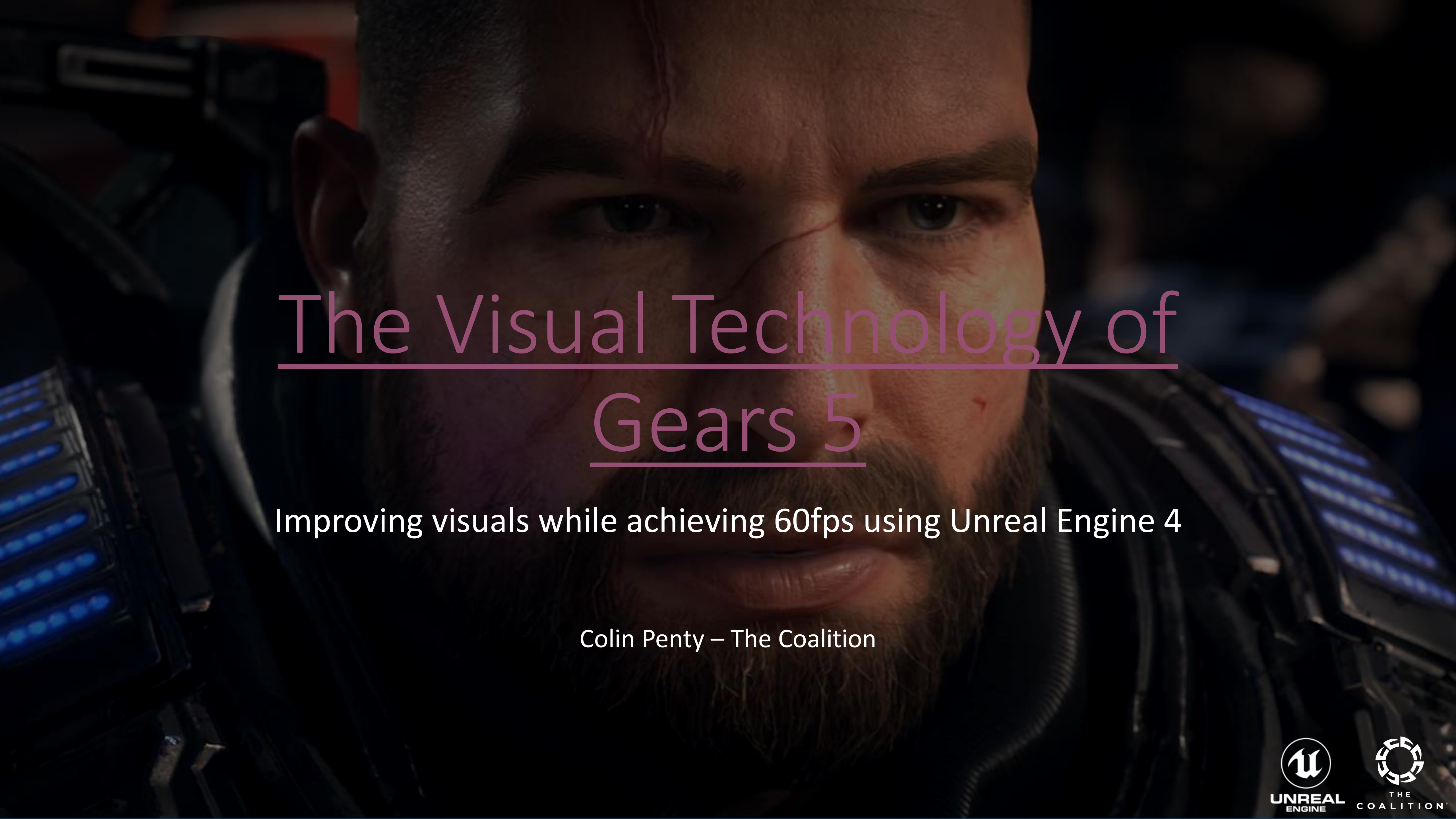


The Visual Technology of Gears 5

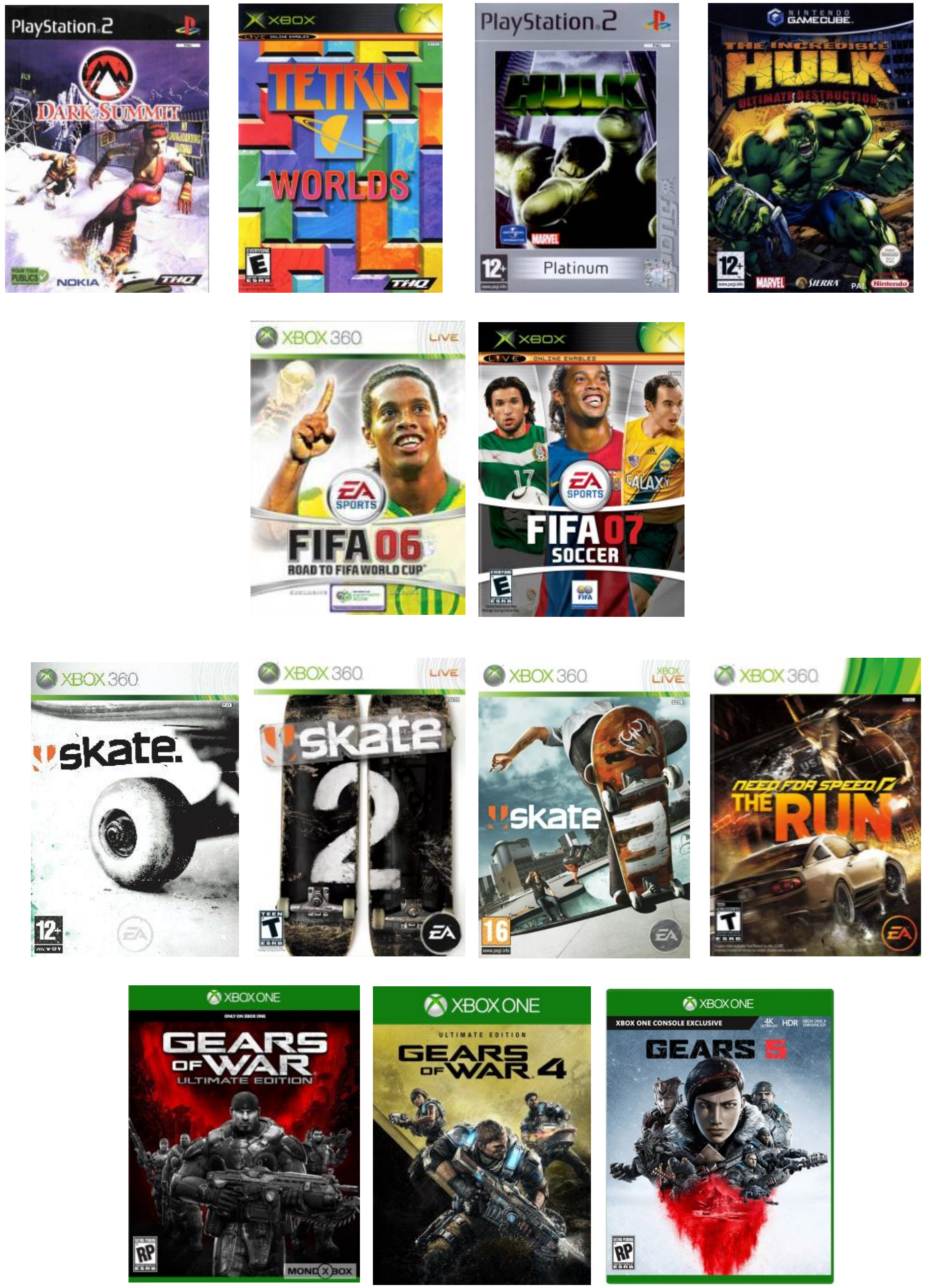
Improving visuals while achieving 60fps using Unreal Engine 4

Colin Penty – The Coalition

Index

Colin’s Background

- 18+ years of Technical Art
- FX Artist/Technical Artist – Radical Entertainment
- Technical Artist/Technical Art Director – FIFA
- Technical Art Director – Skate
- Technical Art Director – Need for Speed
- Technical Art Director – Gears of War



