Lima Oscar Delta!

SCALING CONTENT IN CALL OF DUTY: MODERN WARFARE

Agenda

- Intro
- Motivation
- Offline LOD processing
- Model packaging
- Runtime LOD processing
- World proxy LOD’s
- Conclusion

Intro

Intro


CALL OF DUTY
MODERN WARFARE
MATURE 17+
M
ESRB


TONY HAWK'S UNDERGROUND 2
CALL OF DUTY INFINITE WARFARE
GUITAR HERO WORLD TOUR
GUITAR HERO METALLICA
BAND HERO
65 SONGS BY THE BIGGEST POP ARTISTS
PlayStation.2
TONY HAWK'S AMERICAN WASTELAND
PlayStation.2
TONY HAWK'S UNDERGROUND
TONY HAWK'S PROJECT 8
GUITAR HERO 5
85 SONGS BY 83 ARTISTS
PLAYSTATION 3
XBOX ONE
CALL OF DUTY GHOSTS
PROVING GROUND
PLAYSTATION 3
GUITAR HERO III LEGENDS OF ROCK
XBOX 360
GUITAR HERO WARRIORS OF ROCK
CALL OF DUTY ADVANCED WARFARE
HERO
AEROSMITH

Intro

- Modern Warfare
- F+ renderer
- Game units = inches
- Frontend geometry processing on the GPU [Aaltonen/Haar 2015]
- Historically, streaming was limited to textures and audio
- Pushing content to the limits of modern consoles is the result of many technology and artistic efforts. This is just one of them.

Intro

Intro

Global level optimization

- Offline crunching of all per-level lighting & geometry
- Game rendering != tools rendering

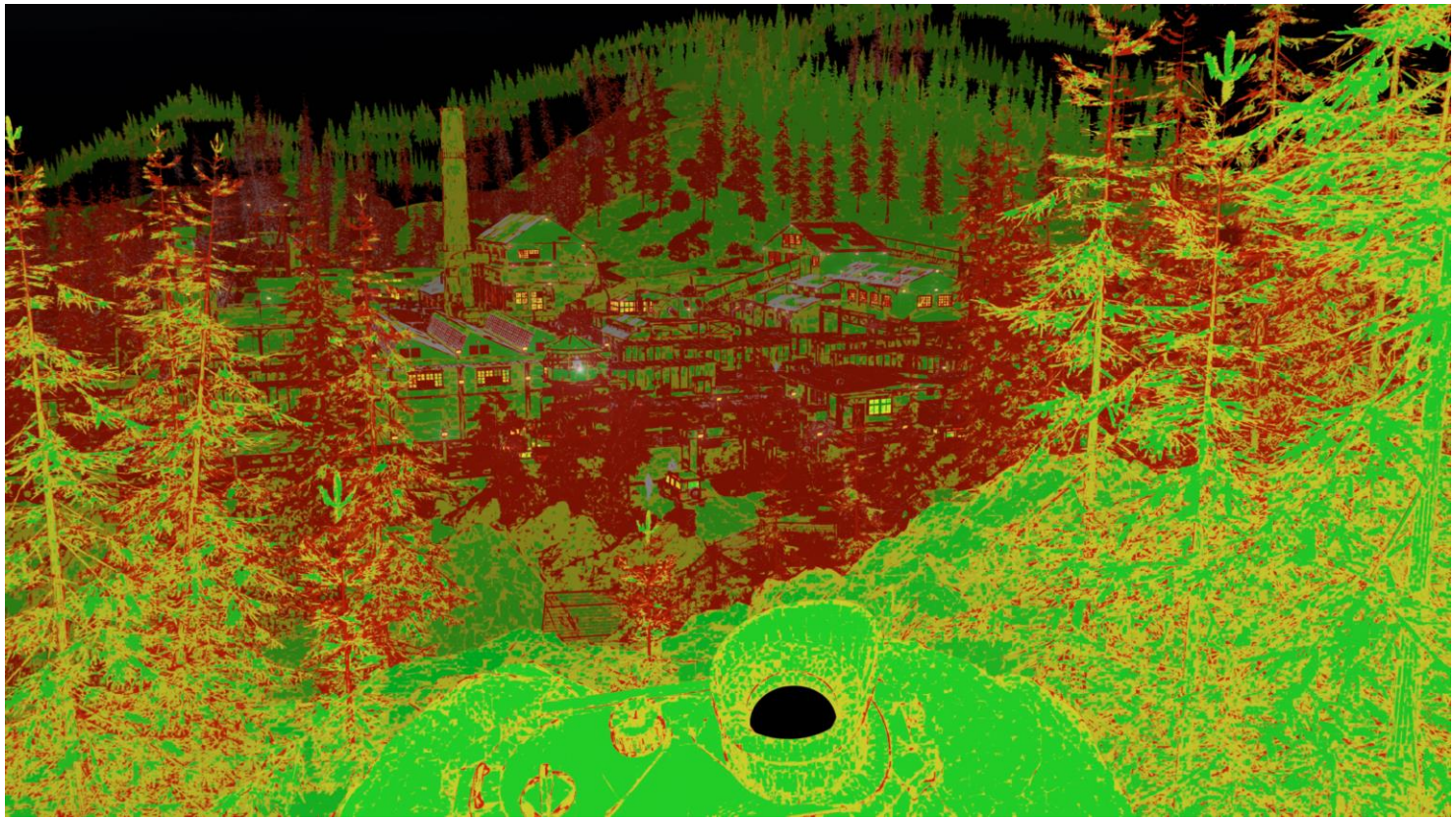
Motivation

Motivation

Call of Duty: Infinite Warfare (2016)
1.7m tris
2.1k models

Call of Duty: Modern Warfare (2019)
3.8m tris
1.2k models

Motivation



GCN Quad efficiency [Drobot14]

Motivation

Streamlining asset creation

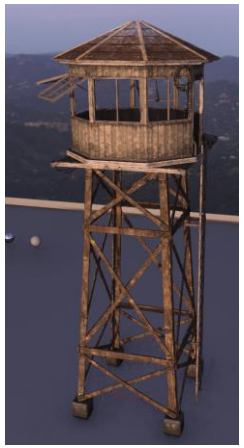
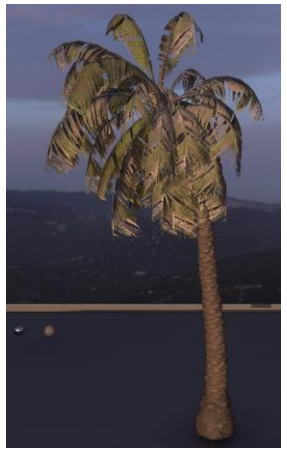
Motivation

High visual bar across all platforms

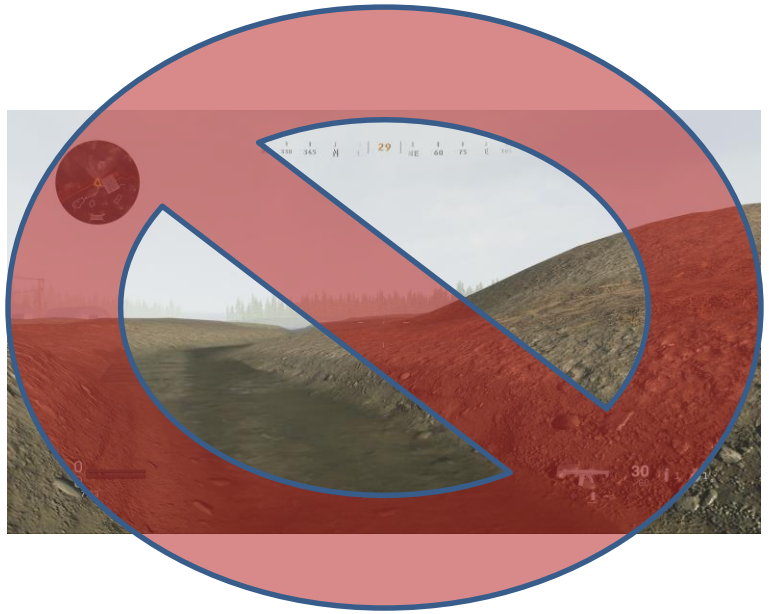
Level of Detail: Offline

AKA: INFANTRY TRAINING

LOD: Offline





LOD: Offline

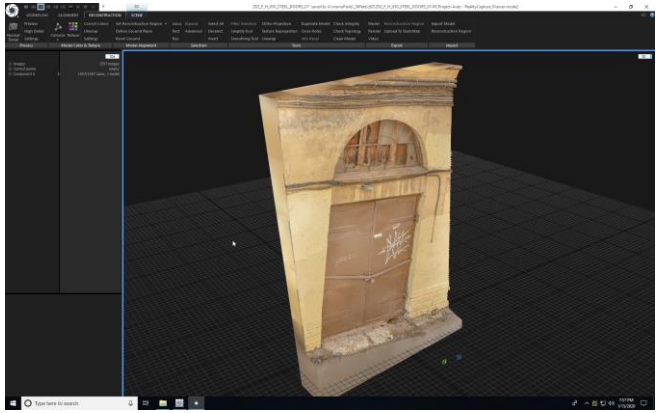
Photogrammetry: Photos

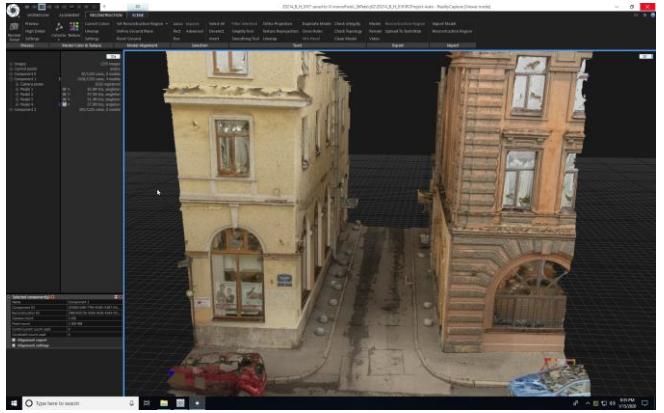


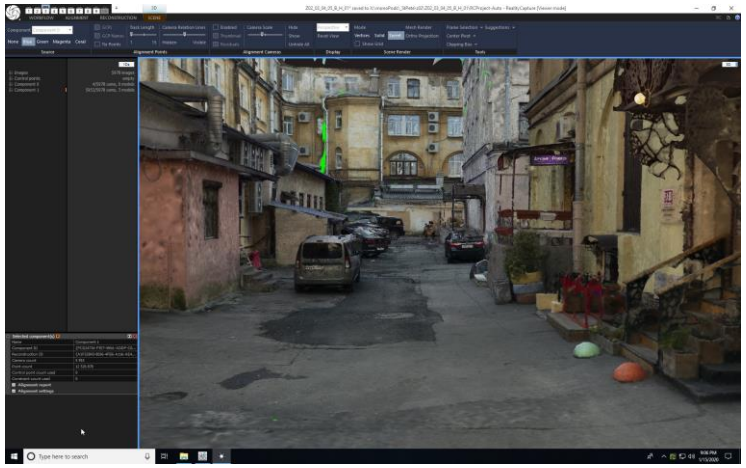


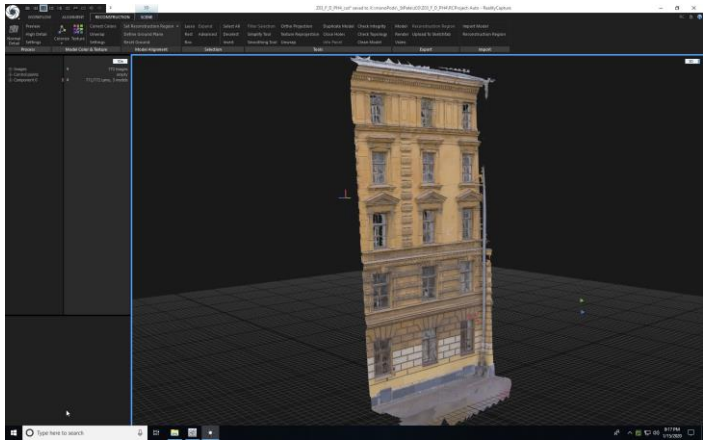
LOD: Offline

Photogrammetry: Hi-res scans (>50M tris)







