Photogrammetry and Star Wars Battlefront

Andrew Hamilton
Lead Environment Artist
Kenneth Brown
Technical Art Director
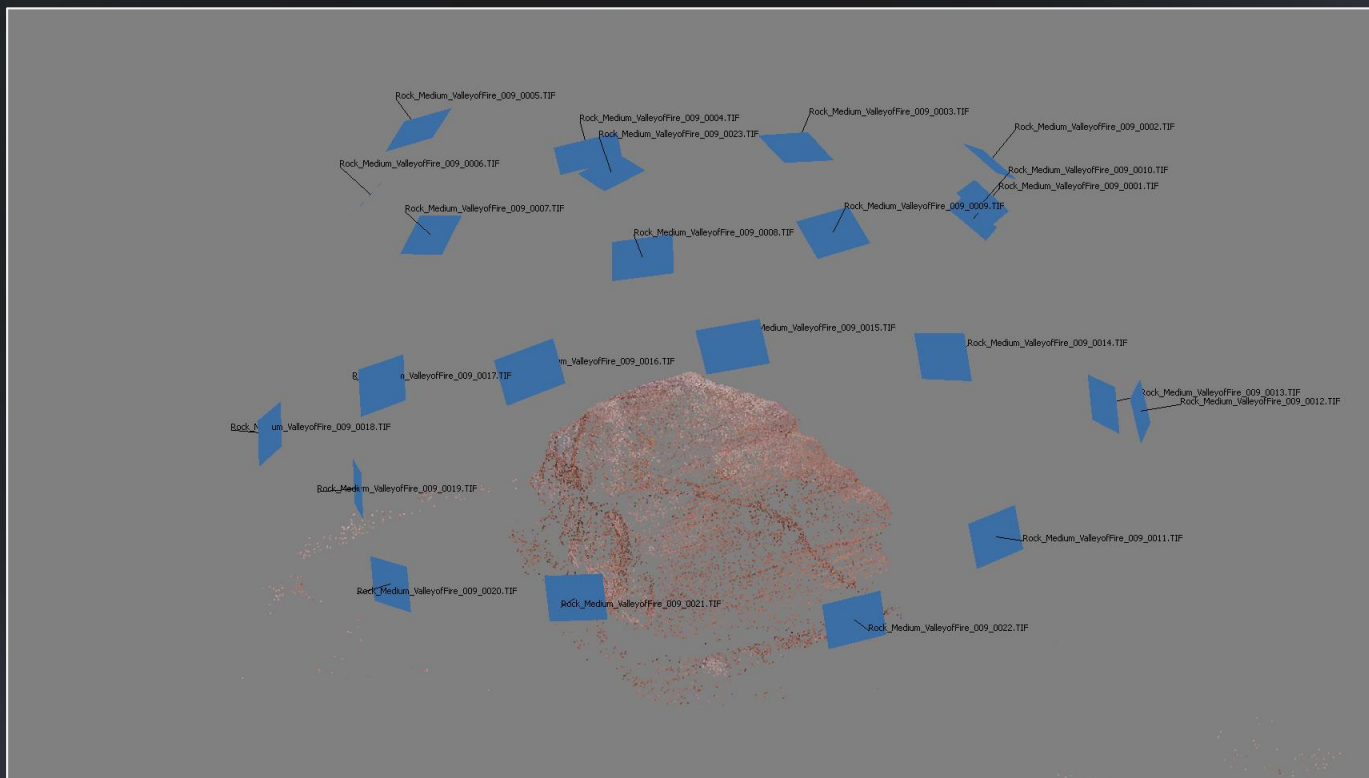**EA DICE**

Photogrammetry
Buckle Up.