UNREAL ENGINE

THE COALITION

Gears 5

- Goal: ***Challenge assumptions*** – visuals and gameplay
- 84 Metacritic
- 3 million unique players in first week. Twice that of Gears 4.
- *"Gears 5 is one of the most **visually stunning** games ever crafted" – The Verge.*
- *"One of the **best-looking** games on Xbox One"* – Digital Foundry
- *"Gears 5 Is **Beautiful** In A Way Nothing Else On Xbox One Or PS4 Can Match"* - Forbes

Gears 5 tech foundation

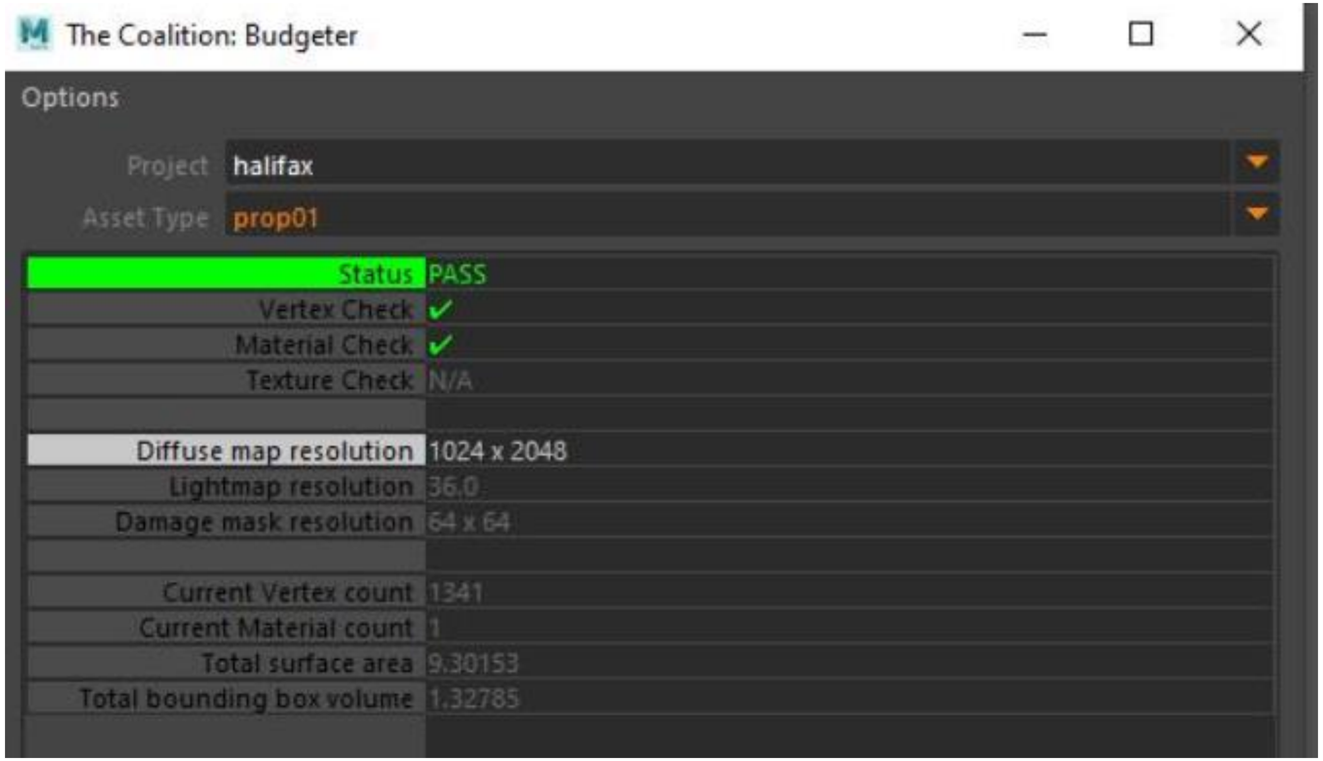
- Used modified UE4 4.11 carried over from Gears 4. Called the Fenix Engine.
- Spot integrated the rendering engine from UE4 4.19 with some additional 4.20 and 4.21 changes
    - Temporal Upscaling
    - Diaphragm Depth of Field
    - Volume Fog
    - Bent Normals
    - Distance Fields
    - Volumetric Lightmaps
- Large visual tech challenges identified early:
    - 60 fps campaign on X1X
    - Escape mode dynamically assembled tiles and map builder
    - 3 player split screen
    - All real-time cinematics
    - 2 large Overworld spaces

Console	Game Mode	FPS Target	Output Resolution
Xbox One and One S	Campaign	30	1080p
Xbox One and One S	Split Screen	30	1080p
Xbox One and One S	Versus	60	1080p
Xbox One and One S	Horde	60	1080p
Xbox One and One S	Escape	60	1080p
Xbox One X	Campaign	60	4k
Xbox One X	Split Screen	30	4k
Xbox One X	Versus	60	4k
Xbox One X	Horde	60	4k
Xbox One X	Escape	60	4k
PC Ideal	All	60	4k
PC Recommended	All	60	1440p
PC Min	All	60	1080p

Frame Rate and Resolution Matrix

Maya tools – Budgeting Tool

- The budget tool would parse your environment asset based on 4 heuristics:
  - Triangles:
    - Volume
    - Surface Area for polygons
    - Curvature
  - Textures:
    - Surface Area for texel density (texture resolution recommendation)
- Budgets for Characters and Weapons are pre-defined by folder location/asset type and not dynamically determined. Will not export unless the assets pass agreed budgets.
  - We have triangle budgets as well as texture resolution and draw call budgets all queried in Maya.

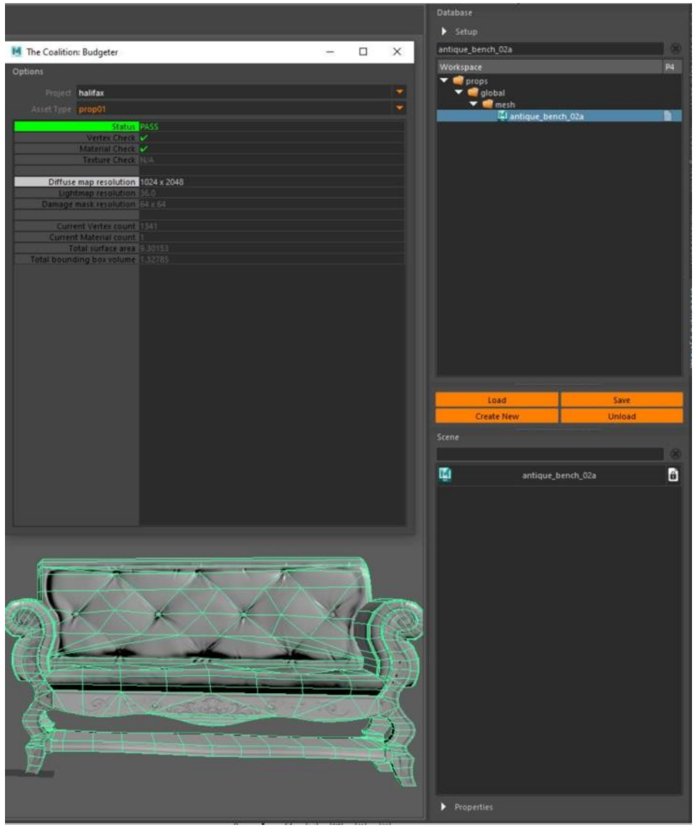


Triangle Budgets

- Hero gameplay characters roughly 80,000 triangles in UE4. Almost 2x more than Gears 4.
- Cinematic characters roughly 100,000 triangles. 50% more than Gears 4.
- Hero weapons roughly 15,000 triangles. Similar to Gears 4.
- Non-hero gameplay characters roughly 60,000 triangles. Similar to Gears 4.