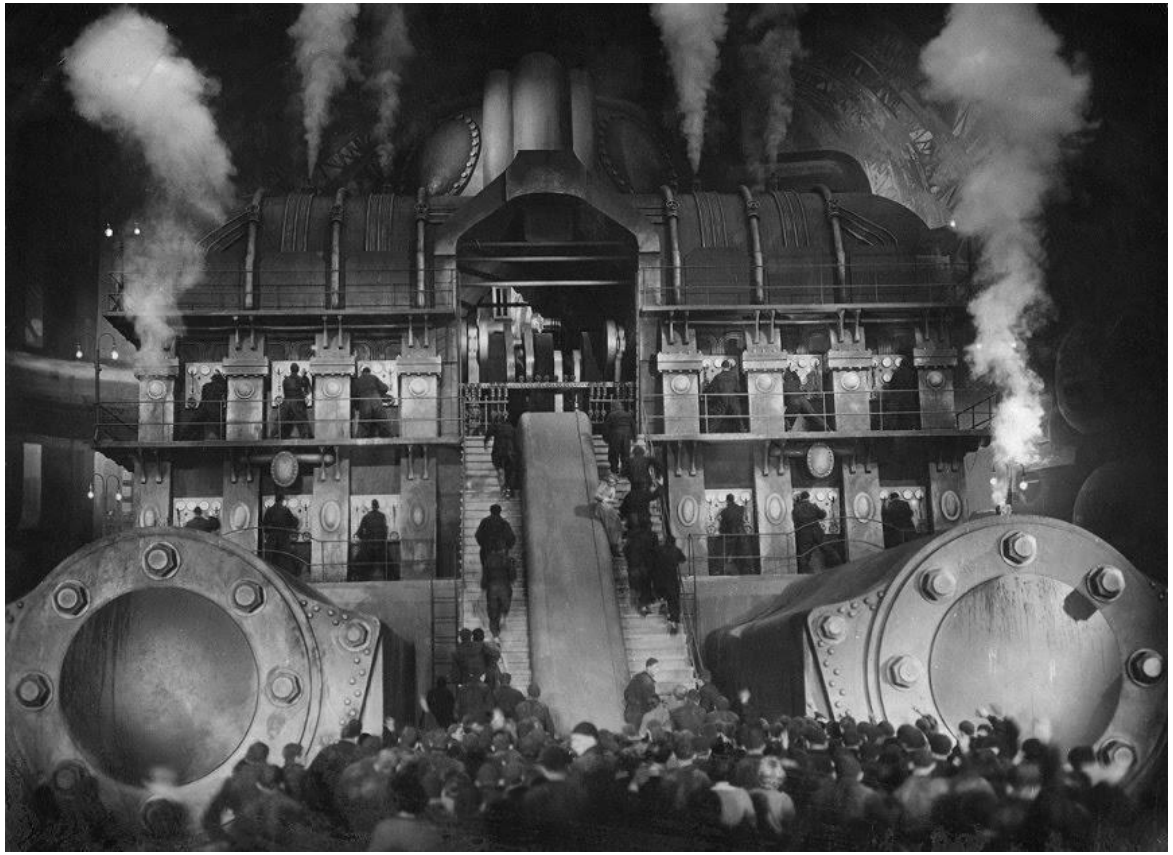


LOD: Offline

Photogrammetry: In-game Assets

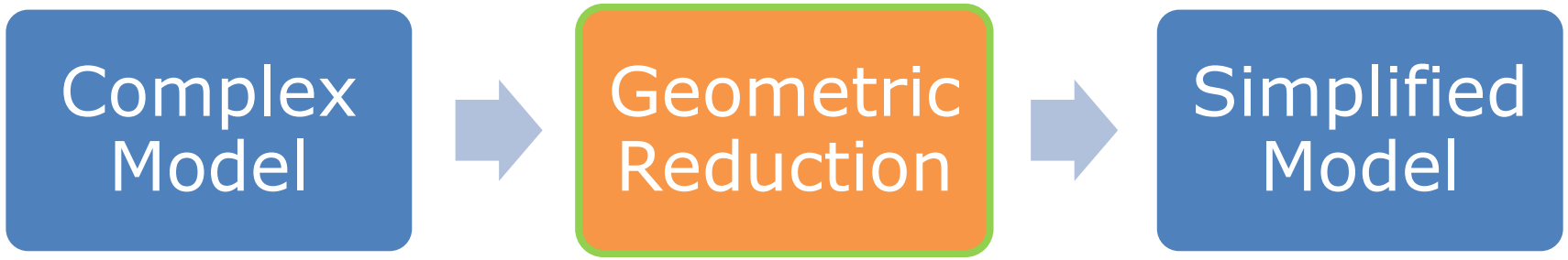
LOD: Offline

Automate the rest



LOD: Offline

First part: Figure out a solution for general purpose geometric reduction (aka. “the easy part”)



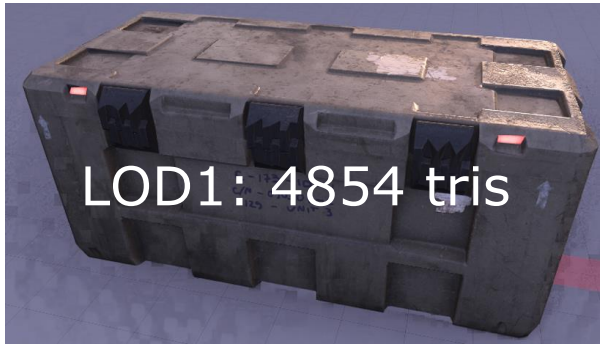
LOD: Offline

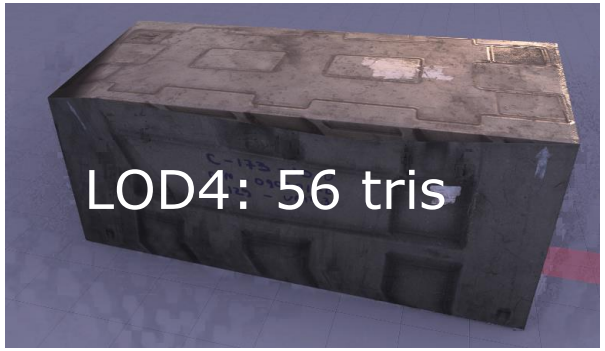
Second part: Everything else

LOD: Offline

How many LOD levels do we want?

- Support up to 5x additional discrete entries
- Wasteful to use all slots for all models.





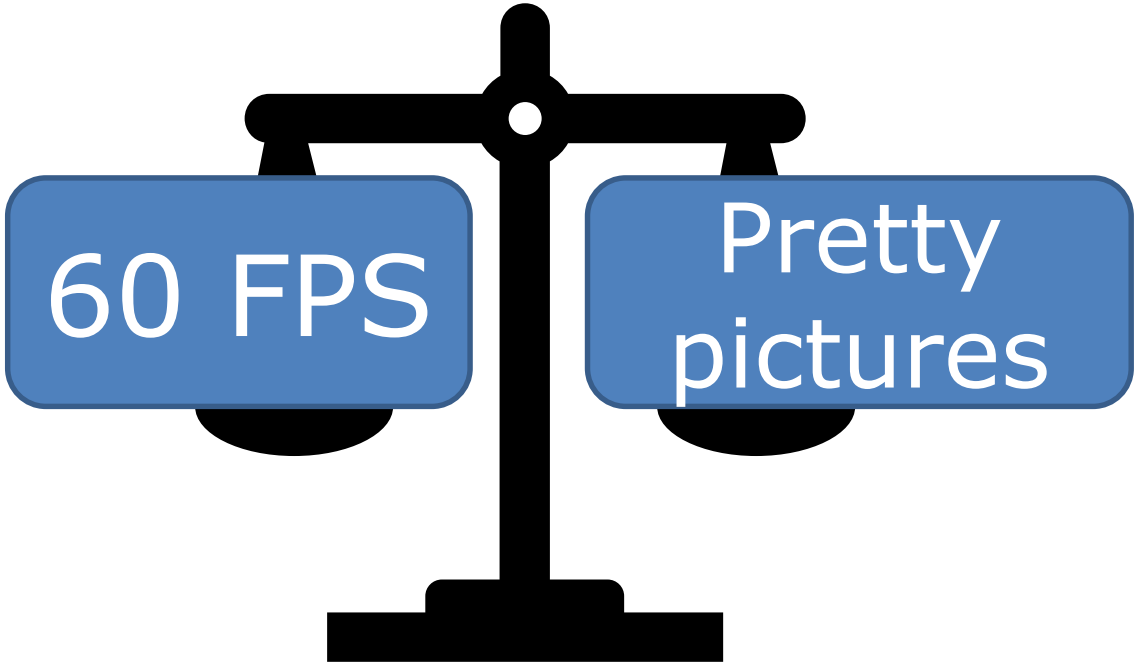
LOD: Offline

What are our reduction targets?

- Verts/Tris
- Normal Variances
- Silhouette retention
- Stream important weights
- …

LOD: Offline

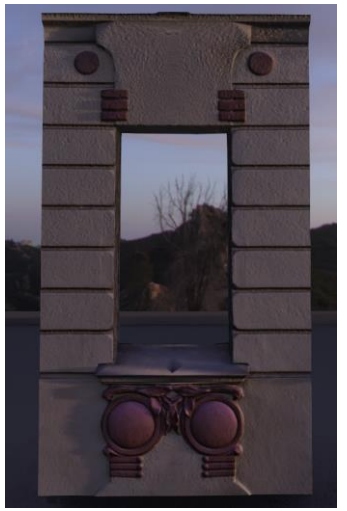
When should we switch between LOD's?



LOD: Offline

How will this model be used in-game?

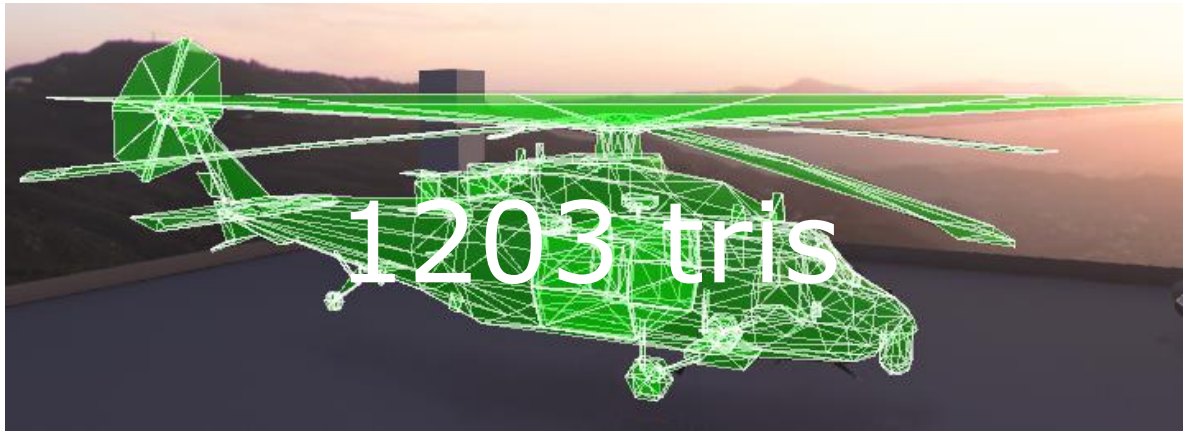
- Hero character
- Simple prop
- Periphery
- View model
- Structural
- …



LOD: Offline

What about collision?

- Match a render LOD or something different



LOD: Offline

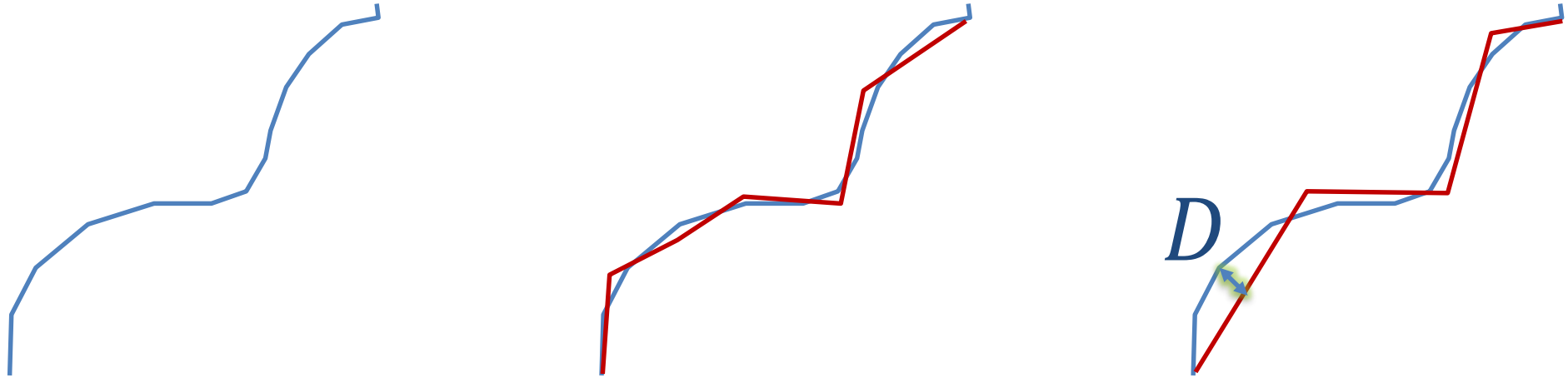
Is it possible to optimize for all of these?