Photogrammetry

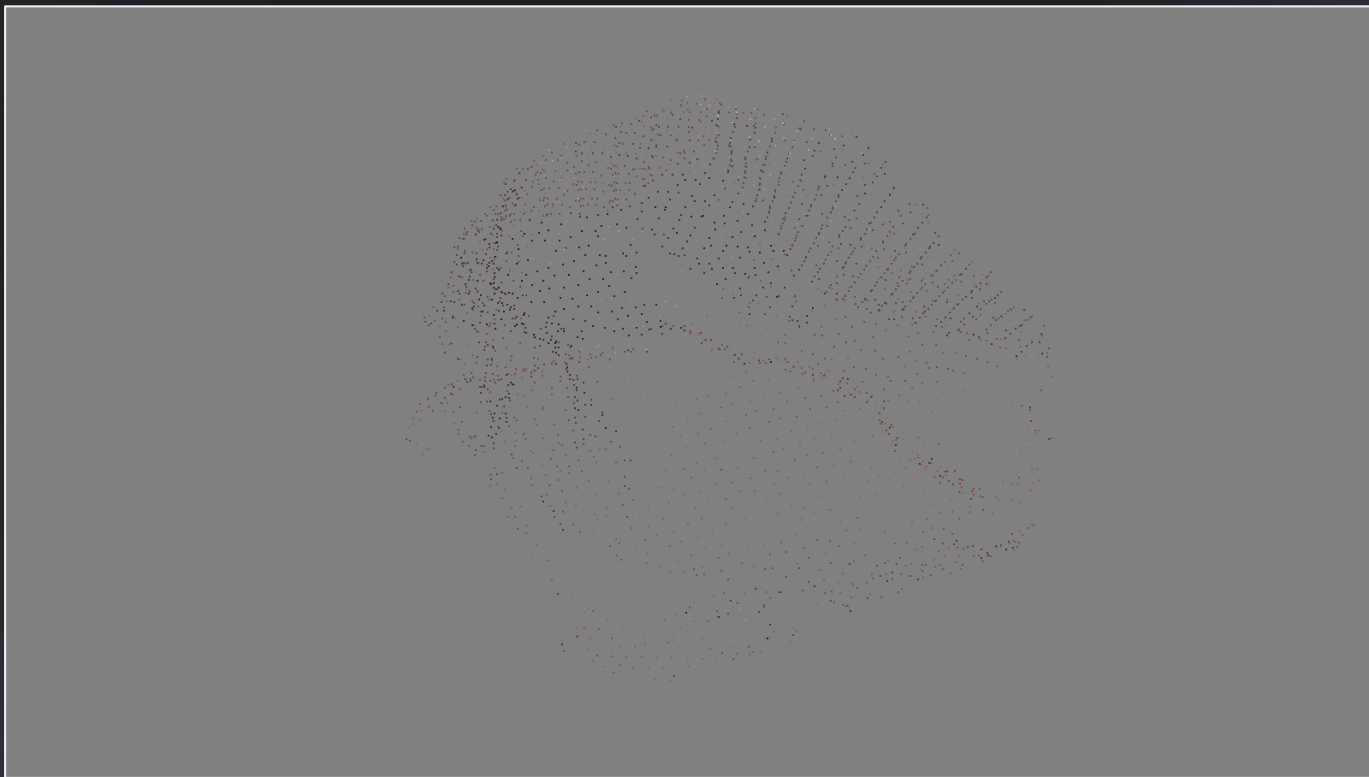Using photos for measurements and, in our case, modelling/texturing.
