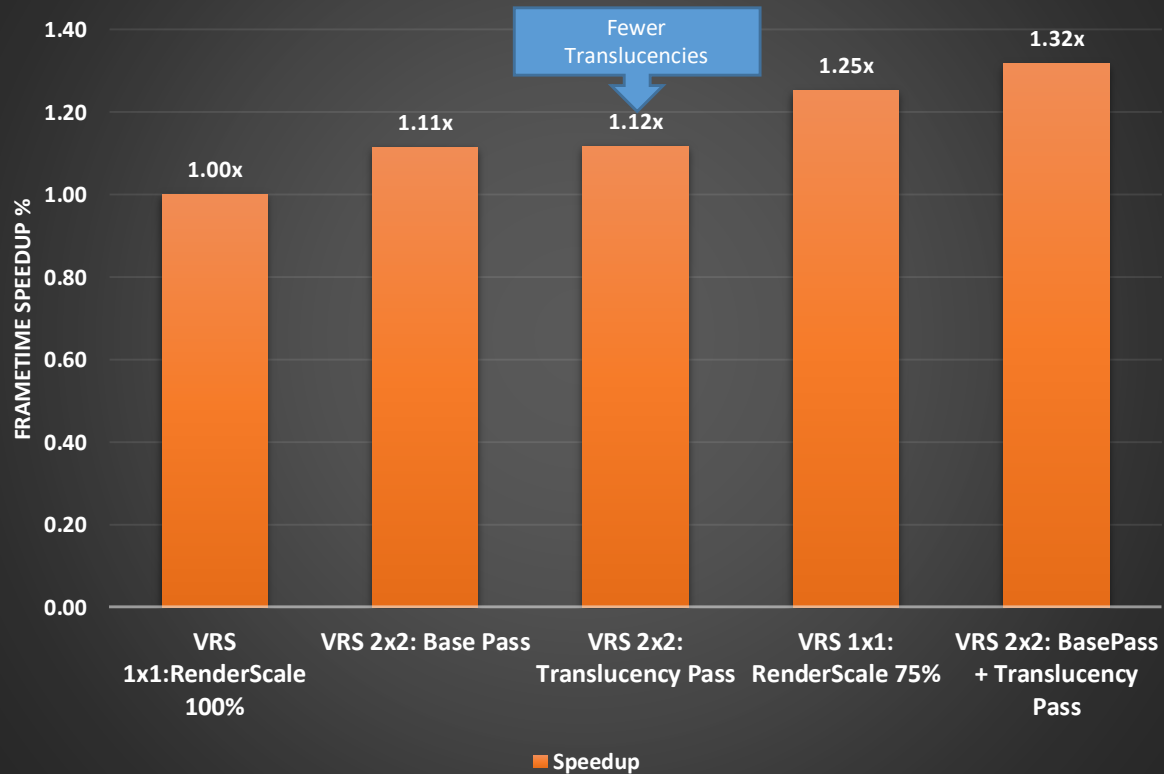


Scripted* Castle Siege Base Pass + Translucency Pass Only
(Smaller Is Better)

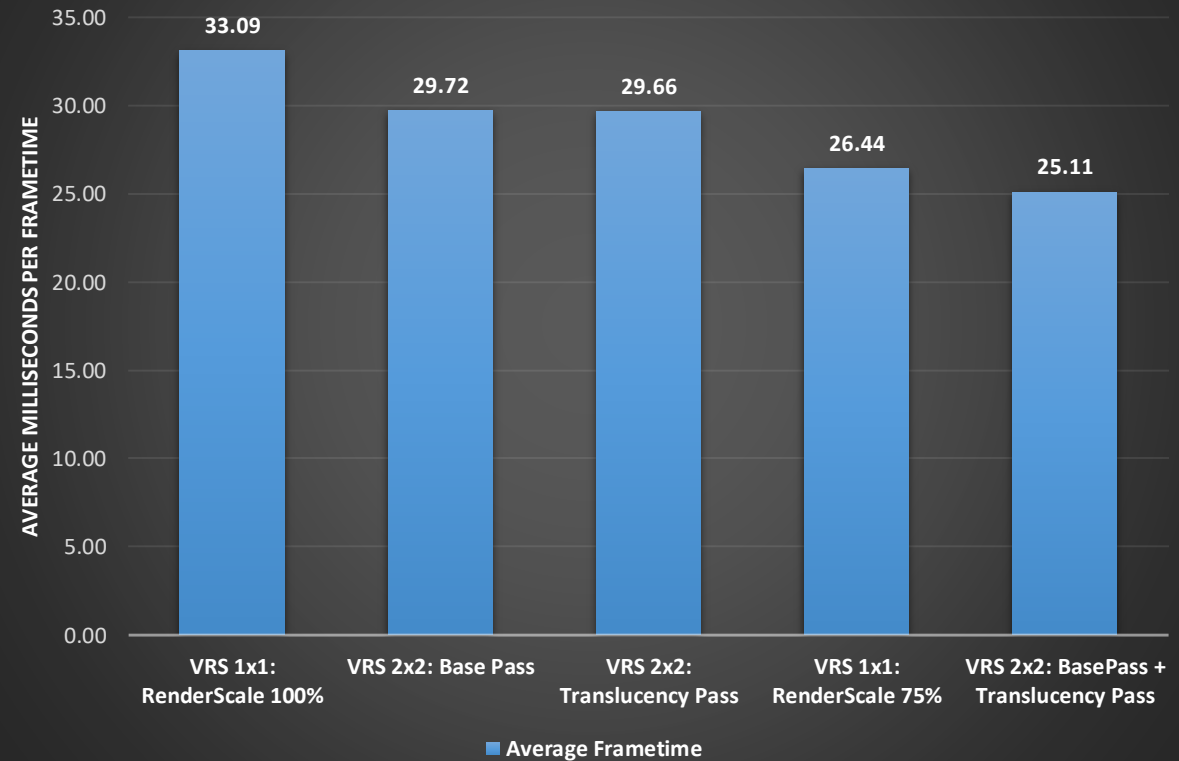


Chivalry II Tournament Grounds* Performance Speedup

Tournament Grounds* Frametime Speedup
(Higher is Better)



Tournament Grounds* Frametime
(Lower is Better)





Thank You!

- Please rate our talk, we get better by listening to you!
- See Our Getting Started with Variable Rate Shading Guide on IDZ
 - <https://software.intel.com/en-us/articles/getting-started-with-variable-rate-shading-on-intel-processor-graphics>