- LOD Count
- Reduction Targets
- Switch Distances
- In-game context
- Collision

LOD: Offline

First attempt: automate everything!

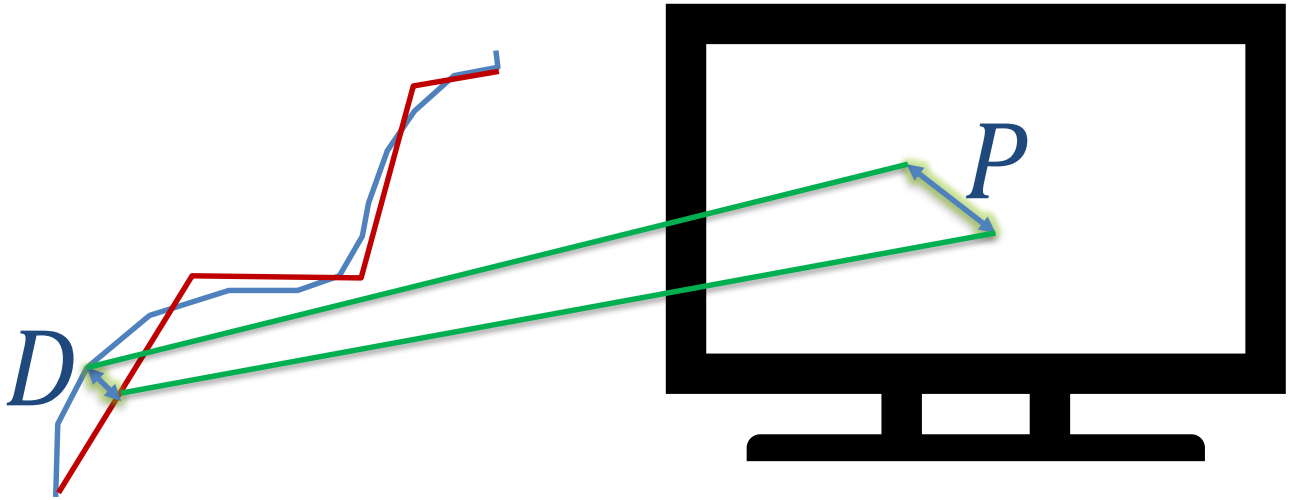
1. Reduce mesh until its geometric deviation passed a certain threshold.



LOD: Offline

First attempt: automate everything!

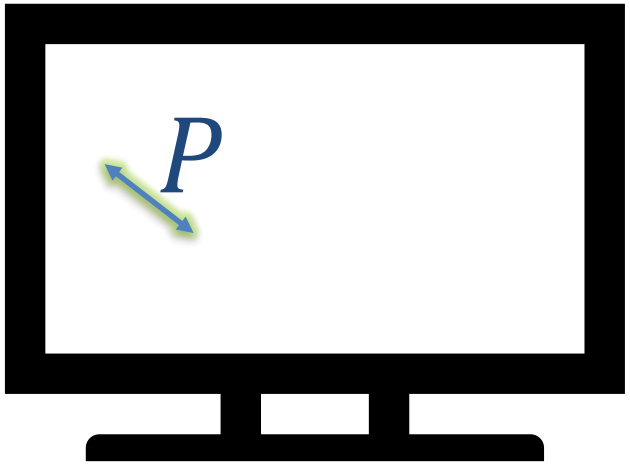
2. Calculate the switch distance at which that threshold consumes a specific number of pixels.



LOD: Offline

First attempt: automate everything!

2. Calculate the switch distance at which that threshold consumes a specific number of pixels.

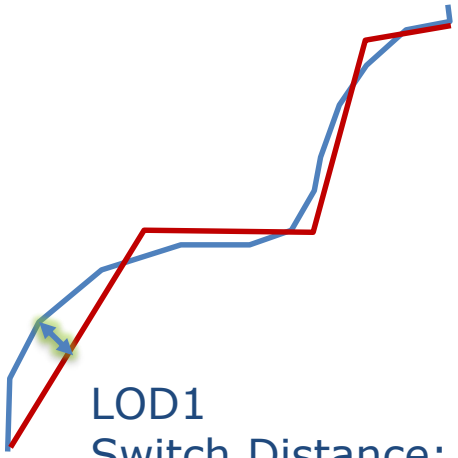


$$P = modelRadius * 2.0/D$$
$$screenRatio = P/screenSize$$
$$nearClipDist = 1.0/\tan(fovRadians/2.0)$$
$$bsphereTheta = \tan^{-1}(screenRatio/nearClipDist)$$
$$switchDist = modelRadius/\sin(bsphereTheta)$$

LOD: Offline

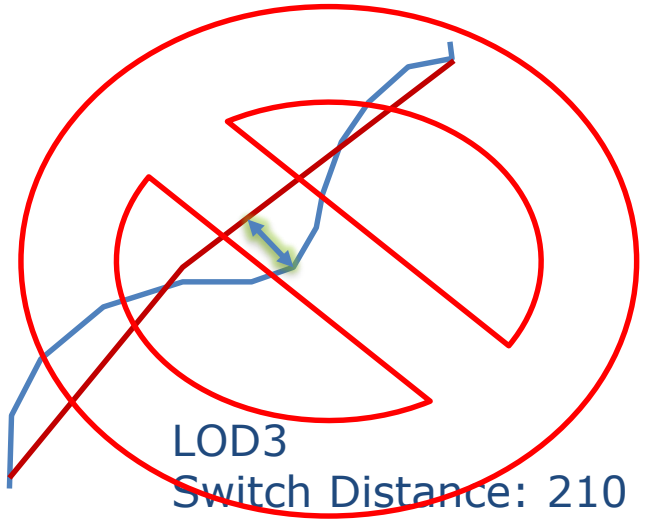
First attempt: automate everything!

3. Skip LOD's whose generated switch distance is "too close" to a previous one.



LOD1
Switch Distance: 100

LOD2
Switch Distance: 200

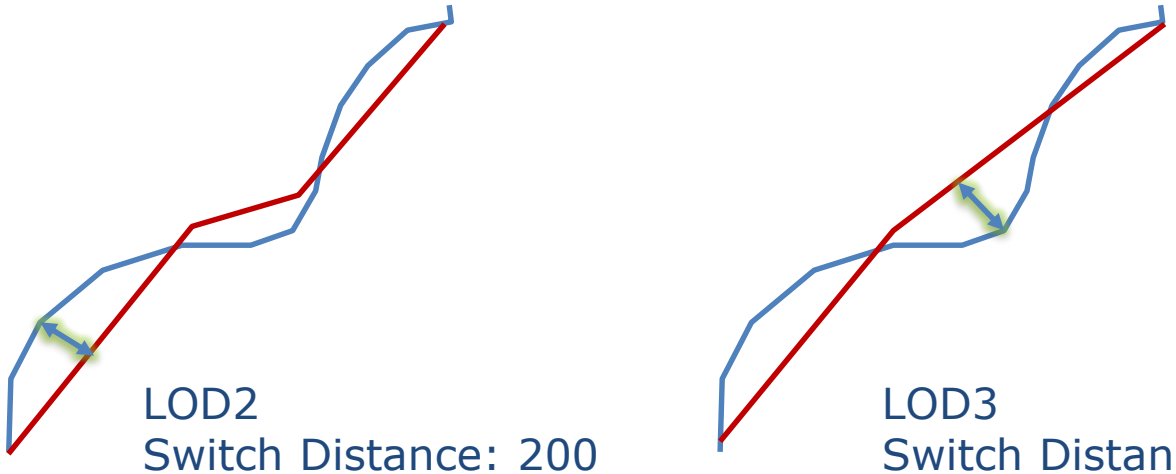


LOD3
Switch Distance: 210

LOD: Offline

First attempt: automate everything!

4. Repeat until a LOD is “too small” at a generated switch distance.



LOD2
Switch Distance: 200
Projected Size: 20

LOD3
Switch Distance: 2000
Projected Size: 2

Num LODs: 4

LOD: Offline

First attempt: automate everything!

- Made sense (to programmers).
- Deterministic perf & visuals
- Completely unintuitive to artists…

LOD: Offline

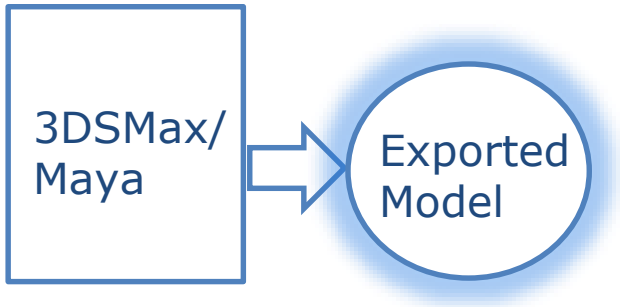
Second attempt: Hybrid solution

- Keep automatic generation of LOD settings, but only as a convenience.
- Allow artists to override pretty much everything, if desired.

LOD: Offline

Second attempt: Hybrid solution

1. Export some model geometry from a DCC tool



LOD: Offline

Second attempt: Hybrid solution

2. Create a new Model Asset in our Asset Management Tool

LOD: Offline

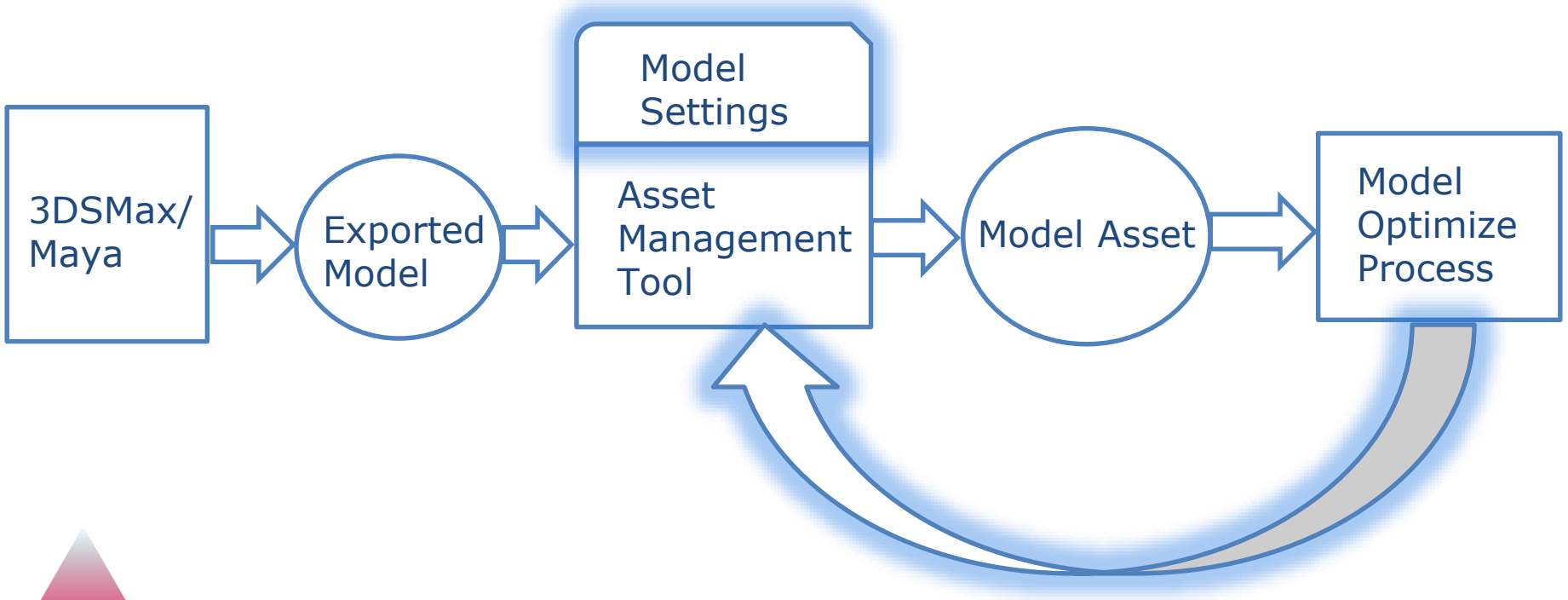
Second attempt: Hybrid solution

3. Kick off a background process that generates the “technically ideal” number of LOD’s, switch distances, etc.

LOD: Offline

Second attempt: Hybrid solution

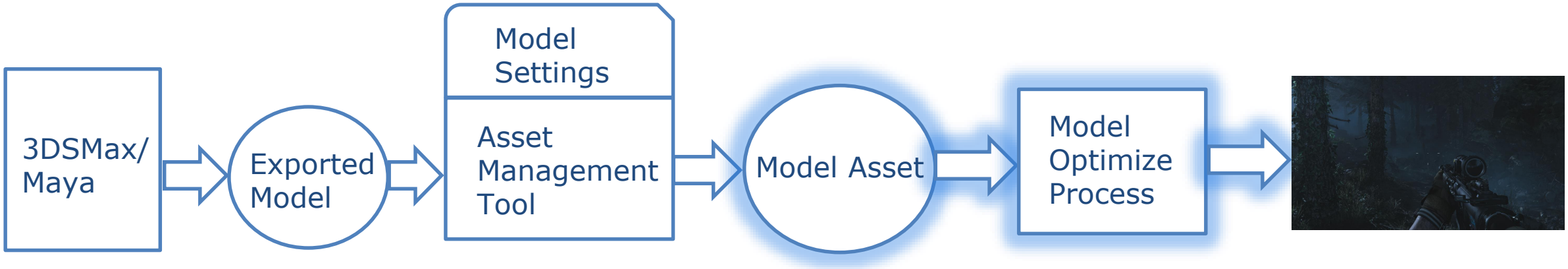
4. Use those values to seed the GUI form in the asset management tool