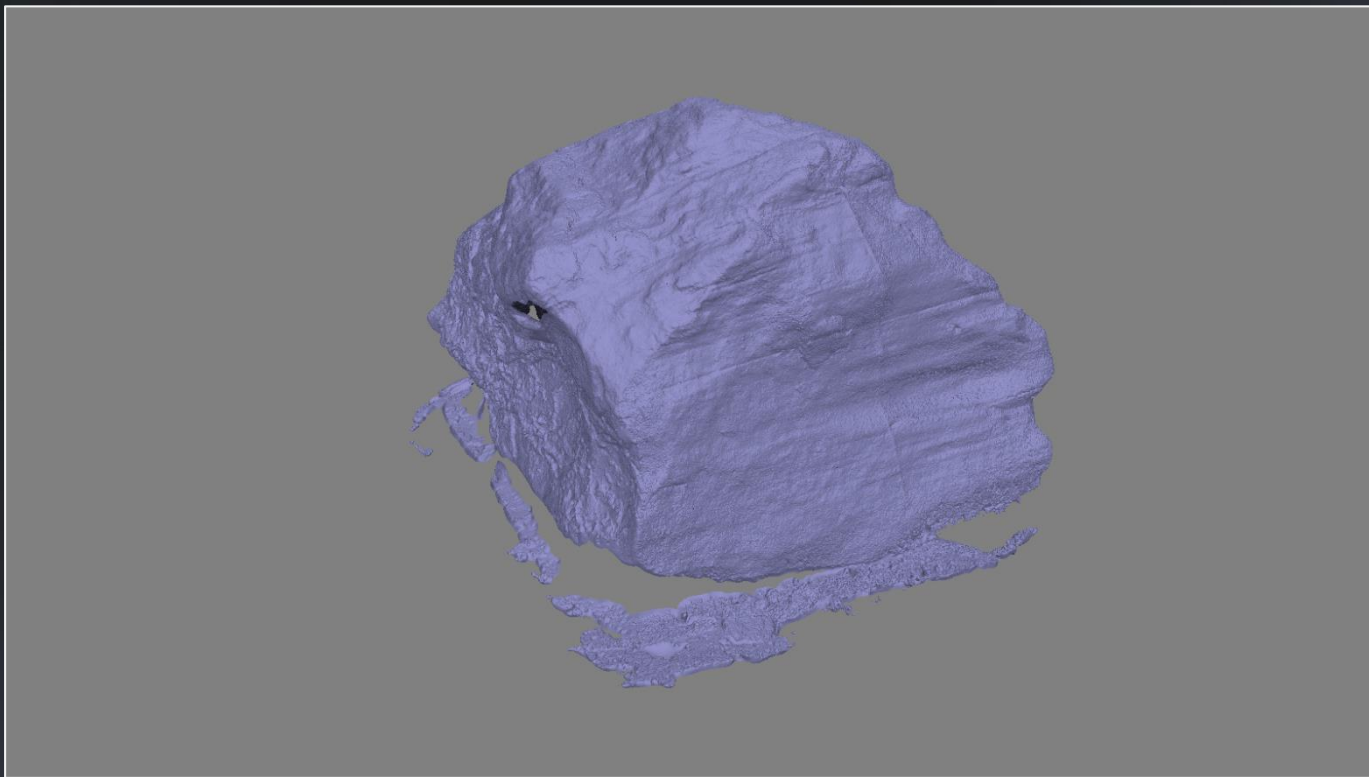
Photogrammetry
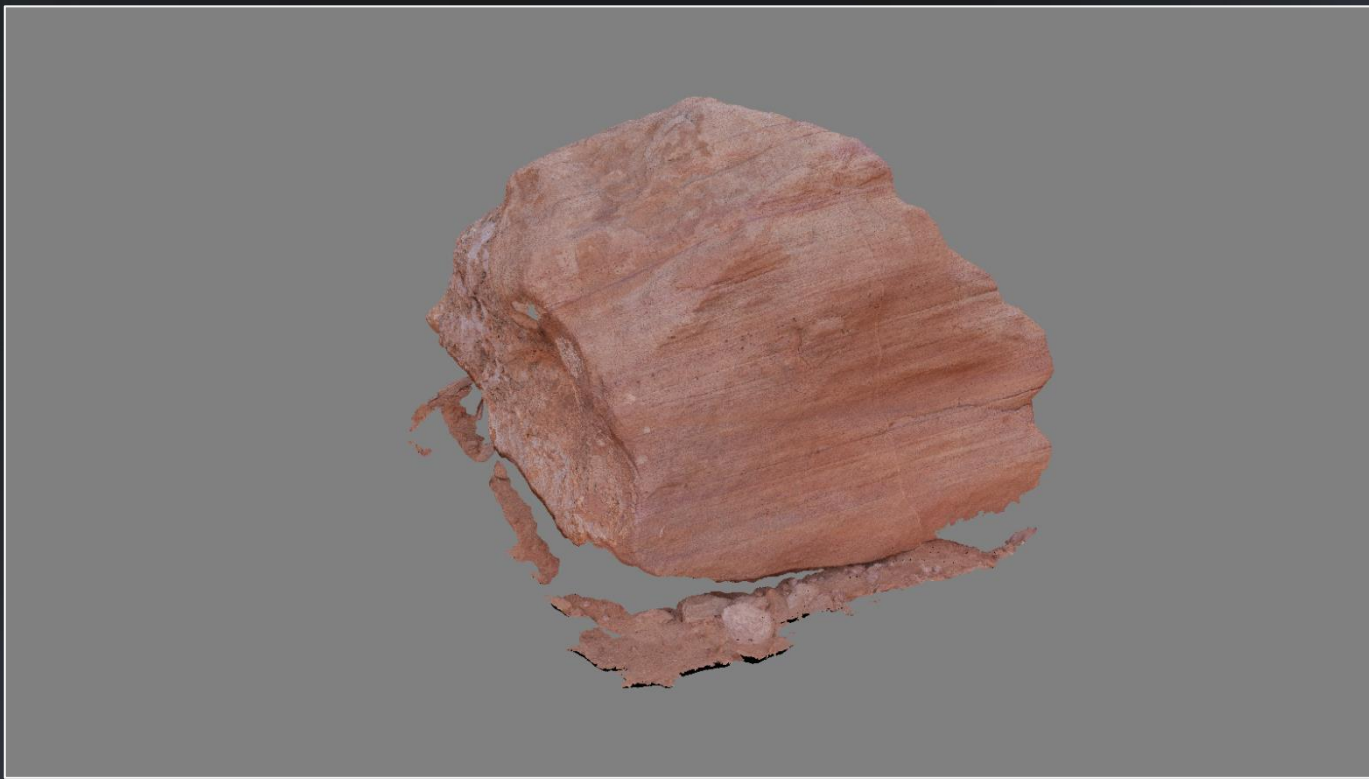
Photogrammetry