UNREAL ENGINE

THE COALITION

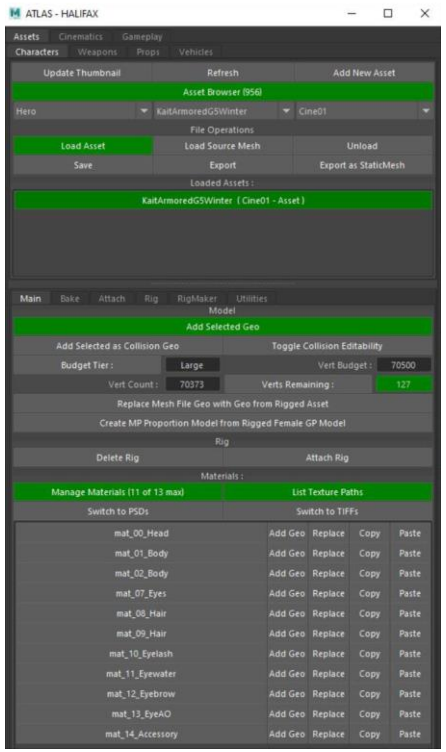


Maya Tools - Cloud Atlas

- Asset organization tool in Maya for Environments
- Primarily using fbx file format
- Ensures all fbx files have Unreal friendly export settings and valid texture paths.
- Essentially a UE4 Content Browser in Maya with exporting abilities

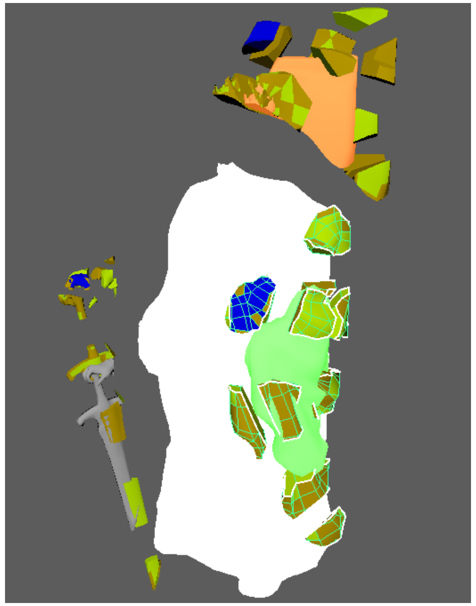
Maya Tools - Atlas

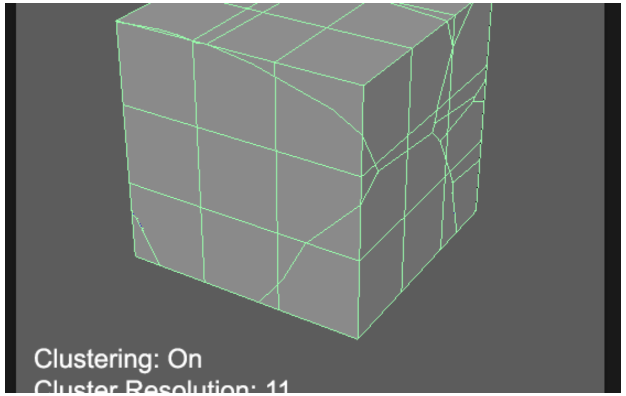
- Asset organization tool in Maya for Characters, Weapons, Animations, Cinematics, Vehicles
- Primarily using ma file format
- Ensures all ma files have Unreal friendly export settings and valid texture paths.
- Enforces budgets – if over won't allow you to submit
- Allows exporting of cinematic sequence animated in Maya into UE4 (matinee)

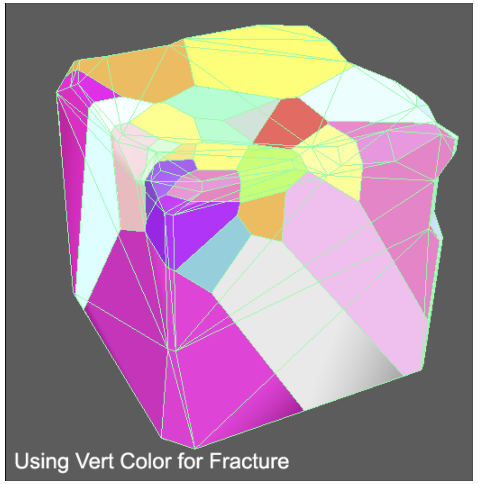


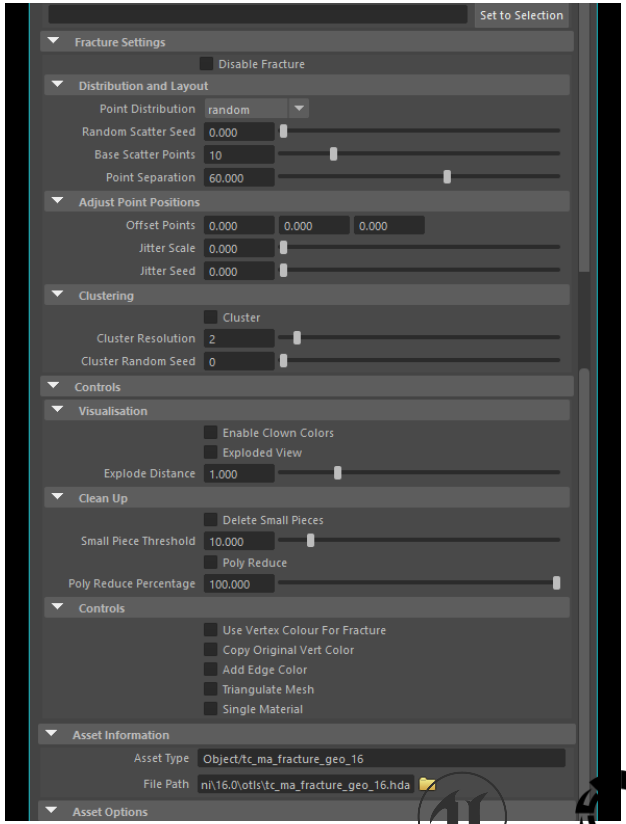
Houdini Tools – Destruction Tool

- Houdini Engine based, used in Maya
- Wired into our Swift Destruction system to create optimal results
- Fracturing tool that will maintain a core for cover objects.
- UV information is used to store fracture point details
- Assigns appropriate materials to outer and inner faces









UNREAL ENGINE
THE COALITION

Houdini Tools – Alembic Cache Destruction

- Custom Alembic Cache Destruction workflows.
- Compressed transforms that we skin per-frame.
- 8 bytes per key frame per object
- Implemented for Gears 4 before Epic had fully implemented in UE4.
- Basically at parity with UE4 4.23’s implementation.
- These can be incredibly expensive for memory usage so use sparingly.

LIGHTING NEEDS TO BE REBUILT (3286 unbuilt object(s))
MATERIALS NEED TO BE UPDATED (6 stale material(s))
TEXTURE STREAMING POOL OVER 633.31 MB

Volume Fog

- Was an important component of our visual aesthetic for Gears 5
- Exposed Volume Fog settings in Post Process Volumes
  - Allowed us to modify grid pixel sizes for performance per area
- Exposed Fog values in post process volumes as well
  - This was mainly to resolve replication issues in split screen that we encountered on Gears 4
- Settled on Grid Pixel Size of 16 as a default