LOD: Offline

Second attempt: Hybrid solution

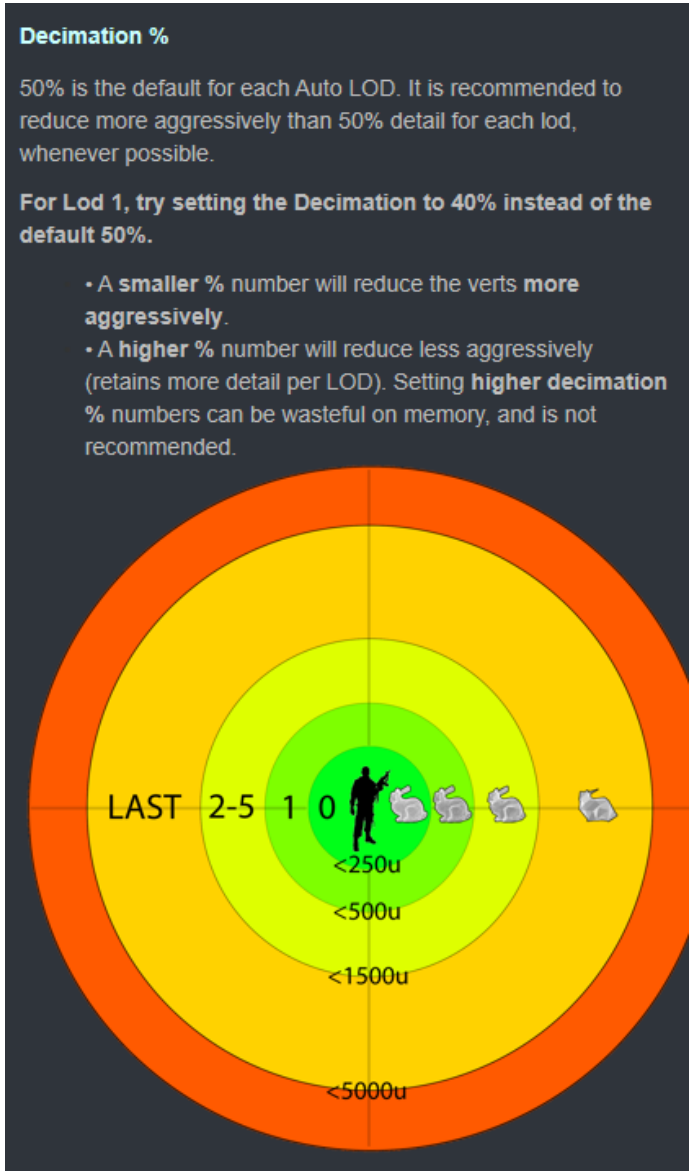
5. If the artist likes what they see, nothing more needs to be done. Otherwise, they can fine tune where desired



LOD: Offline

Second attempt: Hybrid solution

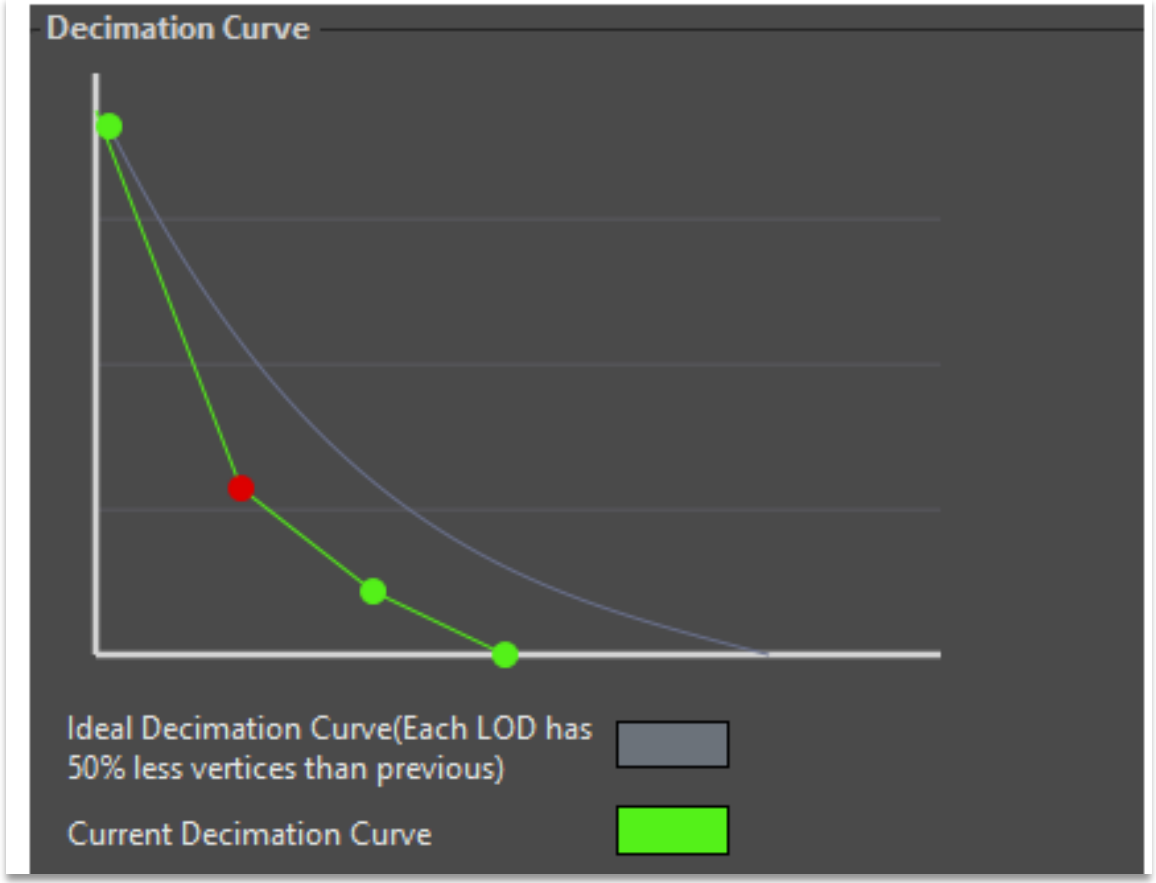
Provide visual, intuitive feedback to help gauge runtime efficiency



LOD: Offline

Second attempt: Hybrid solution

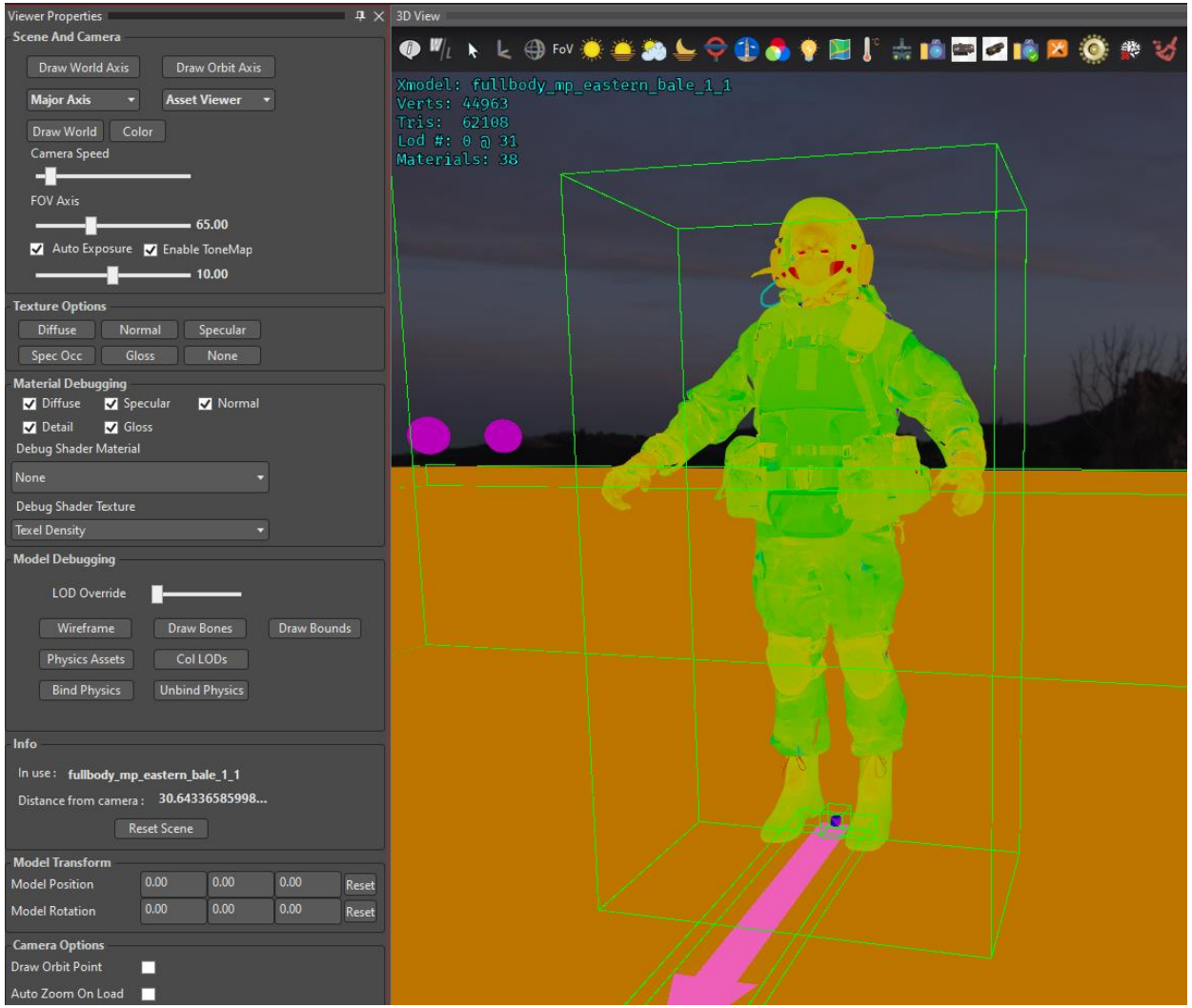
Provide visual, intuitive feedback to help gauge the runtime efficiency of a given LOD chain.



LOD: Offline

Second attempt: Hybrid solution

Curated whitebox level or tool dedicated completely to help in these tuning efforts.

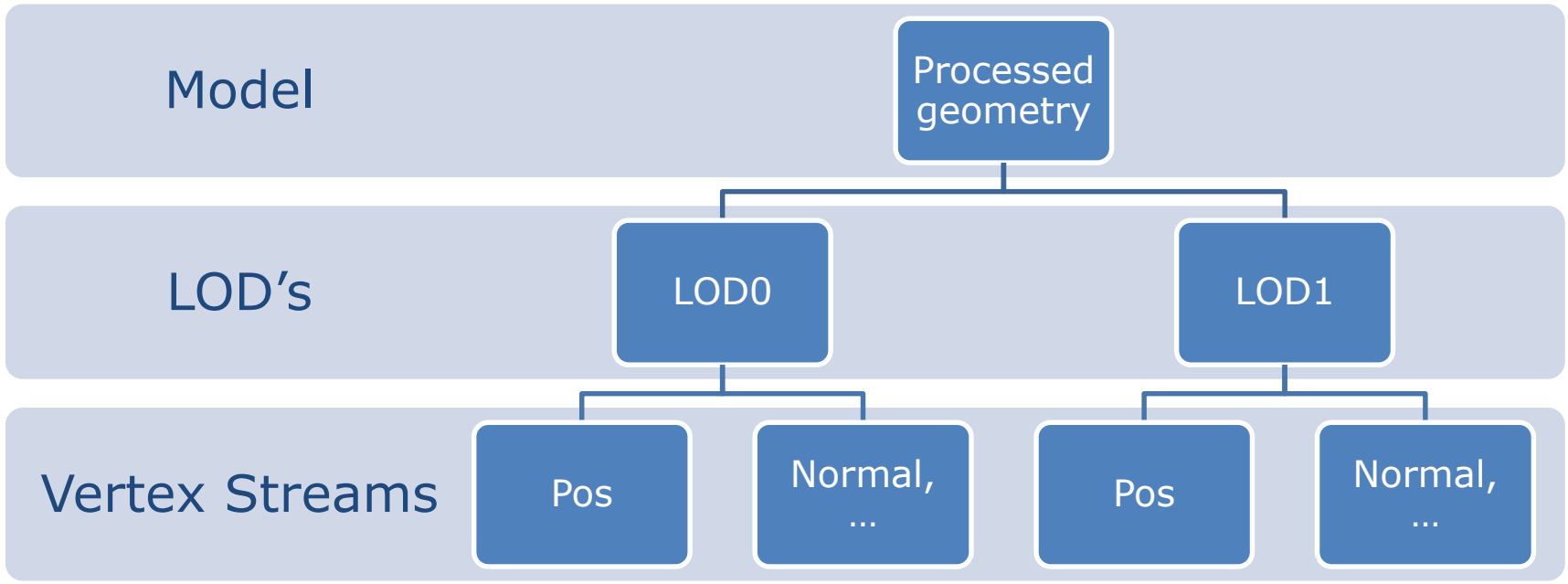


Level of Detail: Packing

AKA: GEARING UP!