Photogrammetry

Photogrammetry

Photogrammetry

The End.

A Retrospective

- The beginning: research and development
- Scanning plans and goals
- Capturing the planets
- Mentality and workflows
- Game assets in a "photogrammetry world"
- Terrain
- Level Construction Kits
- Key learnings and takeaways
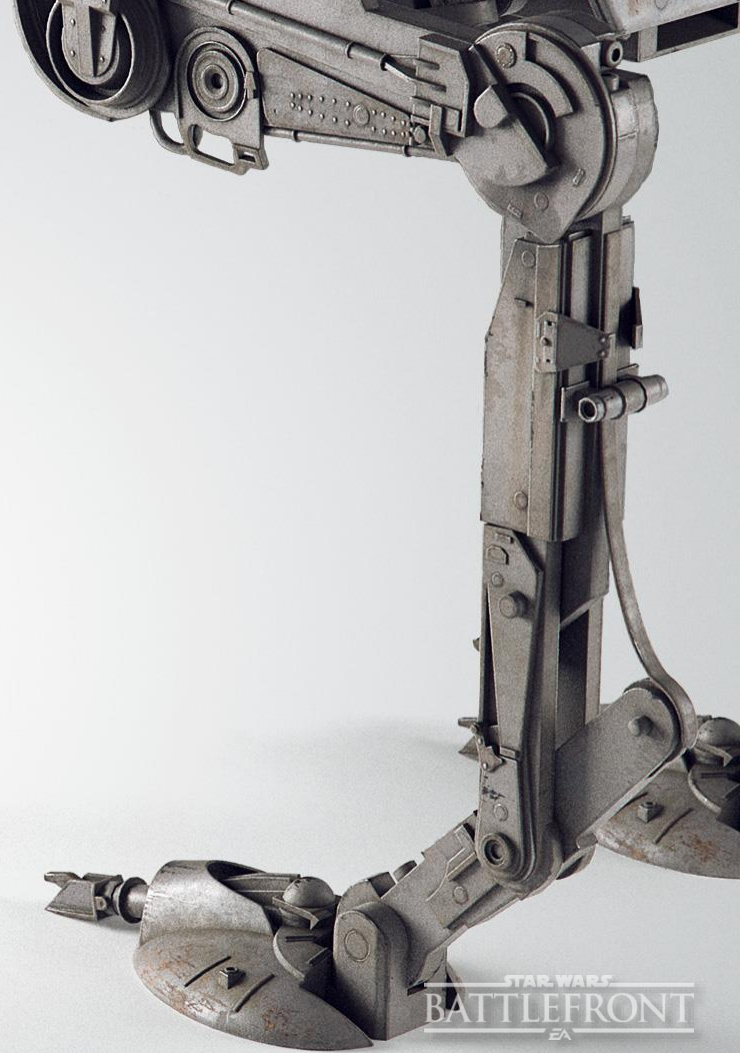
DICE