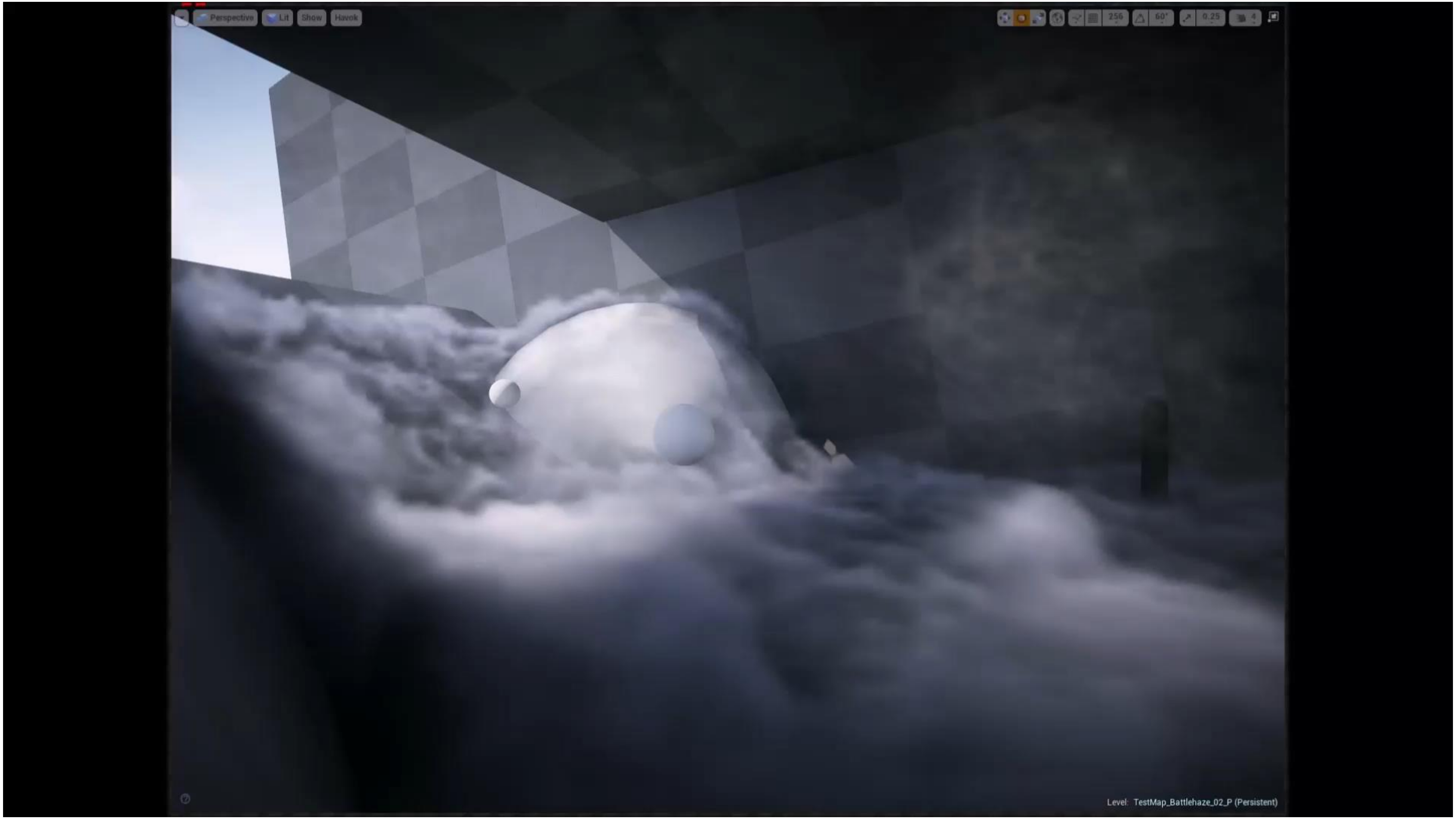
LIGHTING NEEDS TO BE REBUILT (31826 unbuilt object(s))
MATERIALS NEED TO BE UPDATED (15 stale material(s))

Volume Fog Painting Tool

- Added a blueprint based volume fog painting system.
- Allowed us to side-step a lot of the injection cost that comes with volume fog.
- Entire level had a cube around it.
- Would place volume fog primitives around the level. Would be great to see as a 1st class system in the future.
- Would also generate a height field to represent ground for ground fog and integrate with the paintable volume fog system (rendered into the cube).

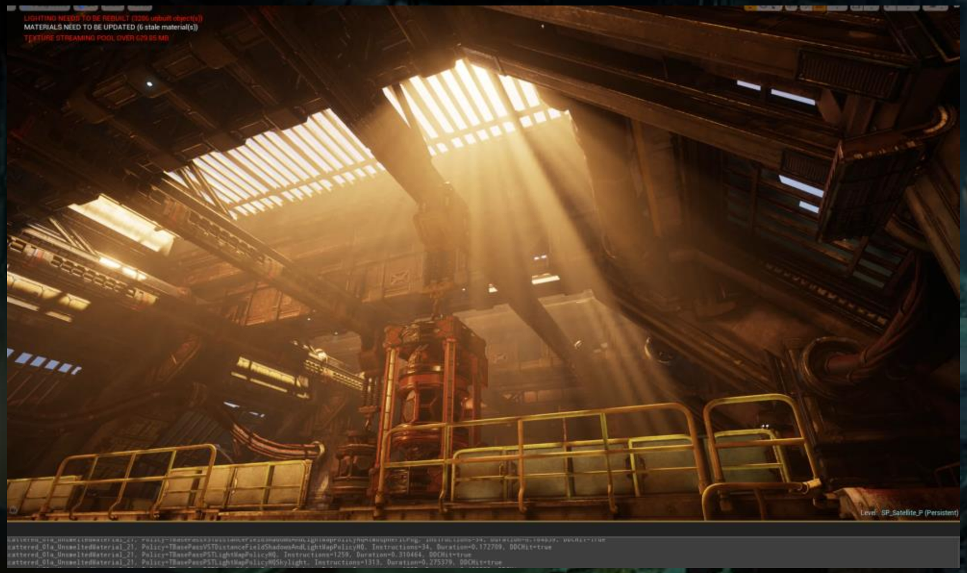
Volume Fog Shaders

- Did many experiments with interesting Volume Fog Shaders
- Was not able to achieve high frequency “noise” inside the shader without making the fog with ridiculous grid pixel size values. (Right video)
- Ended up complimenting low frequency volume fog shaders with additive billboards to add high frequency noise. (Left image)


Perspective
Lit
Show
Havok
256
60°
0.25
4
Level: TestMap_Battlehaze_02_P (Persistent)

Lighting

- Push to a more dynamic lighting solution than Gears 4
  - Looks better, consumes less texture memory, and more flexible.
  - Knew we had large Overworlds where cooking shadows may not be realistic
- Removal of stationary lights (therefore shadow maps)
- Gears 4 used per-object shadows from Directionals. Gears 5 uses cascades + raytraced distance field shadows – inspired by Fortnite.
- Entire game uses static lights and moveable lights only (+stationary skylight for real-time skylighting)
- Added the ability for moveable lights to output GI lightmass cooked data like a stationary light.



Lighting Performance/Memory equilibrium

- Problem: Needed to shrink texture pool to make room for HLOD's and other systems
  - On Gears 4 at peak points the lightmap and shadowmap data could be 400mb
    - This was primarily a result of lighters trying to get crisp shadowmaps it was determined. Not due to bounce lighting fidelity requirements.
- Gears 5 averages about 50mb of lightmap data in the pool due to removal of shadow maps and stricter lightmap memory requirements
- Integrating the moveable light shadow cacheing improvements in UE4 4.2x was a life saver for performance
  - Performance of a few characters standing in a spotlight was faster than using a stationary spotlight.
- Removal of all stationary light shader permutations was a great GPU and Memory win.
- Removal of all Shadow Maps was a great disk space and texture pool win.

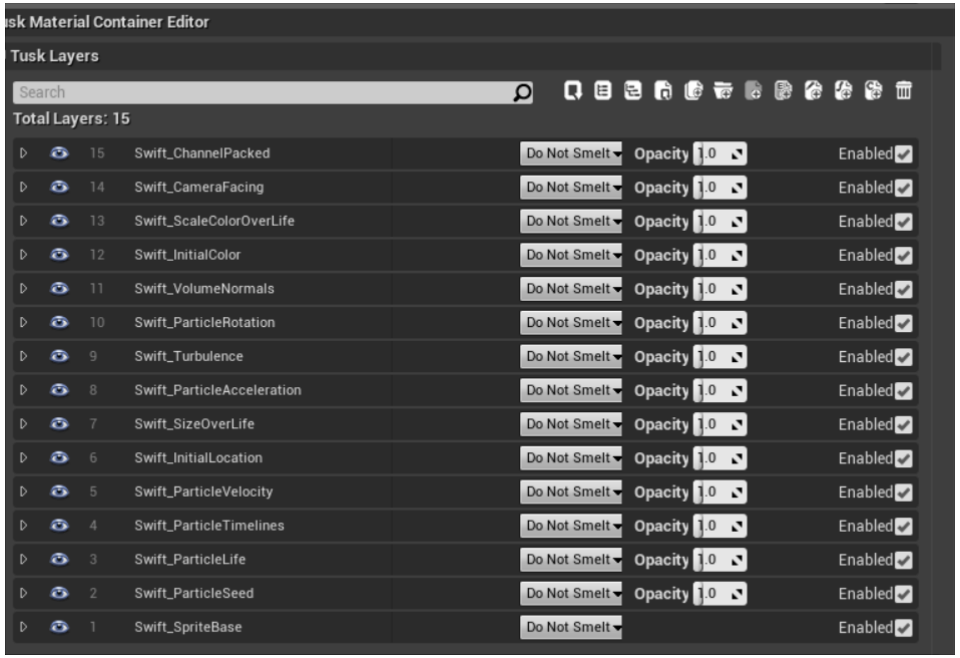
DF Ray Traced Shadows

- All directional lights in Gears 5 use Ray Traced Distance Field Shadows beyond the shadow cascades.
- Was a great fit for Gears 5 due to heavy HLOD use and essentially zero draw thread cost compared to alternatives like “far cascades”.
    - (Even the Xbox One X was heavily Draw Thread bound in campaign at 60fps due to the CPU being relatively weaker than the powerful GPU - so any Draw Thread win was great.)
    - (HLOD’s need to toggle visibility, this is traditionally hard to do with baked shadows)
- Was a delicate balancing act on the cascades transition distance.
    - Ray trace shadow transition being pushed back = improved GPU performance due to less objects needing ray traced shadows but will incur worse draw thread performance due to additional objects in the cascades.
- SDF resolution was lowered as low as possible to reduce memory overhead. This was usually “fine” due to knowing the objects were only ray traced in the distance and never close.