LOD: Packing

Split LOD’s into individually streamable chunks



LOD: Packing

Object-space positional quantization

$$scale = \frac{(pos - boundsMid)}{boundsRadius}$$

$\text{mask} = 0\text{x}1\text{FFFFF}$ ← (21-bits per component)

$$packedPos = clamp(\,(scale * 0.5 + 0.5) * mask, 0, mask)$$

LOD: Packing

Promote rigidly animated surfaces to soft-skinned

LOD0
Bones: 16
Materials: 6
Surfaces: 16

LOD0
Bones: 16
Materials: 6
Surfaces: 6

LOD: Packing

Aggressive merging of geometry and materials

LOD0
Bones: 16
Materials: 6
Surfaces: 6

LOD0
Bones: 16
Materials: 4
Surfaces: 4

LOD: Packing

Stream Tree

Level of Detail: Runtime

AKA: OFF TO BATTLE!

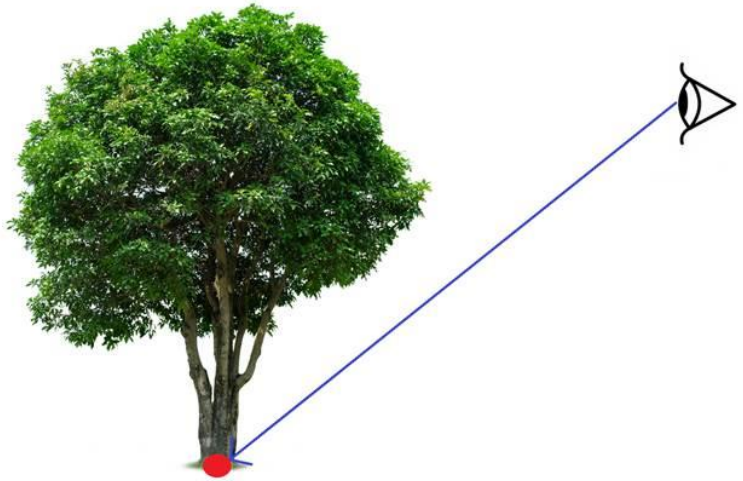
LOD: Runtime

All about the LOD selection

Use switch distances established offline?

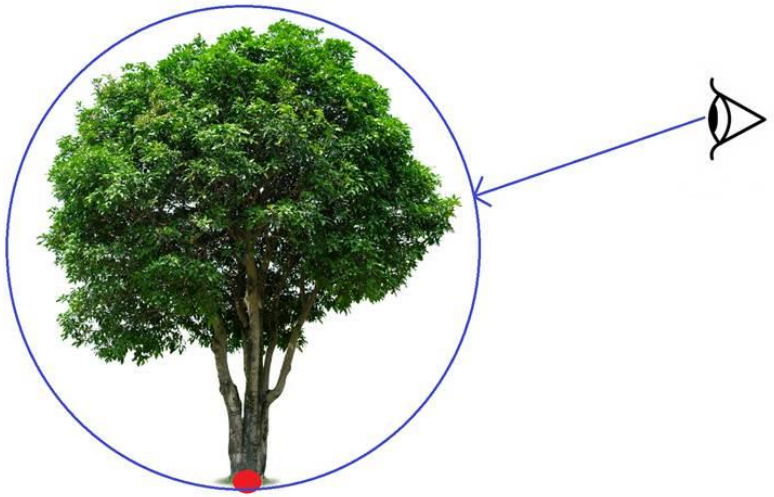
LOD: Runtime

Authored switch distances



Small objects: Distance to pivot

$$\vec{D} = length(\overrightarrow{\text{eye}} - \overrightarrow{\text{pos}})$$



Large objects: Distance to bounds

$$\vec{D} = length(max(|\overrightarrow{eye} - \overrightarrow{boundsMid}| - \overrightarrow{halfSize}, \vec{0}))$$

LOD: Runtime

LOD selection criteria: Authored switch distances


NW 330 345 N 25 NE 30 75 E
CHICKEN FARM
LB + RB
0
0
0:00.0
5
20
1
1

LOD: Runtime

Account for FOV changes

- Use vertical FOV to ensure same results regardless of display aspect ratio (4:3, 16:9, 16:10, …).

$$tanHalfFovY = \left(\frac{9}{16}\right) * tanHalfFovX$$

$$invFovScale = \frac{tanHalfFovY}{\left(\frac{9}{16}\right) * \tan(\,80.0 * 0.5)}$$

$$\boldsymbol{invFovScale = tanHalfFovY * 2.11867305}$$


330
345
N
25
NE
60
CHICKEN FARM
Orellana [RnG1]
0
0
0:00.0
5
20
1
1

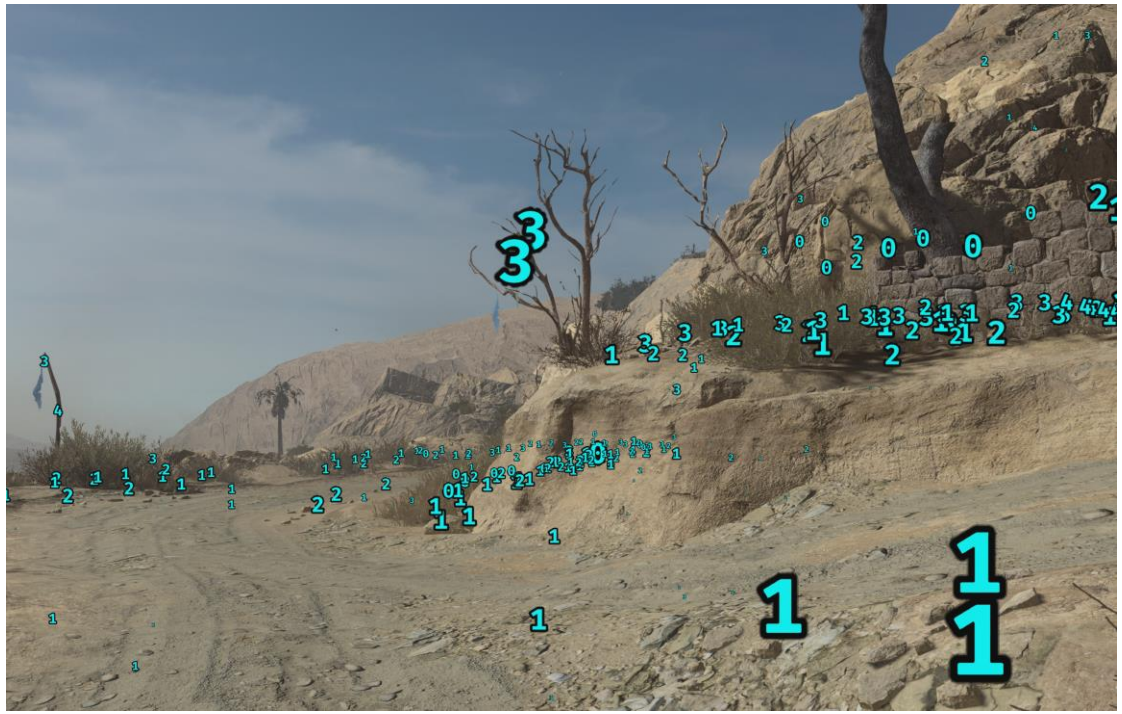
LOD: Runtime

LOD selection criteria

- Authored switch distances
- FOV scaling

LOD: Runtime

Scene resolution scaling



Displayed at: 4k
Authored at: 1080p

LOD: Runtime

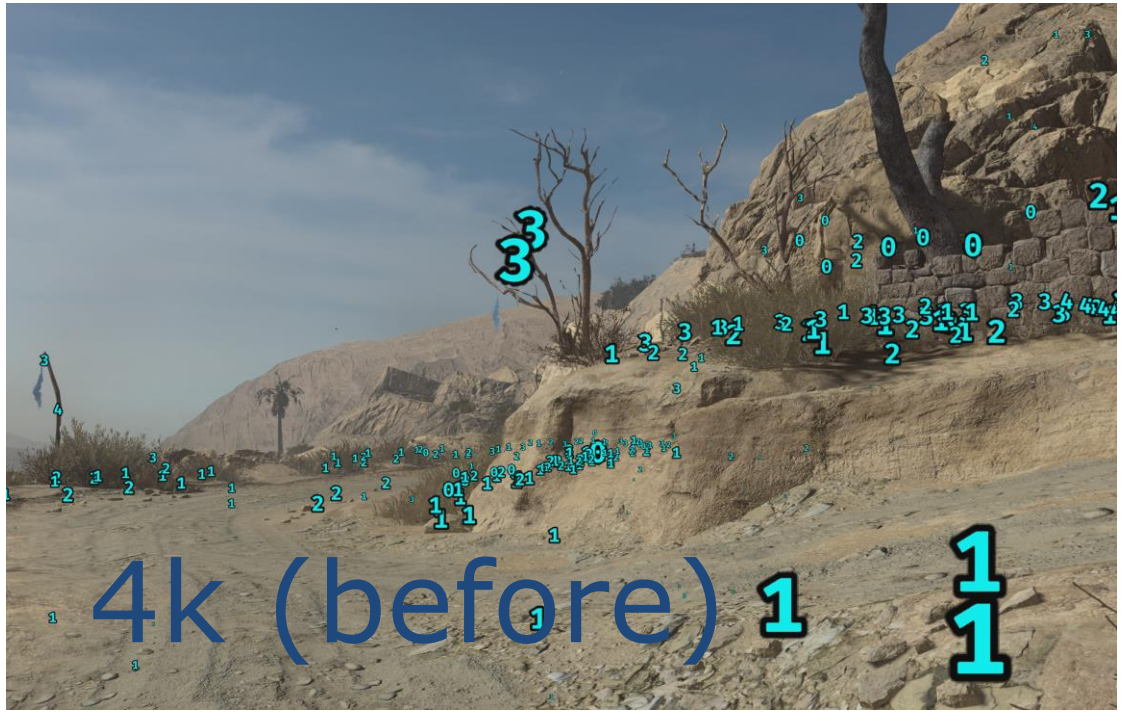
Scene resolution scaling

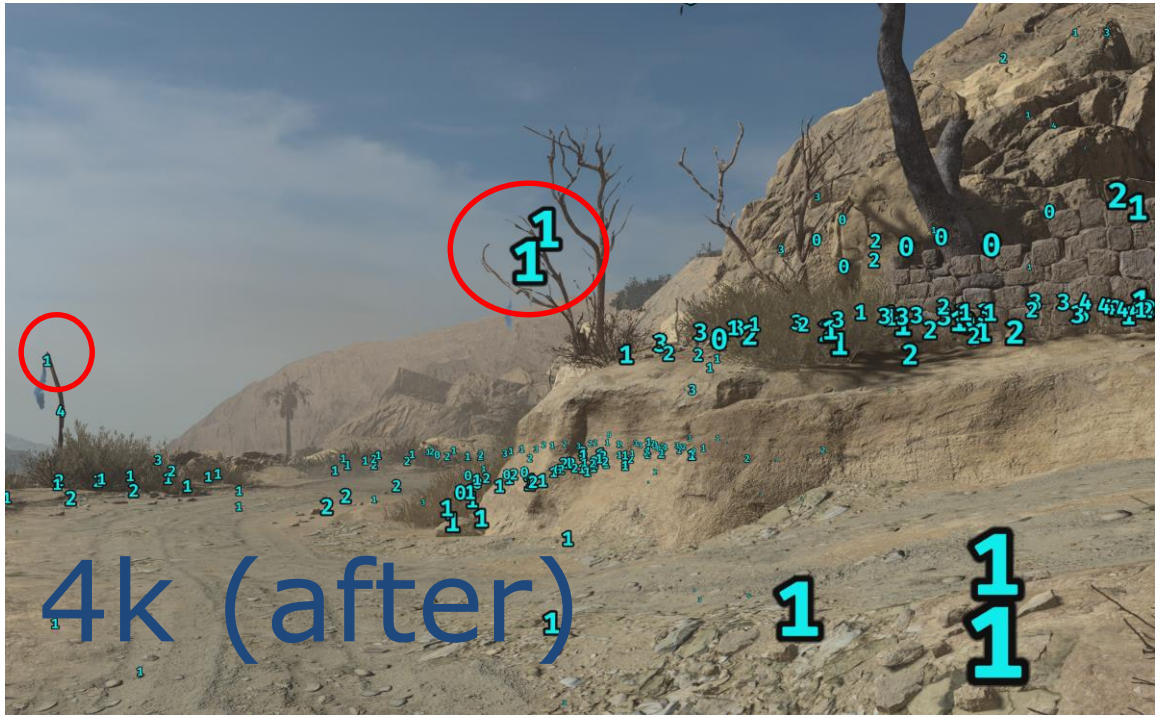
$$nearClipDist = 1.0/\tan(defaultFov/2.0)$$
$$screenRatio0 = targetScreenHeight/defaultScreenHeight$$
$$angle0 = \tan^{-1}(screenRatio0/nearClipDist)$$
$$angle1 = \tan^{-1}(1.0/nearClipDist)$$
$$\boldsymbol{sceneResScale = sin(angle1)/sin(angle0)}$$