STAR WARS
BATTLEFRONT

Big Changes

- New console generation
- New franchise
- Very high expectations to deliver after BF4
- A clean slate; we want to do it right!
- PBR on the way but not ready (in the beginning)

Challenges

- Star Wars!
- Learning from the past
- Photogrammetry pushing the boundaries
- A really tight schedule
- If we're saying 60, we need to be 60
- Keeping it together
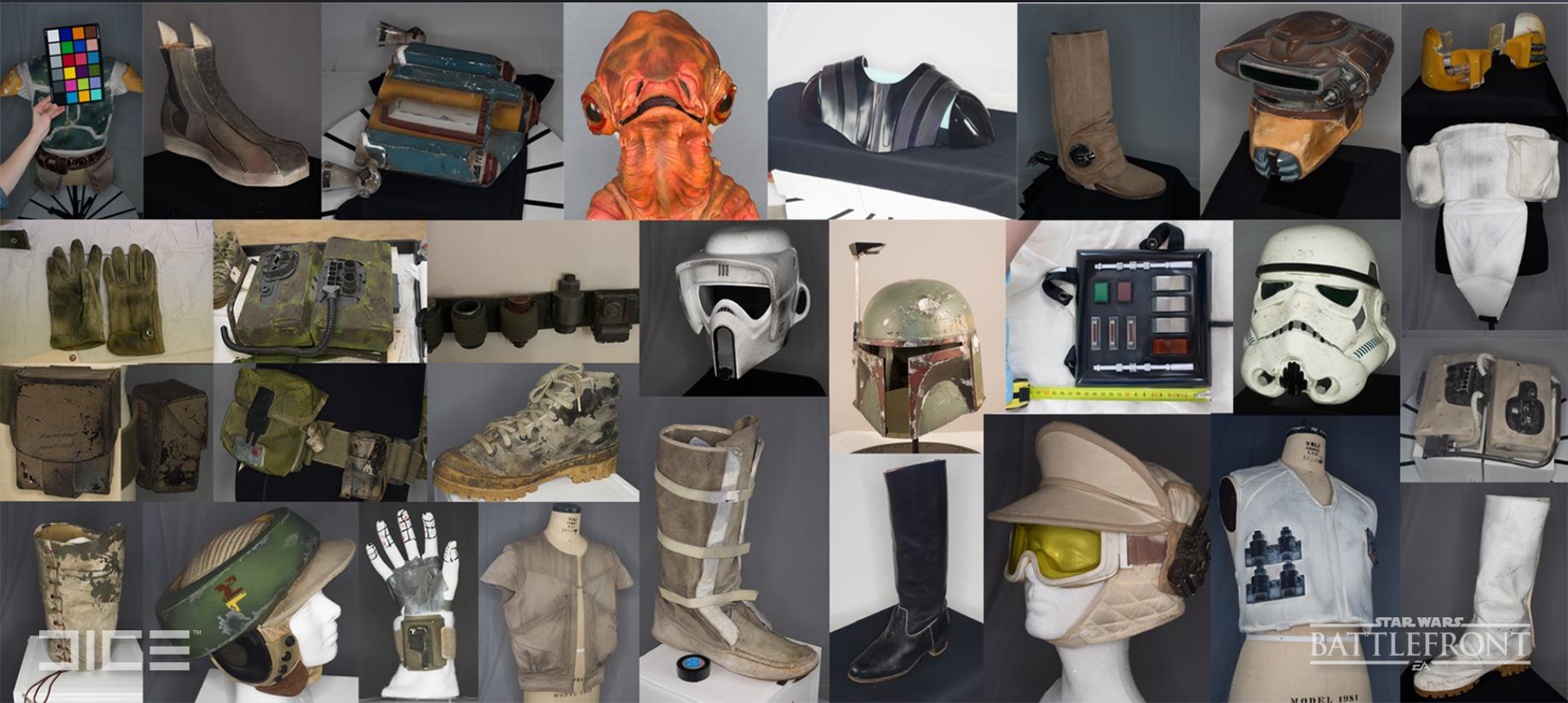
Dev Structure