UNREAL ENGINE THE COALITION

Swift Particles

- Particles was determined to be one of our main roadblocks to 60fps campaign. Needed a GPU solution.
    - GPU Particles are great but still incur a CPU Spawn cost.
- We did 2 things to address this:
    - Created a GPU Spawning pass for Cascade inspired by Rare's technology.
    - Created "Swift Particles" which is a particle system built in the material editor leveraging vertex offsets for environmental VFX.
- Swift particles is a fully featured particle system with the ability to add your own material functions for new behaviour like vector fields. It was built on the backbone of our Material Masking System (MMS) which is a modular material creation system.



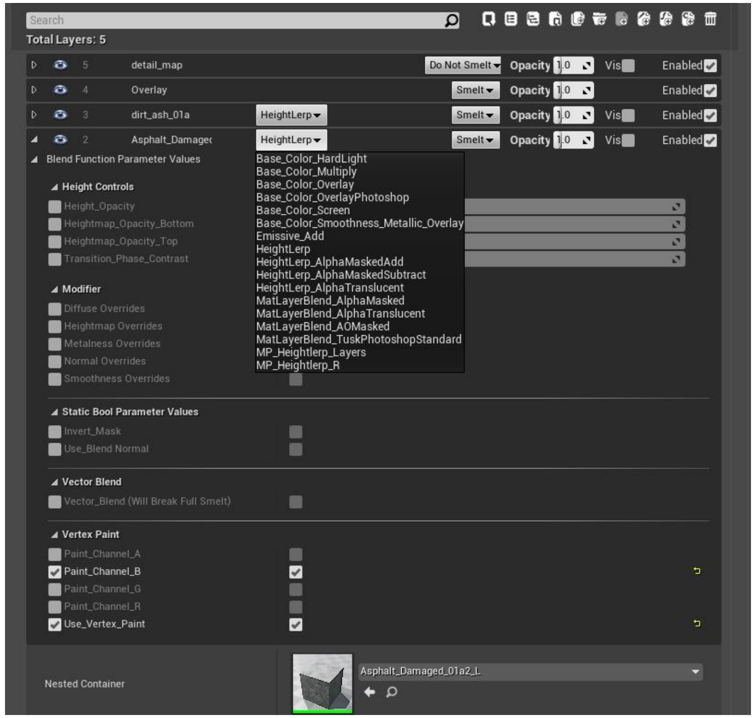
Swift Particles - Venom

Swift Destruction

- Leveraging the same core approach as Swift Particles – we built a vertex offset based destruction system.
- This was tied in tightly with our Houdini Engine destruction tool and various UE4 editor tools.
  - We would automatically generate a high res height map of the ground around a Swift Destructible using a Blueprint so the individual pieces could collide with the ground.
  - We could have roughly 4 fracture points on a Swift Destructible before the GPU overhead of the material would become prohibitive.
  - We would bake a lot of the destruction info into the UV’s in Houdini Engine/Maya to pass through to UE4 and the Swift Destruction material.

Material Masking System (MMS)

- Built upon our proprietary MMS system from Gears 4. Presented at SIGGRAPH 2017.
- Allows the dynamic creation of a material using “layers” in UE4, then on save – it saves an optimized version of the material – cooking out textures where possible.
- Smelter = what we call the texture cooking pipeline
- Container = what we call a collection of material layers
- Container can be a collection of “live” and “smeltable” layers. Smelter does it’s best on save.
- Containers can be nested into other containers
- Each layer has photoshop like controls such as opacity, mask texture, blend mode, etc
- MMS improvements from Gears 4:
    - More workflow than technological: Moved to a unified shared material library for all teams. No longer had to re-invent “concrete” 50 times throughout the project!
    - Improved texture packing (more on this later)
    - Artist visualization to what layers are going to smelt properly, partially smelt, or not smelt at all based on colour coding. This helped reduce confusion for why a material might be expensive or not.

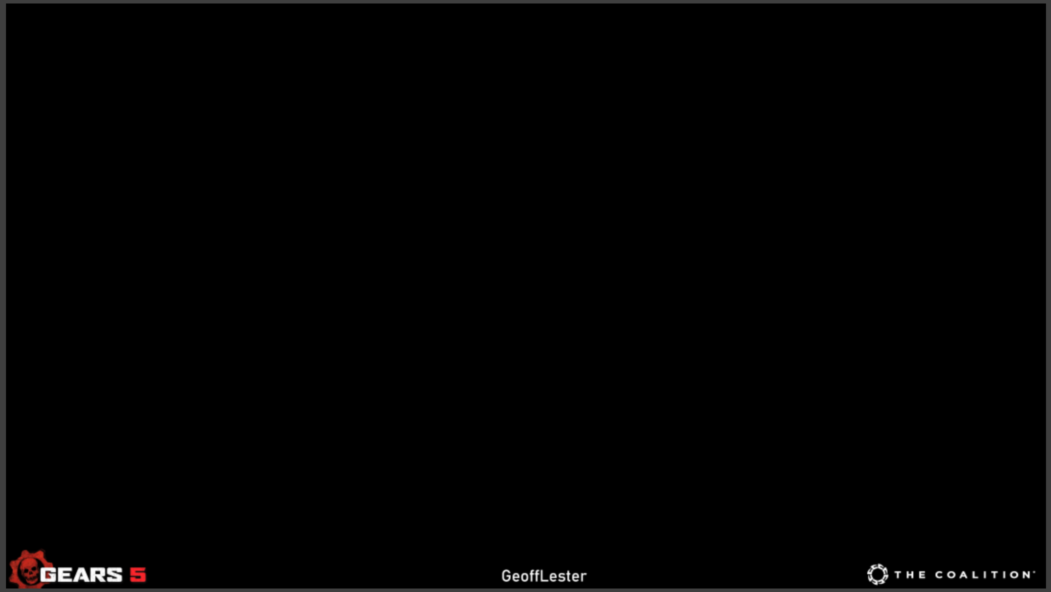


Snow and Sand

- Complex materials that needed to allow multiple layers of blending for the overworlds to look believable but also had to be extremely efficient and had to work with our deformation system.
- Deforming the sand/snow would reveal a different material layer underneath as well as have a "perturbed" material blended along the edge of the deformation.
- Consisted of glitter material function and tuned Fresnel values.
- Had landscape associated debris per-layer for instance mesh spawning
- Could dynamically paint it on top of objects using up-vector
- Had ability to "flood flatten" all sand/snow down for QA Testing.

Snow and Sand - Deformation

- Initially tried to using our particle deformation system, but particles get expensive to keep around for permanent trails
- Went back and built a new system that could store trails for a large data-set.
- Persistent deformation is stored in a texture atlas with an indirection layer for efficiency. (Think megatexturing)
- The system would output a depth value and a mask that tech art could then use to build material deformation materials around.
- Async tessellation was key to achieving visual quality on trails. It would cost about 0.3ms to run tessellation on Xbox One X.