- Calculated once at boot
- defaultFOV = 65.0
- defaultScreenHeight = 1080.0

LOD: Runtime

Scene resolution scaling





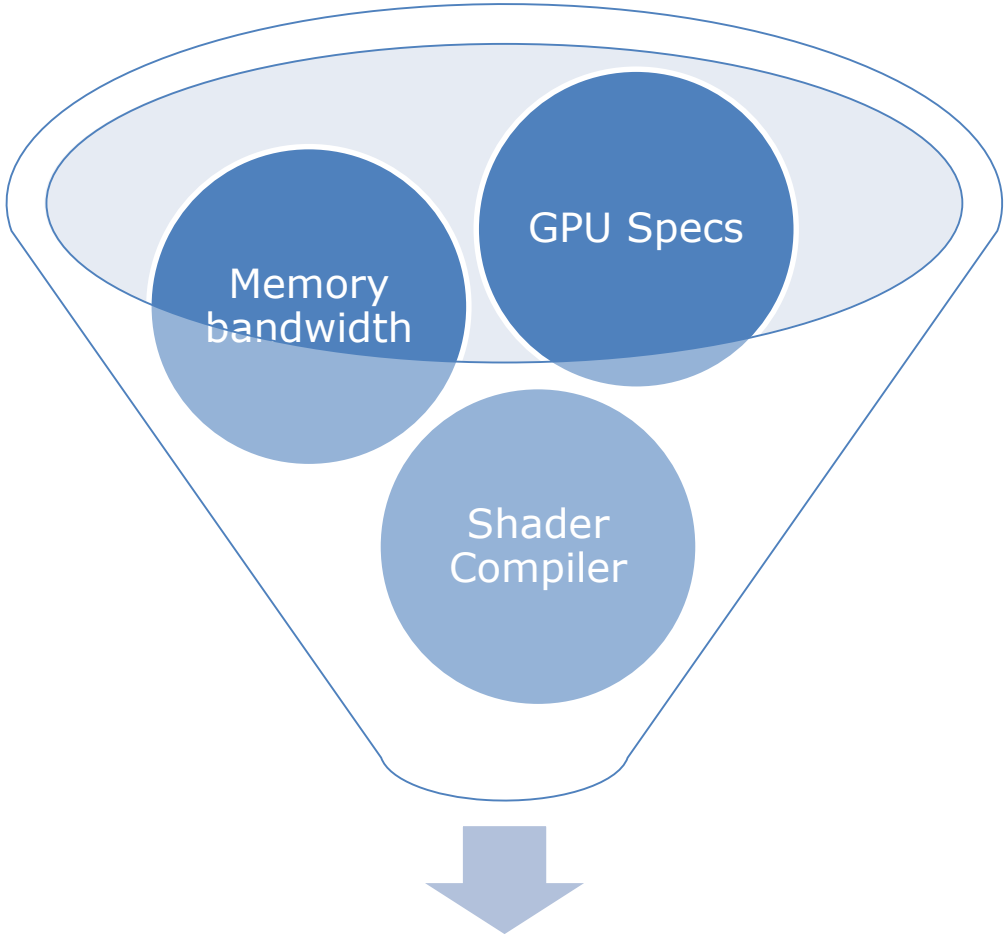
LOD: Runtime

LOD selection criteria

- Authored switch distances
- FOV scaling
- Scene resolution

LOD: Runtime

Vertex processing efficiency



$$vertexProcessingScale$$

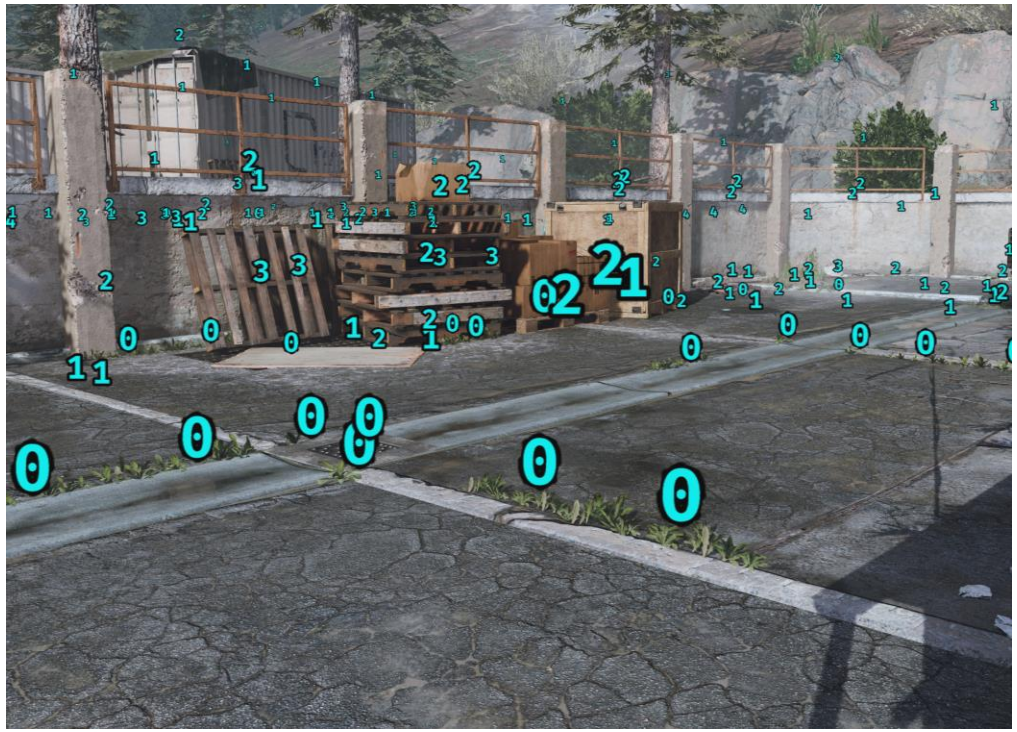
LOD: Runtime

LOD selection criteria

- Authored switch distances
- FOV scaling
- Scene resolution
- Vertex processing efficiency

LOD: Runtime

General GPU Performance

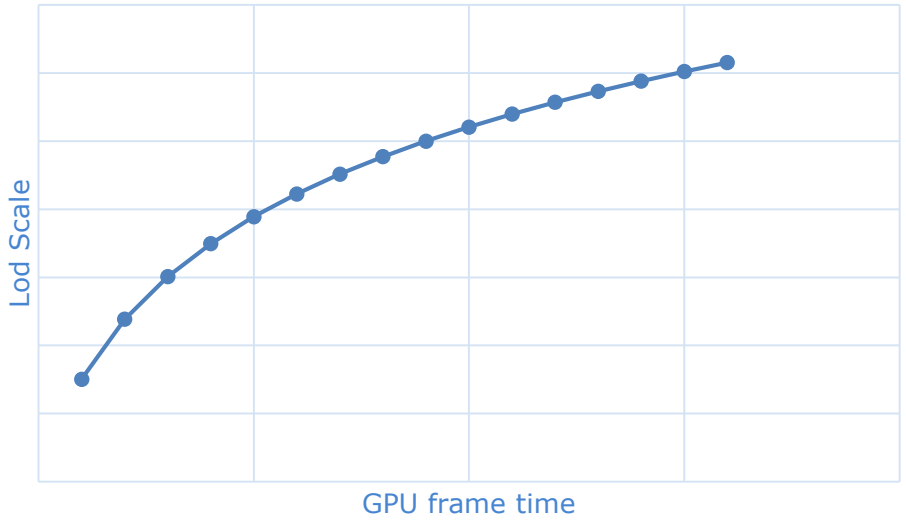


GPU: 16.72ms

LOD: Runtime

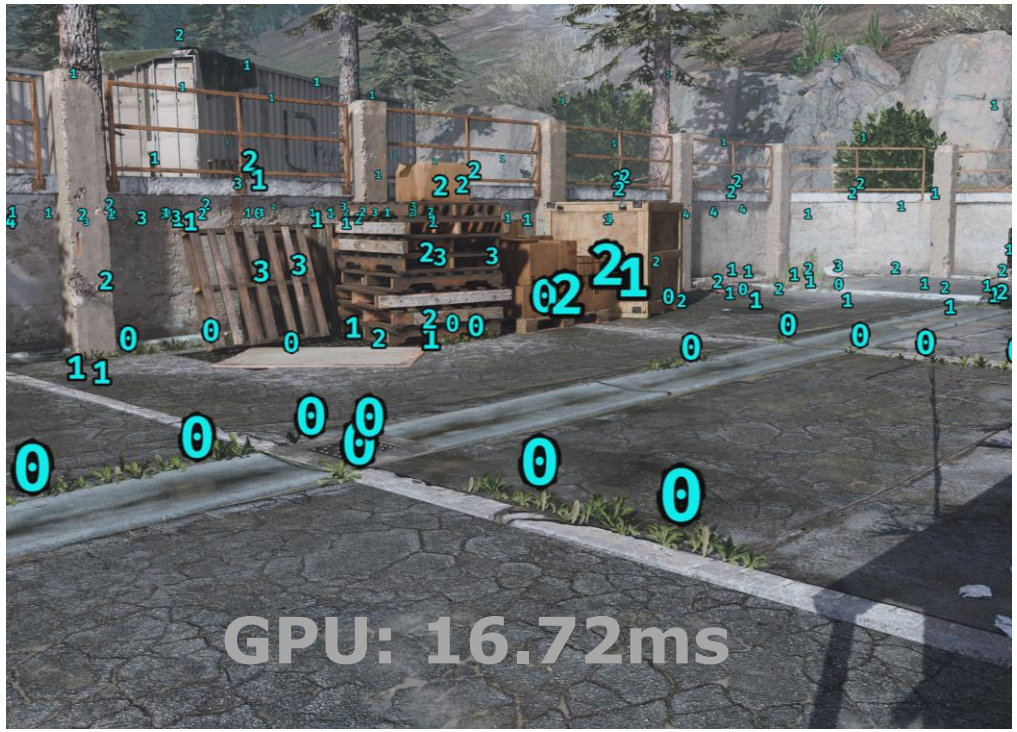
General GPU Performance

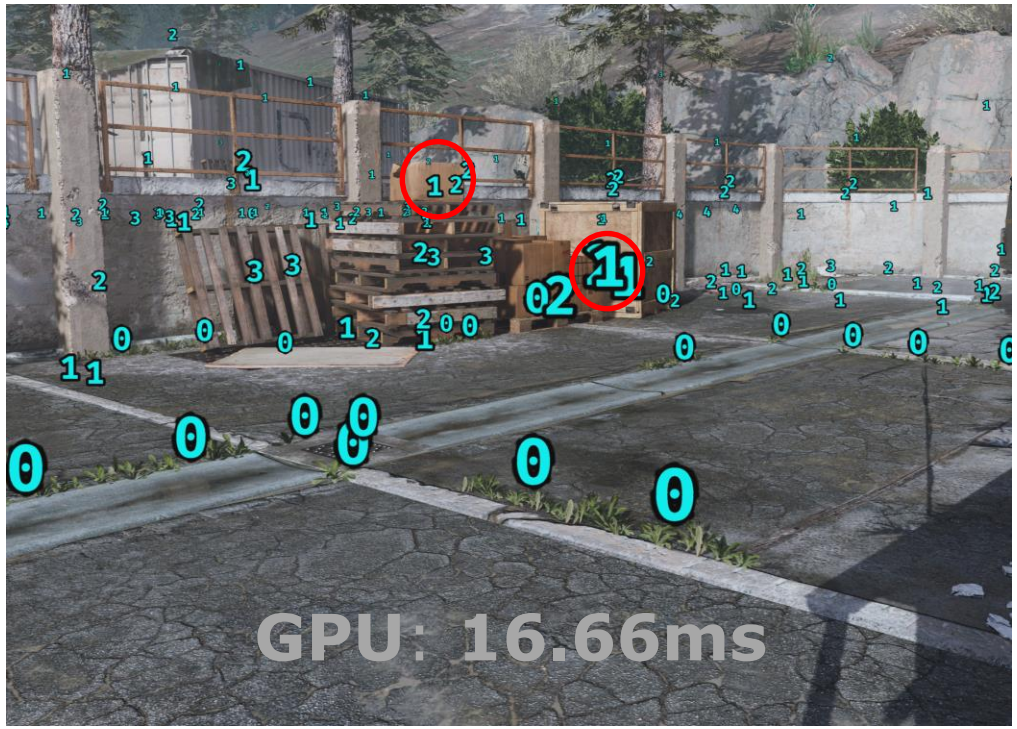
$$gpuPerfScale = mapToLogCurveWithHysteresis(gpuFrameTime)$$