- 3 month development cycles

- Content development until the end

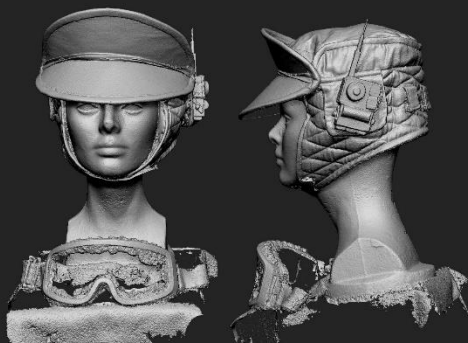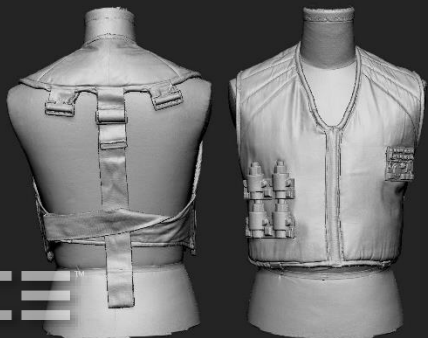- Need for 2 week buffer discovered

- Approval lock

- Stabilization

The beginning
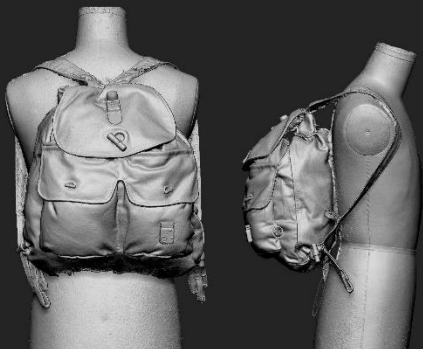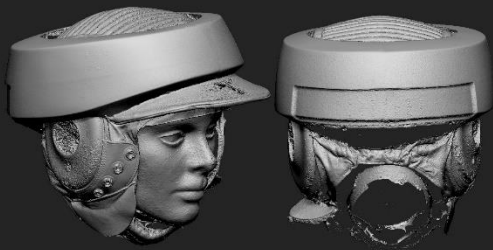
Into Star Wars: Battlefront
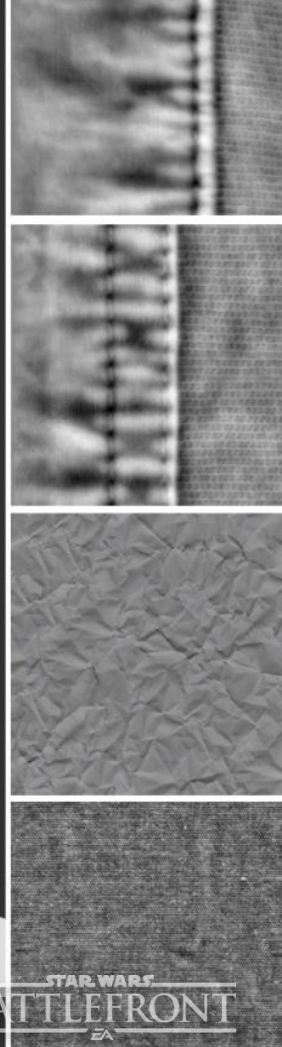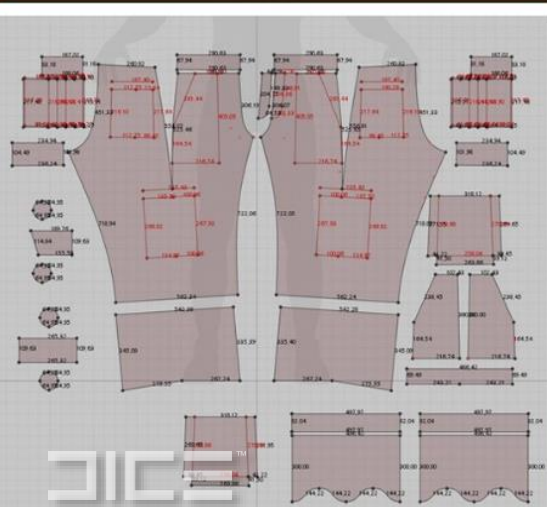
Scan everything!..?