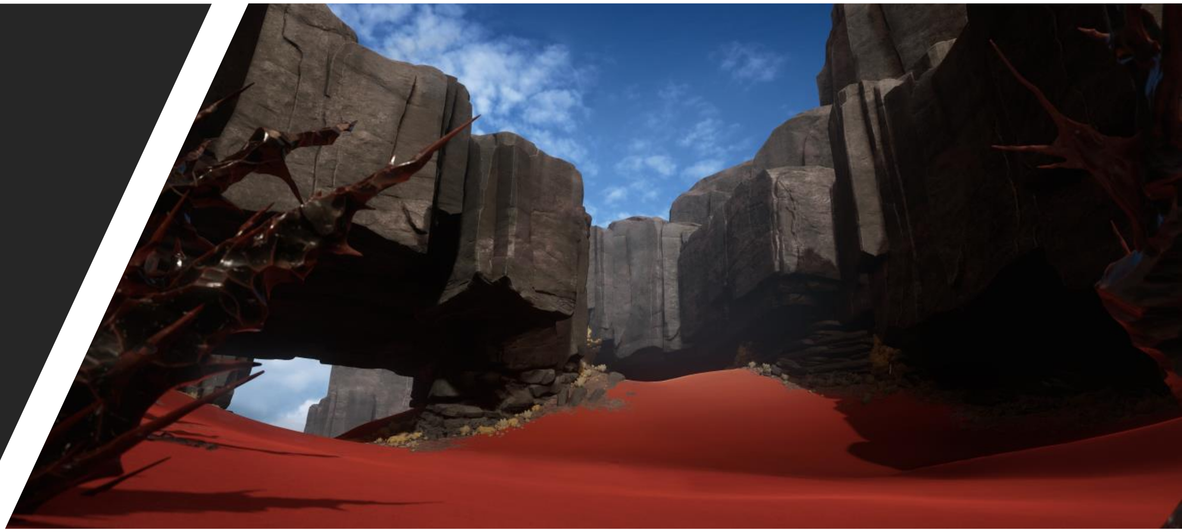
UNREAL ENGINE THE COALITION

Relaxed Cone Step Mapping

- Based on old Nvidia paper
- Produces stronger visual results than Parallax Occlusion Mapping and is faster to compute on GPU.
- Requires pre-computed “cone step map” to be generated from height map.
- This docks in well for us with our Material Masking System.
- Can poke depth buffer and self-shadow or contribute to SSAO for additional cost

Overworlds

- Challenges:
  - Large outdoor spaces that were up to 7km x 5km.
  - Skiff vehicle that could move extremely fast
- Authored sculpted landscapes in World Machine then imported into UE4's landscape system
- Used "World Composition" for streaming
  - Broke sub-levels into Large, Medium, Small categories for streaming
- A large exercise in memory management.
  - LOD Counts
  - Triangle Counts
  - HLOD usage
  - Texture memory
- Used Simplygon HLOD Proxy, Epic Merge, and Aggregate techniques
  - Some mountain ranges would use HLODs up-close and far away to prevent popping
  - Did dithered transitions to HLOD's to reduce popping
- Used Imposters to reduce GPU load in the distance on vegetation. Octahedral imposters also where necessary.
- Levered tessellation shaders and dynamic shadows to bring a lot of detail into the foreground

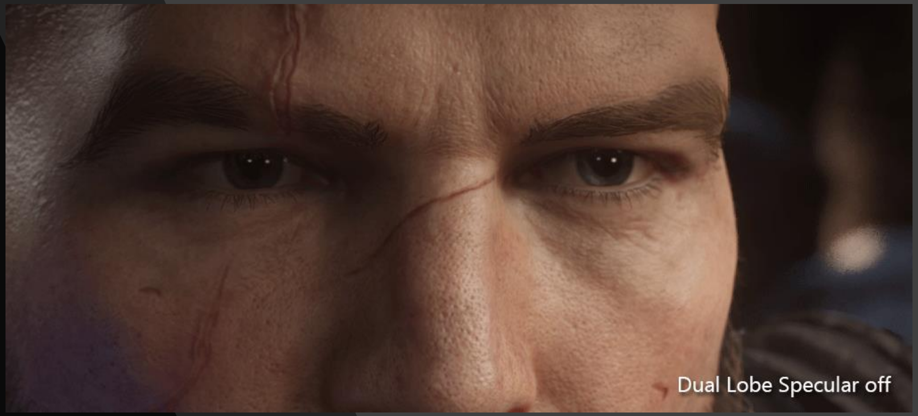


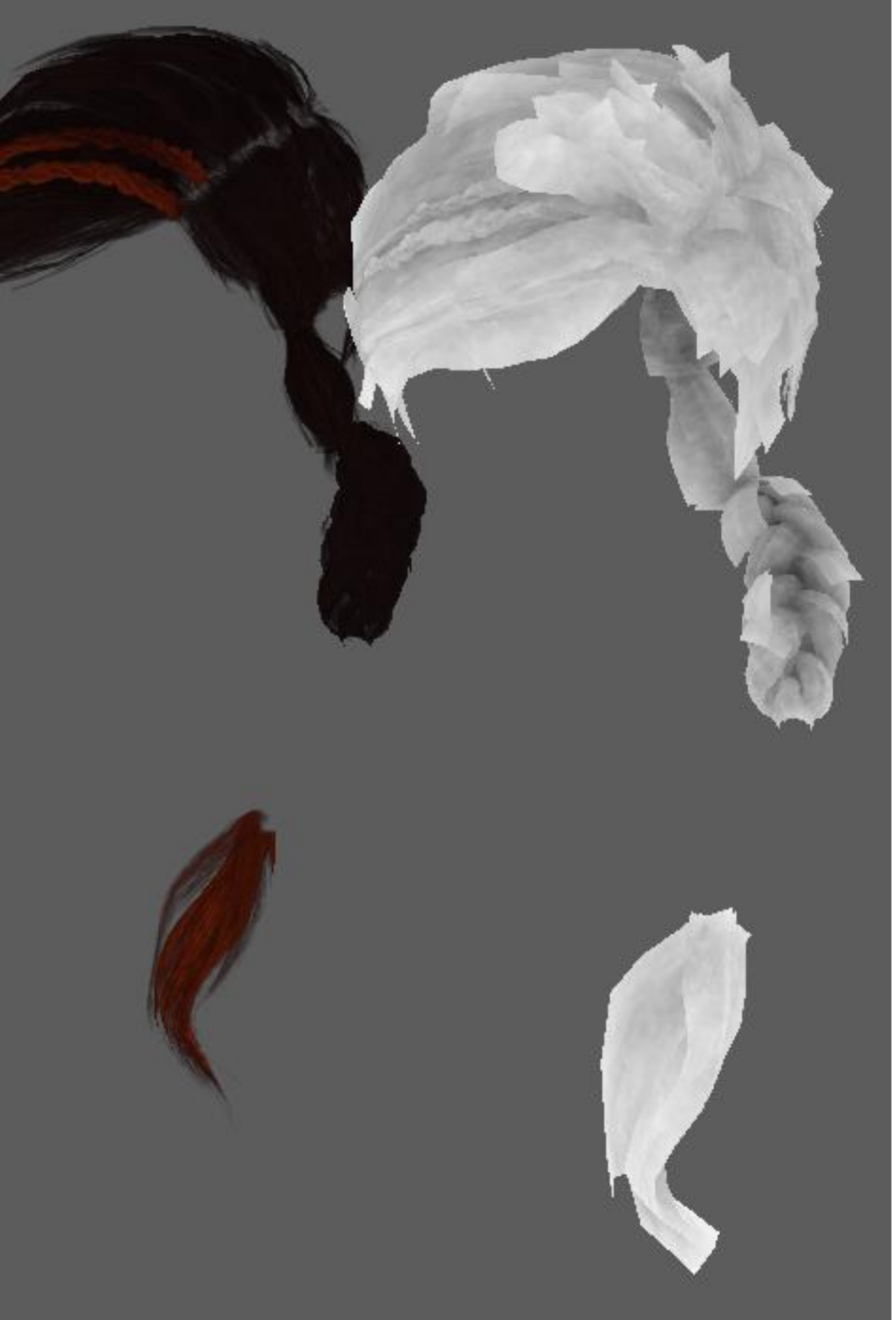
Characters - Eyes

- Added the Paragon bump offset to the Iris
- Setup the eyes use proper SSSSS (Gears 4 did not do this)
- Added Eye Water, Tear Duct, and Eye AO geometry
- Added Eyebrow geometry
- Added a dynamic Iris Caustic system – inspired by Jorge Jimenez and Javir von der Pahlen GDC 2013 talk
- Cubemap based reflection map support

Characters – Skin

- Created a new shading model specifically for Cinematic Skin
- Greatly improved backscattering compared to Gears 4
- Added dual lobe specular (UE4 has this now)
- Detail normal maps for skin pores
- Much improved blood flushing – simulated regional blood flow under the skin
  - Blood flushing tied to expansion and compression wrinkle maps in 19 regions of the face





Characters - Hair

- Identified issue that most DCC tools don't have great support for generating AO maps.
- Built a custom tool for this using Houdini Engine
- Standing up a ray tracer for AO in Houdini allowed lots of control such as:
  - Let rays partially penetrate through cards to simulate transmittance
  - Read the hair alpha texture and lets rays through areas that are transparent
  - Multiple levels of quality presets for quick baking
- Otherwise our hair shader was mostly unchanged from Gears 4 – still using Disney Marschner shading model.