LOD: Runtime

General GPU Performance





LOD: Runtime

LOD selection criteria

- Authored switch distances
- FOV scaling
- Scene resolution
- Vertex processing efficiency
- General GPU Performance


240 255 W 285 NW 330 345 N
Song [Cmp1]
Garnell [CQB1]
Hancock [RnG1]
Reyes [RnG1]
Faulk [Cmp1]
McClellan [RnG1]
Heder [RnG1]
Isibor [Cmp1]
Bright [RnG1]
Wright [CQB1]
Uvidia [CQB1]
Lincoln [RnG1]
0
0
0:00.0
30
60
1
1
LB + RB

LOD: Runtime

Budget-based biasing

- Skinned vertex limits
- Geometry pipeline limits

```
lodThreshold=[geoLimit*minPcnt, geoLimit*maxPcnt]
if ( curGeoUsage >= lodThreshold[0] )
{
    if ( geoBias == geoBiasStep )
        geoBias = 2.0* geoBiasStep
    else
        geoBias = geoBiasStep
}
else if (curGeoUsage >= lodThreshold[1] )
        geoBias = max( geoBias – geoBiasDecay*dT, 0.0 )
```

Ramp up/down Lod Bias each frame


240 255 W 285 NW 330 345 N
Song [Cmp1]
Garnell [CQB1]
Faulk [Cmp1]
Hancock [RnG1]
Reyes [RnG1]
Heder [RnG1]
McClellan [RnG1]
Isibor [Cmp1]
Bright [RnG1]
Wright [CQB1]
Uvidia [CQB1]
Lincoln [RnG1]
0
0
0:00.0
LB + RB
30
60
1
1

LOD: Runtime

LOD selection criteria

- Authored switch distances
- FOV scaling
- Scene resolution
- Vertex processing efficiency
- General GPU Performance
- Budget-based biasing

LOD: Runtime

LOD selection criteria

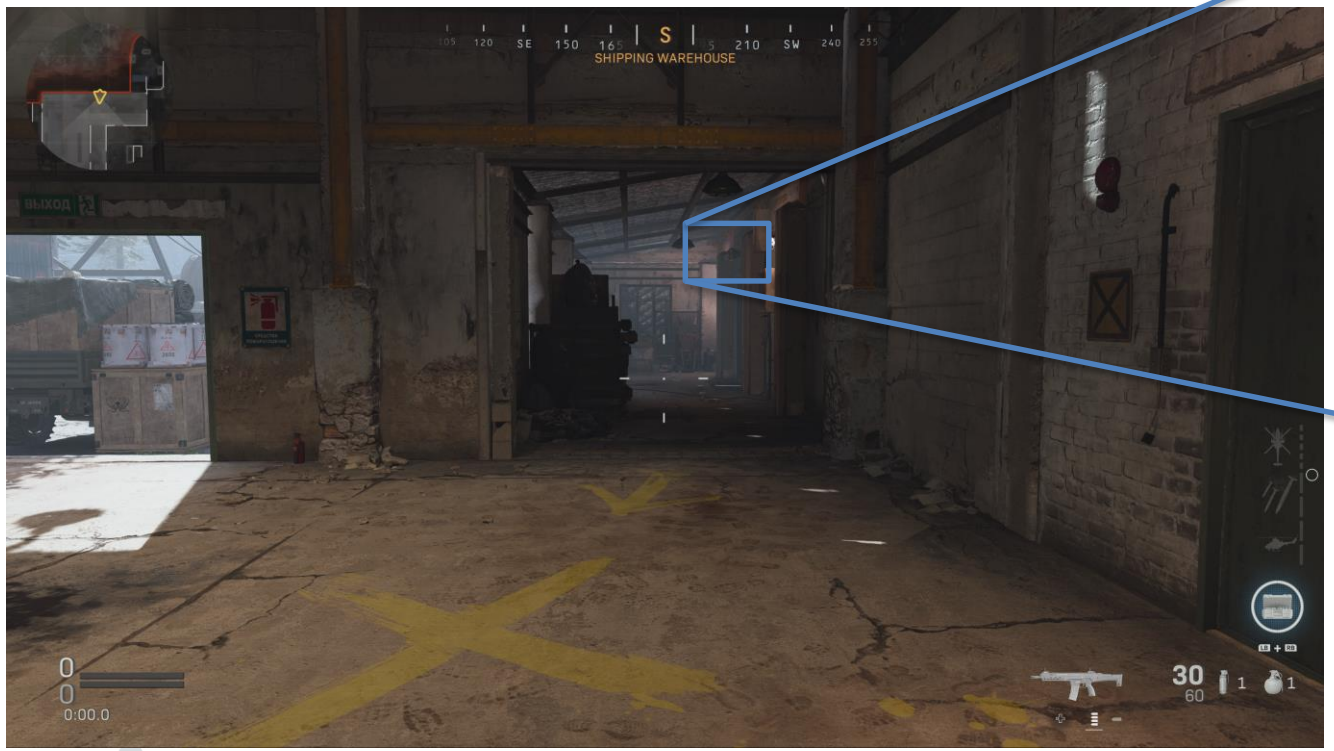
$$lodDist = \overrightarrow{\|D\|} * invFovScale * scenResScale * vertexProcessingScale * gpuPerfScale + geoBias$$

LOD: Runtime

Additional Considerations

LOD: Runtime

Reject small objects



LOD: Runtime

Reject small objects

LOD: Runtime

Procedural motion

LOD: Runtime

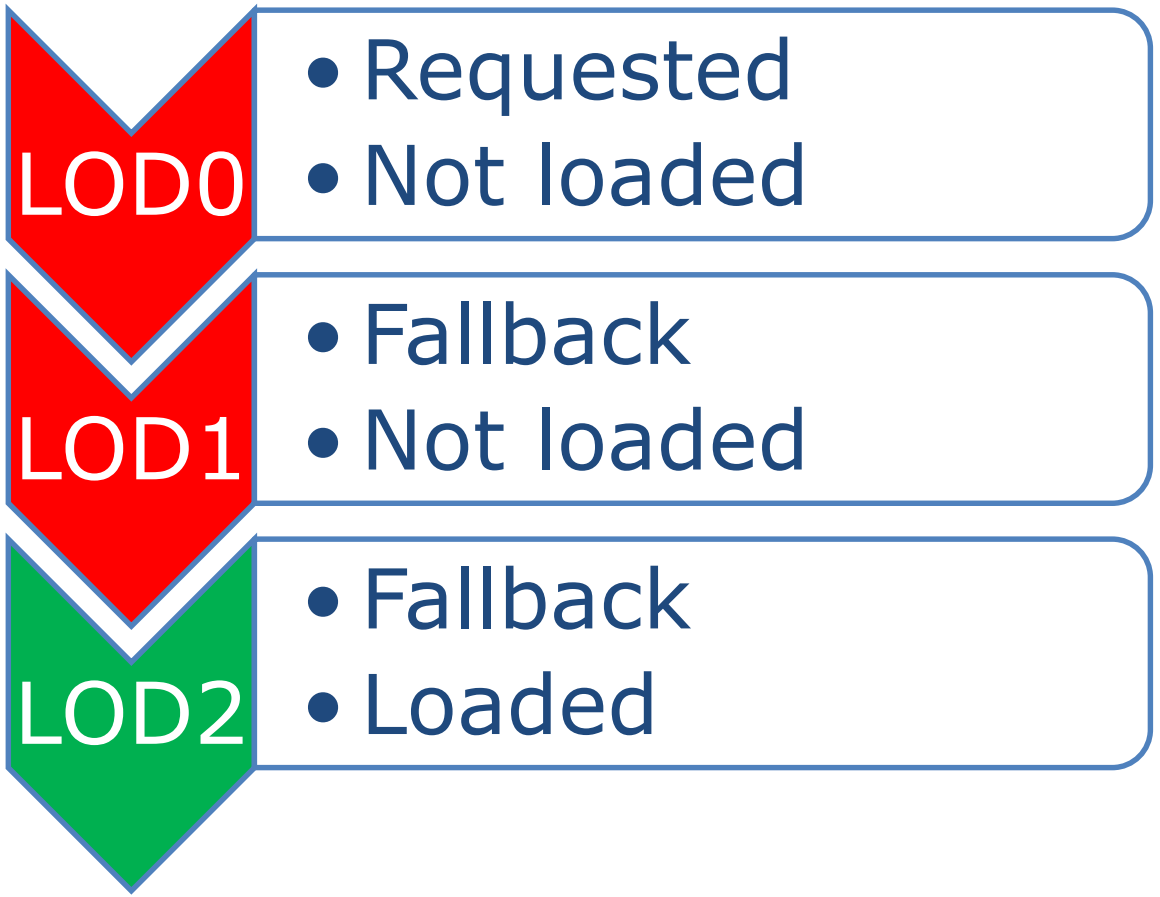
Static models – keep it fair

LOD: Runtime

Static models – keep it fair

LOD: Runtime

Streaming



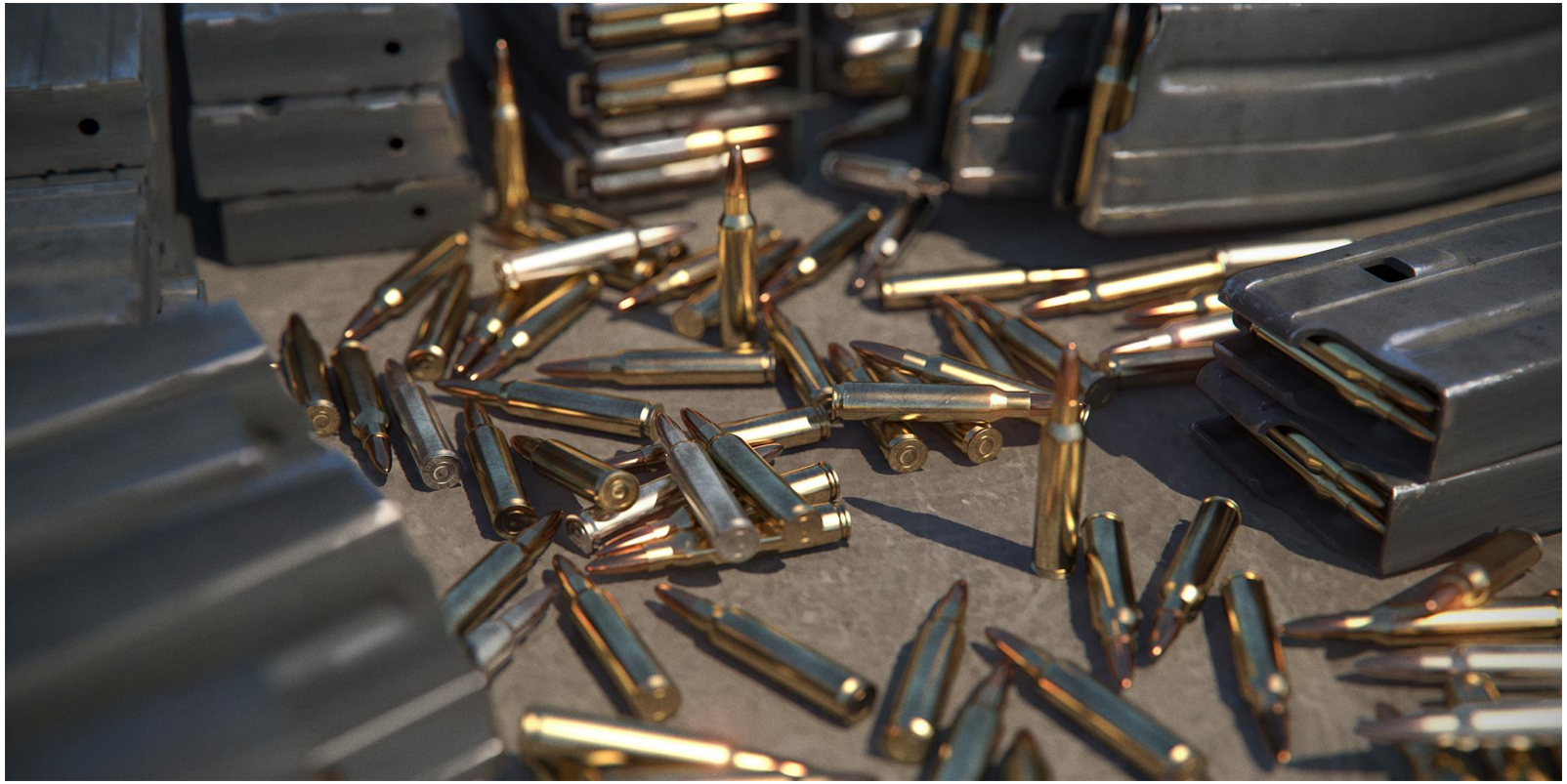
LOD: Runtime

Additional Considerations

- Reject small objects
- Procedural motion
- Static models – keep it fair
- Streaming

LOD: Runtime

Ship it?