- Bridge the gap between scanned characters and their surroundings.
- Make the best looking Star Wars game ever! ..at 60fps 😐
- We have so little time…

From here...

Battlefield 3: Armored Kill

..to here!

Location capture goals

- Capture the complete asset libraries of all planets.
- Focus and capture only what is essential, not everything we see.
- Capture meaningful variation and features.
- Visit the real film locations for true inspiration and understanding.

Capturing on location

- Simple kit (24mm lens, Canon 6D, tripod, color chart, etc)
- Ideal weather conditions
- On-location breakdown
- Thinking in terms of usable pieces
- Aware of Frostbite strengths/weaknesses
- Focus on quality over quantity of captures

A. Dirt ground
B. Leaves and twigs
C. Clover ground
D. Clover leaves
E. Fern large
F. Fern medium
G. Fern small
H. Log large
I. Log medium
J. Log small
K. Tree large
L. Tree medium
M. Tree small
N. Tree stump
O. Vegetation
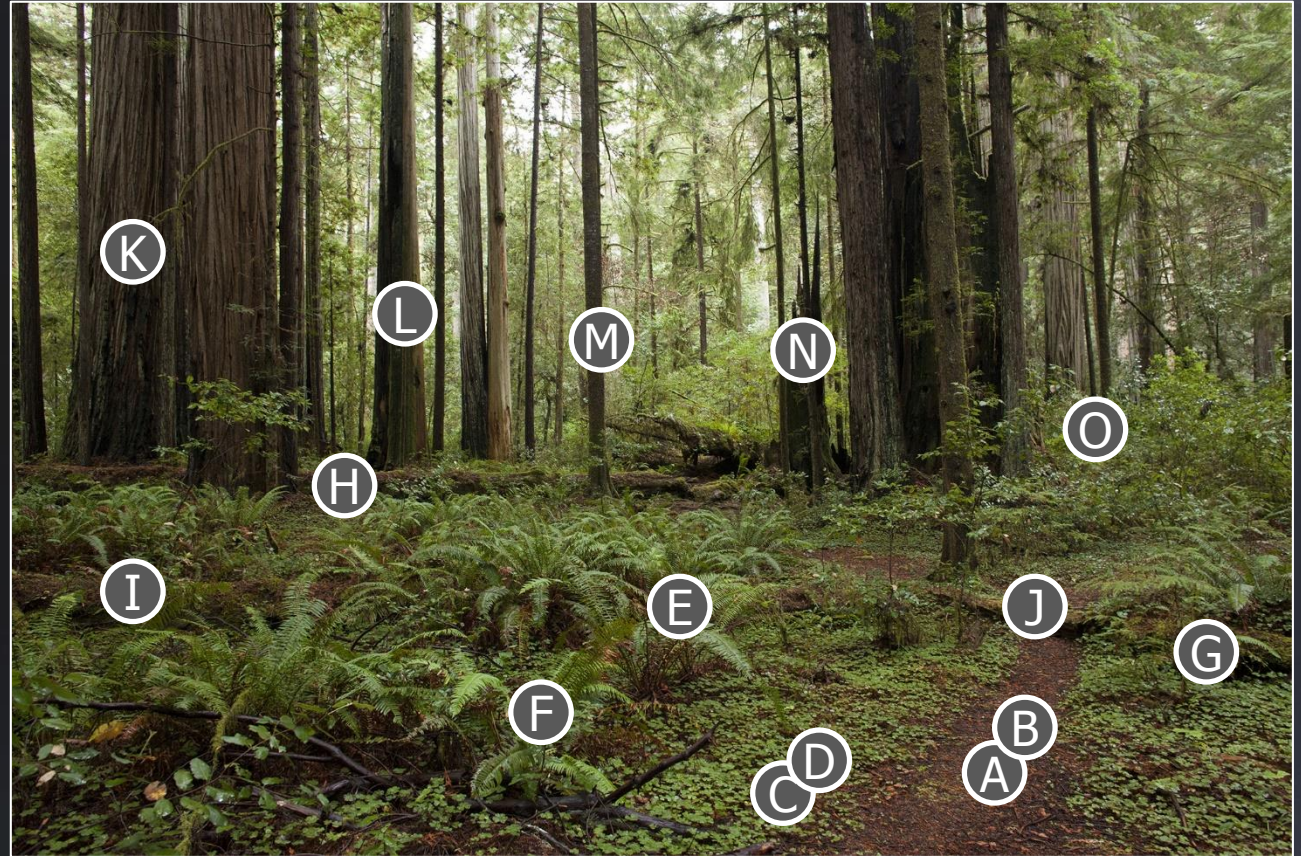
Best practices