Bent Normals

- This is one of many integrations we made from the newer versions of UE4
- For hero characters we baked out Bent Normals and AO maps to improve quality
- Great for plugging specular and diffuse leaks
- Works great for a 3rd person game with characters wearing armor

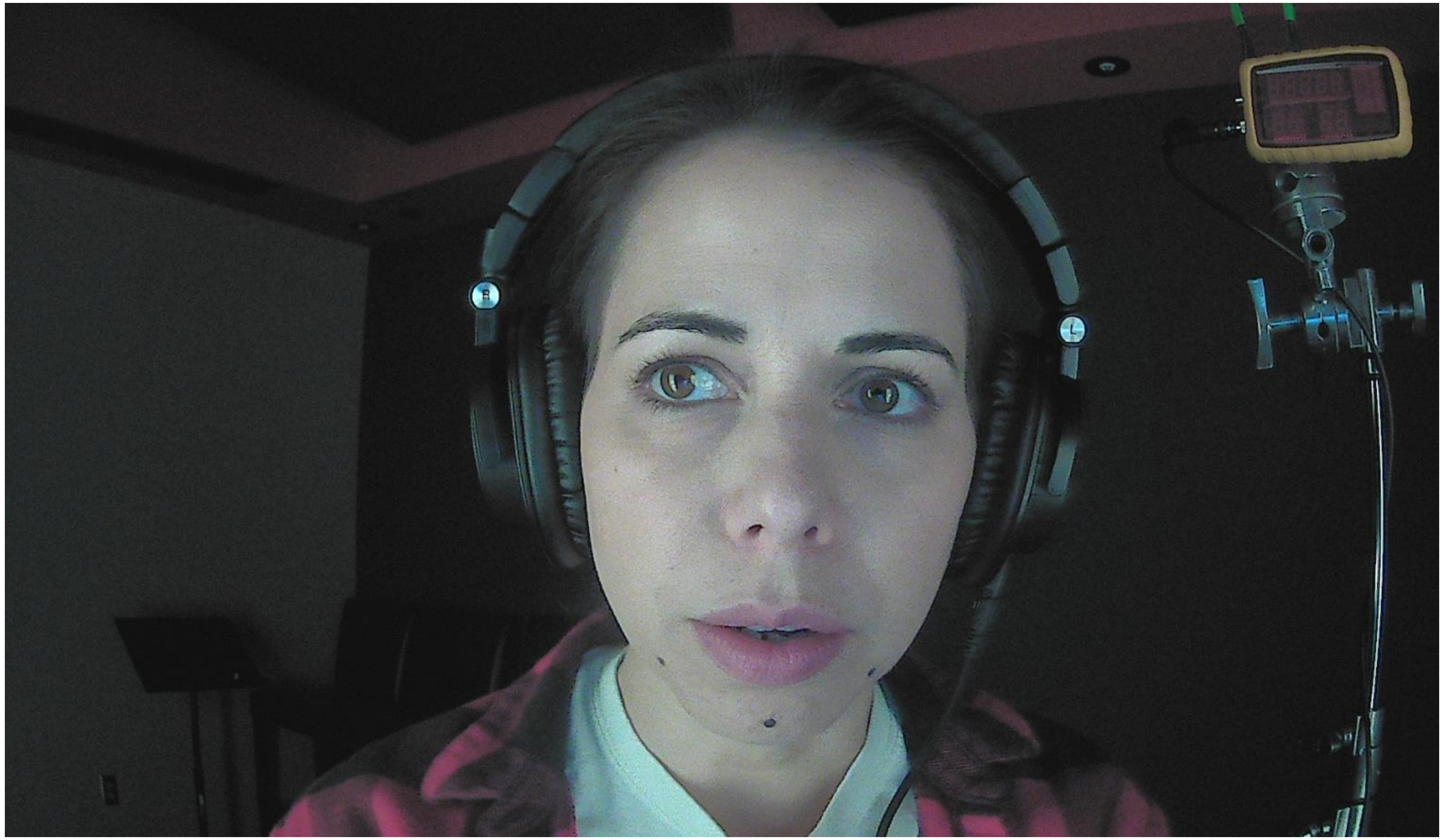
UNREAL ENGINE

THE COALITION

Next-Gen Character Face Rig

- Blendshapes
  - Base Face Shapes - 103
  - Eyebrow Shapes – 28
  - Eyelash, Eyewater, TearDuct Shapes – 26 each
  - EyeAO Shapes – 14
  - Full beard Shapes – 75
  - Total Shapes – 298
- Bones
  - Animated Bones – 67 (43 are skinned)
  - Channel Animation Transfer bones – 93
- Rig Driven Shader FX
  - 19 Wrinkle Regions for Expansion & Compression (Normal map blending)
  - Pupil dilation

UNREAL ENGINE THE COALITION



Characters – Face Performance Capture

- Cinematic face animation is captured and processed with FaceWare
- Uses raw video to track face movements and maps to our custom face controls
- Used for both:
  - Performance Capture Sessions
  - Voice Acting Booth
- Animation team would take this data and work with it to create a compelling performance

HDR

- "*Gears 5 takes the crown as perhaps the best implementation of high dynamic range rendering I've experienced to date*" – John Linneman – Digital Foundry
- Was a blended HDR buffer
    - 50% Machine Learning algorithm from Redmond ATG group
        - ML was used to train an inverse tone-mapper for colour-space conversion
    - 50% Reinhard buffer
- There was a lot of manual tuning of HDR buffer ranges as well as VFX and Lighting assets needed to achieve the ideal visual result. Back and forth between Engineering, Tech Art, and Art.
- Assets were not authored in Wide Color Gamut.
- HDR Calibration screen is just as important as the HDR implementation itself

Real-time cinematics

- Had to essentially optimize a real-time feature film to run at 30fps on current gen hardware.
- Strict GPU budgets were enforced. CPU was usually OK.
- Xbox One X/PC has numerous additional visual passes for cinematics the Xbox One couldn’t support.
  - Lens Flares
  - Screen Space Reflections
  - High quality Depth of Field
  - Refraction
- No loading screens in-game, so needed level streaming without hitches during cinematics
- Seamless transitions from cinematics into game.
- Built using Matinee – Sequencer was too large of a spot-integration, 4.11 only has Matinee.

UNREAL ENGINE THE COALITION

Performance: 60 frames per second

- Performance firsts for Gears 5:
  - 60 fps campaign at 4k resolution
    - Didn’t want to do the compromise of 4k *or* 60fps/1080p which Gears 4 had.
  - Horde mode pushed to 60fps on Xbox One (was 30fps for Gears 4)
  - Escape mode at 60fps on Xbox One
  - 3 player split screen at 30fps
- Philosophy: Try to not compromise visual quality in any noticeable way to hit 60fps.
- Needed to balance performance across GPU, Draw, and Game Thread for all platforms.

Performance Tools

- Xbox One X the lead performance testing SKU for campaign
- Xbox One the lead performance testing SKU for MP
- QA would play through the levels and their performance results would populate our internal Stats page we called “Watchtower”
- We primarily would track our “percent at 60fps” but would also look at things like “average resolution”
- We had spike tracking stats as well: large, medium, and small spikes
- We tracked what thread would bound us by what % of time (ex: draw thread bound: 6%)
- Set performance targets for our various Visual Levels:
    - Visuals 1 (grey box): 70% @ 60fps
    - Visuals 2 (partial art): 80% @ 60fs
    - Visuals 3 (all art is in): 90% @ 60fps
    - Blitz (polished art): 99%+ @ 60fps
- This process combined with our up-front real-time performance stats was quite effective.