Proxy LOD's

AKA: ALL OUT WAR!

Proxy LOD’s

“Warzone” overview

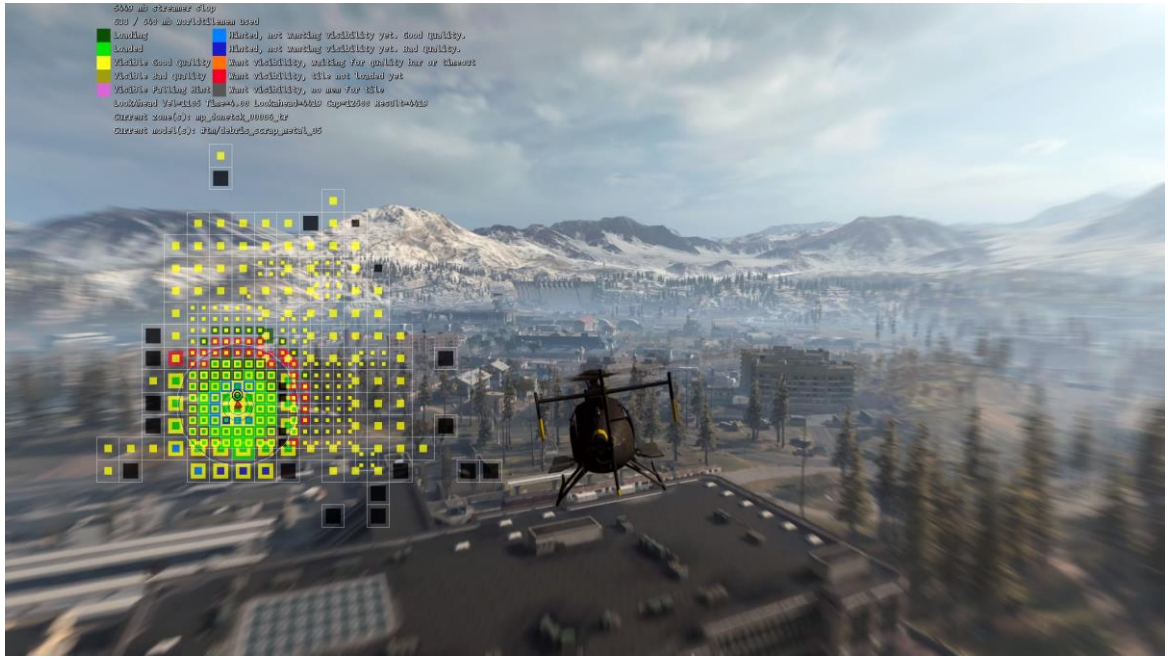
- 200-players
- Ground and air vehicles
- 120x the area of most 10v10 maps but equally as dense
- Long sightlines

Proxy LOD’s

- “Warzone” problem #578: too many small draws (GPU)
- “Warzone” problem #898: too many sub-pixel triangles (GPU)
- “Warzone” problem #976: too many visible models (CPU)

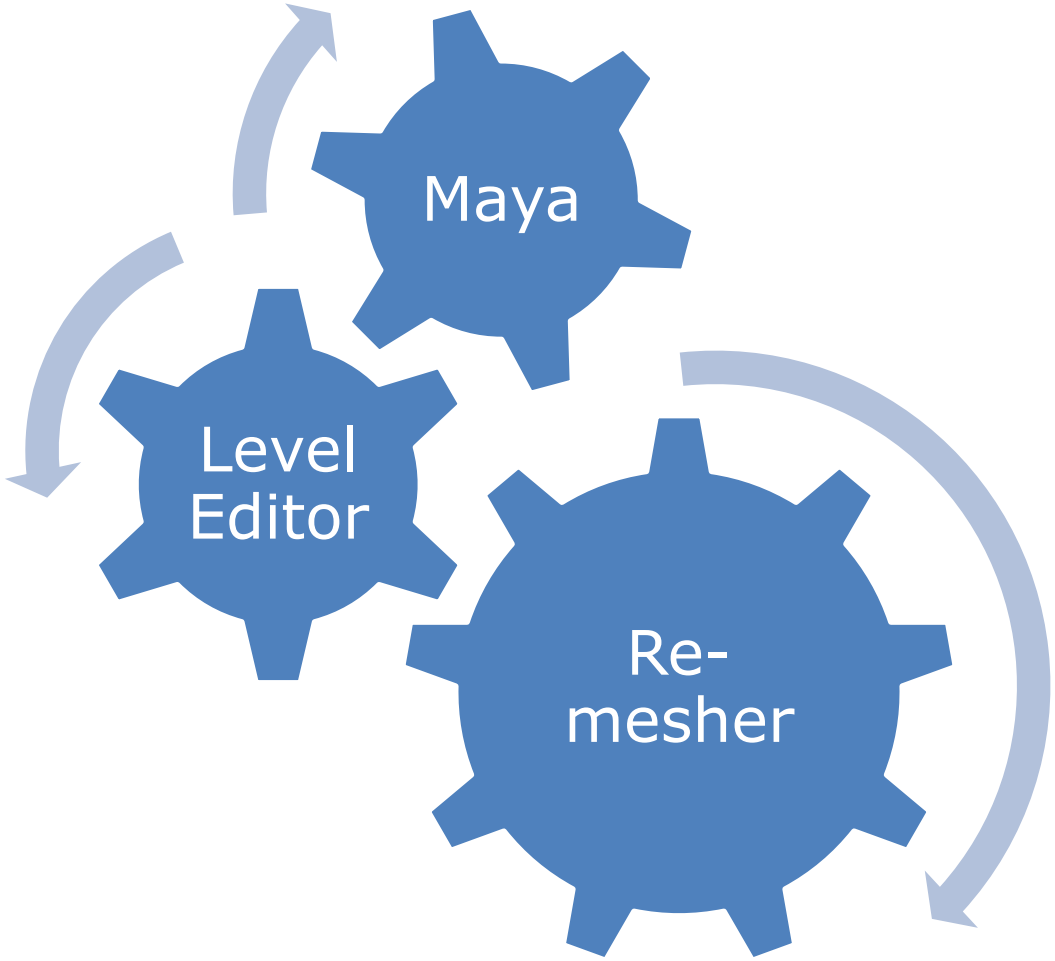
Proxy LOD’s

A single model that represents a world grid cell. It is a remeshing of all constituent bits of static geometry in that cell.



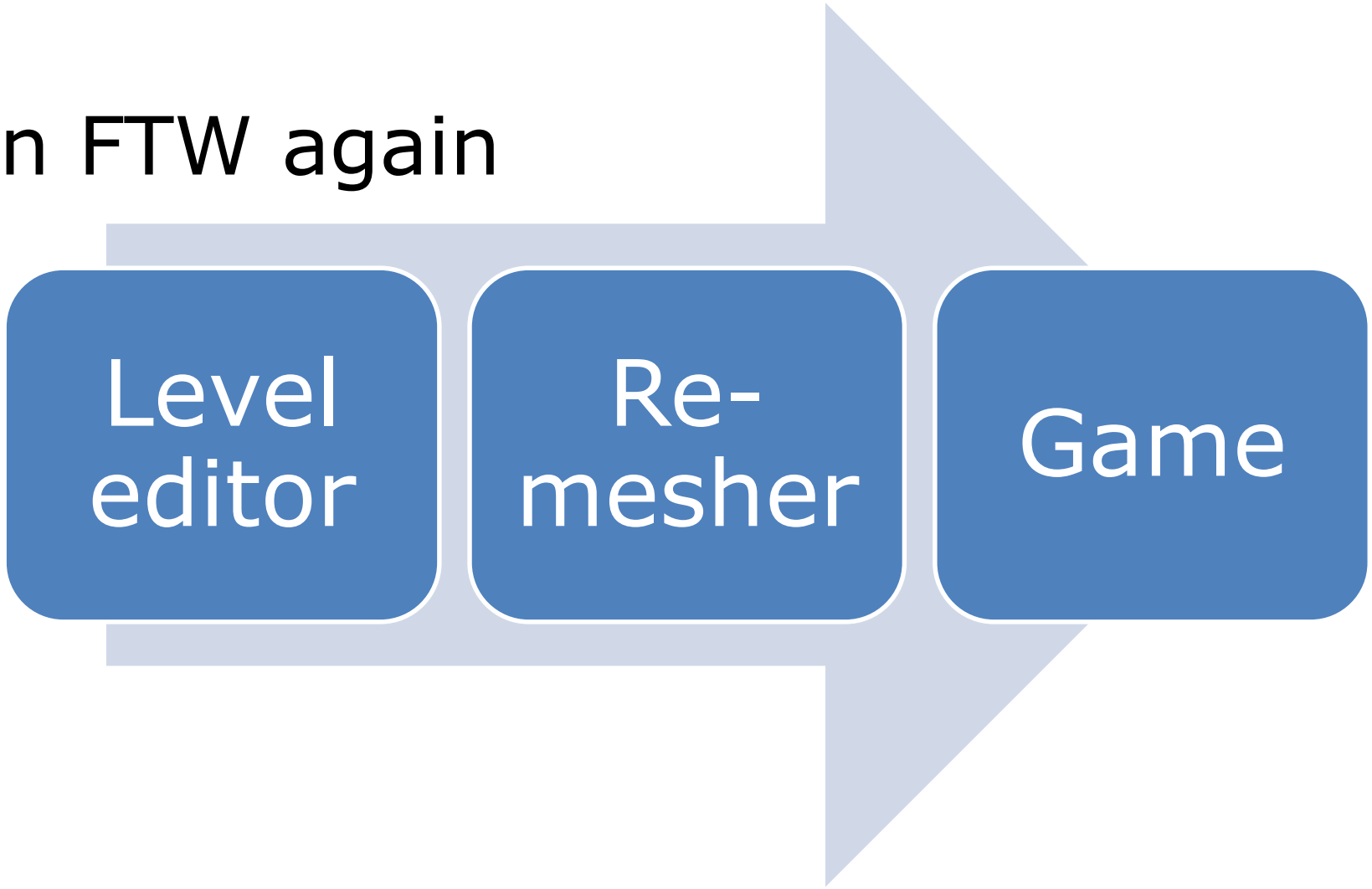
Proxy LOD’s

Manual generation?



Proxy LOD’s

Automation FTW again



Proxy LOD’s

- Input: 1000’s of unique models, other world geometry, and materials. 10’s of millions of triangles!
- Output: A pair of models, each with a single material. They represent medium and far distance versions of the world cells.


АКРОПОЛЬ
КОНСТРУКЦИИ


АКРОПОЛЬ
КОНСТРУКЦИИ

Proxy LOD’s

Hurdles

- Matching material model
- Handling building interiors
- Transparent surfaces
- Trees
- Large objects
- Small objects
- Small objects that make up large objects
- Caching
- New HW (+specs)

Conclusion

AKA: MISSION COMPLETE!

Conclusion

- A robust set of tools is crucial for creating geometric content that strikes the appropriate balance of runtime visual and performance targets.
- Core behaviors of these tools need to be ironed out early - not only for efficiency, but to instill confidence in artists that they’re not working with a moving target.
- These tools should guide artists with actionable feedback, not force their hand.
- Optimal runtime geometric LOD selection should be based off a number of dynamic criteria – not just distance from camera.
- Largescale world LOD’ing has a different set of requirements to balance against.
- Since all the inputs to world LOD’ing have been fine-tuned, it’s desirable for this process to be (almost) fully automated.

Thank you!

- JOSH CARATELLI
- ANDREW LACEY
- ANDREW SHURNEY
- GEOFF CARLTON
- MICHAEL WHITE
- JEFF DEGENHARDT
- KYLE MCKISIC
- RICCARD LINDE
- BREK BRADLEY
- JOSIAH MANSON
- MICHAL DROBOT
- KH SING
- DARREN HORROCKS
- DAN NELSON
- ANTHONY CAROTENUTO

Questions?

TWITTER: @RULONR