Temporal Upscaling and Async Post Processing

- Integrated UE4’s Temporal Upscaling
    - Allowed us to lean into resolution scaling without concern of the image becoming soft.
    - Can have noticeable “sizzling” below 66% scale – but overall the quality is incredible.
    - 1440p Temporal Upscaled to 4k looks a lot better than 1800p with standard upscale to 4k.
- Implementing Async Post-Processing
    - Given we were always running our Post Processing at native resolution, this made it quite expensive – so it was a natural choice to run this expensive pass async over the next frame.
    - Man-years of work to convert all post process passes to Compute and refine the async post process pipeline.
    - Overall post process async got us back roughly 1.3ms, depending on the platform.

UNREAL ENGINE

THE COALITION

Game Thread

- Largely engineering led effort with some tech art support
- Extensive threading work on various systems
- Extensive animation system optimizations
- Dozens of secondary systems optimized
- Switched from PhysX to Havok
- Tech Art support for VFX (particles) optimizations
  - Reduced max particle counts
  - Converted as many systems as possible to Swift Particles or GPU & GPU Spawn
- Tech Art support for collision mesh optimizations
- Code-ified any Blueprints that were noticeable cost or extensively used.
- Tuned the texture streamer per game mode.

Draw Thread

- Setup shadow casting draw call budgets and enforced
- Implemented a “shadow in shadow” computation pass that would occur with a lightmass bake which would disable shadow casting for any objects in shadow from the directional light only.
- Reduced particle emitter counts
- Integrated the very latest HLOD tools from Epic. Did an HLOD generation pass for most of our outdoor areas. This became one of the trickiest tech art challenges on Gears 5.
- Used the HLOD mesh as a Shadow Caster even if the HLOD wasn’t being rendered.
  - This saved us additional shadow casting cost on the Draw Thread and GPU
  - If it created visual artifacts, we could disable the HLOD shadow for that cluster or remove the problem asset from that cluster.
- Used HLOD for occlusion testing
- Implemented custom tile based occlusion culling system for Escape mode.
- Implemented on-screen draw call counts with categories
- Characters all used “Proxy LOD’s” for their lowest LOD.

GPU

- Tech Art owned the GPU budgets (and draw calls) and enforced per game-mode budgets.
- Implemented in-game on-screen budgets that were "live" to better identify performance spikes.
- Refactored numerous base material and material functions from Gears 4 to be more efficient (ex: Height Lerp blend)
  - Move as much to the vertex shader as possible
- Leveraged Rendering team for numerous GPU optimizations such as:
  - HDR optimizations
  - Depth bounds test on cascades
  - Ray Traced Shadow Optimizations
  - SSAO Optimizations
- Also cherry-picked optimizations from Epic
- Added ability to disable/enable RCA's and Skylight reflection in PPV to further balance GPU performance.
  - For outdoor environments just a skylight reflection was usually sufficient.
- Constantly balanced performance of Volume Fog, SSR, and RCA's using PPV's across campaign.
  - This can all be done using level script but we prefer PPV's for replication reasons.

Split Screen

- Xbox One X mostly took care of itself for split screen, as we lowered it to 30fps for 2p and 3p.
- Xbox One however was a fair amount of work to achieve results.
- For 2pss on Xbox One we disabled some post process passes such as motion blur, refraction, lens flares, etc as we had mostly just some GPU issues.
    - This was because we had Draw Thread performance headroom on Xbox One because we had a 60fps Xbox One X campaign.
- For 3pss on Xbox One we were heavily Draw Thread bound.
    - Dropped to 1 cascade to reduce draw thread overhead.
    - Disabled all vegetation
    - Had level artists do a “detail object” tagging pass across the campaign to hide anything that didn’t have collision on it that wasn’t required.
- Takeaway: If we hadn’t made the campaign 60fps on Xbox One X I don’t know how we would have delivered 3 player split screen on Xbox One.

PC

- Much more scalable game than Gears 4
    - Mostly dynamic real-time shadows
    - Tessellation Shaders
    - HLOD representation of environments
    - Volume Fog
    - Screen Space Reflections (Gears 4 had this)
    - Allowed our “Ultra” graphics settings to really shine
- Shipping game on Steam exposed us to a lot larger variety of PC configurations than we had seen before on the Windows Store for Gears 4. A lot of work to land that.
- Having a smaller texture pool on Xbox One helped scale the game down to lower end PC’s overall.
- Created a texture group just for Low and Medium end PC’s where we could force specific environment textures to be higher resolution just for those platforms that QA identified as too low quality.

Content Memory usage

- Overall goal of compressing texture pool 350mb from Gears 4 to make room for additional new tech and art systems. Gears 5 texture pool was 1150mb on Xbox One. Gears 4 was 1500mb.
  - 150mb HLOD to reduce draw calls
  - 30mb Volumetric Lightmaps
  - 20mb Ray Traced Shadows
  - 50mb Async Tessellation
  - 100mb needed for additional asset density, variety, signed distance field representation
  - 25mb of pool needed for Detail Mapping
- Added a real-time memory tracking system to the game called Malloc Tracker that displayed real-time numbers with budgets

Content Memory Texture Reduction

- Made room by compressing texture memory with aggressive texture packing:
  - Improved texture packing. Gears 4 had a separate texture for smoothness, metalness, and AO. Eliminated this texture and incorporated it into the Normal Map thus making a BC7 texture (instead of BC5) that was:
    - R: Normal X
    - G: Normal Y
    - B: Smoothness
    - A: Metalness
  - This allowed us to reduce our texture footprint by 10% - 40% per material, taking further pressure off our texture pool and disk footprint. Gears 5 was *big* compared to 4. We needed space everywhere we could get it!
  - Combined with the elimination of Shadow Maps - our materials were almost always Vector ALU bound on Xbox not Texture Bound like Gears 4 improving GPU performance.

Conclusions

- Was able to maintain a “no compromise” approach to frame rate and visuals all the way through shipping.
  - Made some educated bets
    - 60fps is the new “base line”. Line in the sand early.
    - Smart optimizations (what does the gamer actually notice?)
    - Visual Investments that paid off
  - Had a clarity of vision after Gears 4 was completed.
    - Took learnings from Gears 4 Xbox One X
      - Pixel density and detail maps
      - X1X strengths and weaknesses
      - HDR
      - 60 fps
  - Had a little bit of luck:
    - Temporal Upscaling
    - Xbox One X power
    - Epic continuing to improve UE4 in parallel
- Very proud of the product.
  - Massive team effort across over a dozen support studios and hundreds of people

Thanks!

- The Coalition Art, Tech Art, and Engineering Teams
- Epic for continuing to improve Unreal Engine
- Splash Damage
- Behaviour
- Disbelief
- AMD
- Rod Fergusson
- Mike Rayner
- Christi Rae
- Aryan Hanbeck
- Jamie McNulty
- James Sharpe
- Geoff Lester
- Brad Sweder
- Ian Wong
- Christopher Wallis
- David Lucas
- Jerry Edsall
- JP Flouret
- David Bollo
- Simon Wong
- Miles Kaech
- Samuel Cloutier
- Tim Wang
- Mike Perzel
- Bruno Melo De Souza
- Andy Koo
- Malcolm Andrieshyn
- Anthony Marraffa
- Ryan DowlingSoka
- David Coleman
- Tony Wang
- Cam McRae
- Aeva Palecek
- Steve Anichini
- Aurel Cordonnier
- Robert Millar
- Claude Marais
- Cindy Wong
- Phil Cousins
- Nick Christiani
- Quentin MacMillan
- Luis Sandrin
- Stu McKenna
- Dan Phillips
- John White
- Brian Karis
- Guillaume Abadie
- Stuart Maxwell
- Keswick Allen
- Eugene Slautin
- Amy Pejic
- Miles Kaech

THE COALITION

We’re hiring!

- Senior Rendering Software Engineer
- Senior Services Software Engineer
- Senior Narrative Designer
- Art Director (Multiplayer)
- Lead Mission Designer
- Senior Lighting Artist
- Senior Level Artist