- Overlap!!

- Covering *all* angles to reduce cleanup.

- Avoid cast shadow - ambient light is easily removed later in the process.

- Most reliable results occurred when full asset was visible in most photos.

- Avoid objects with too intricate details.

- Be sure that all capture methods are consistent and clean.

- Prepare yourself and equipment for various conditions – weather can make the process sloppy and destroy consistent results.

Capturing the planets

Successful location scanning

- A small and manageable group with light equipment is ideal.
- Structured and focused on location.
- Absolutely key to connect and understand how to recreate real locations.
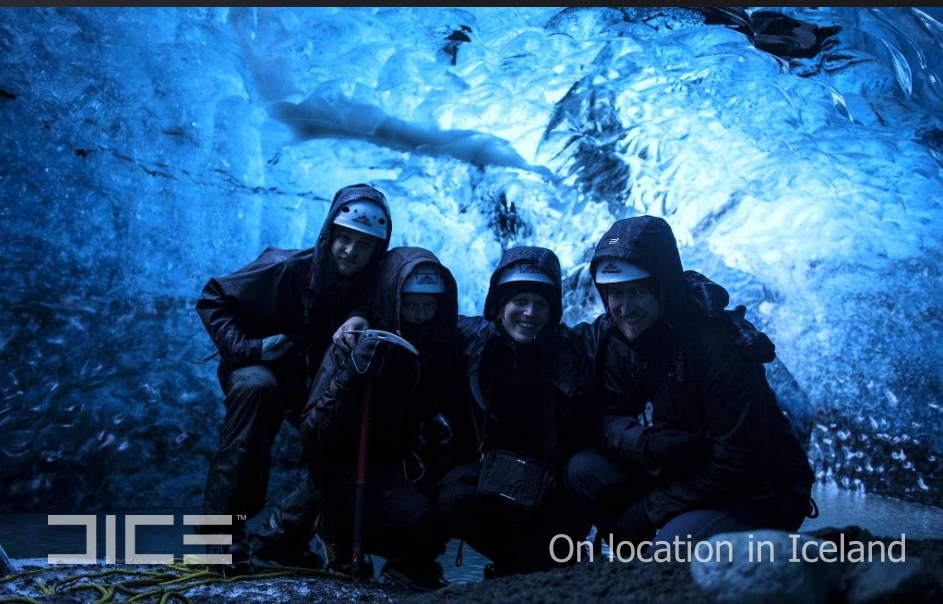
On location in Iceland

Managing the data

- Over 100,000 photos to manually sort
- Over 2000 assets to manually review
- Over 14 hours of video to categorize
- Less than ideal storage solution
- *A lot* of room for improvement here!

Workflow

Workflow

Now we have the data, so let's go.

How long does an asset take?
With Automation

Days Per Asset
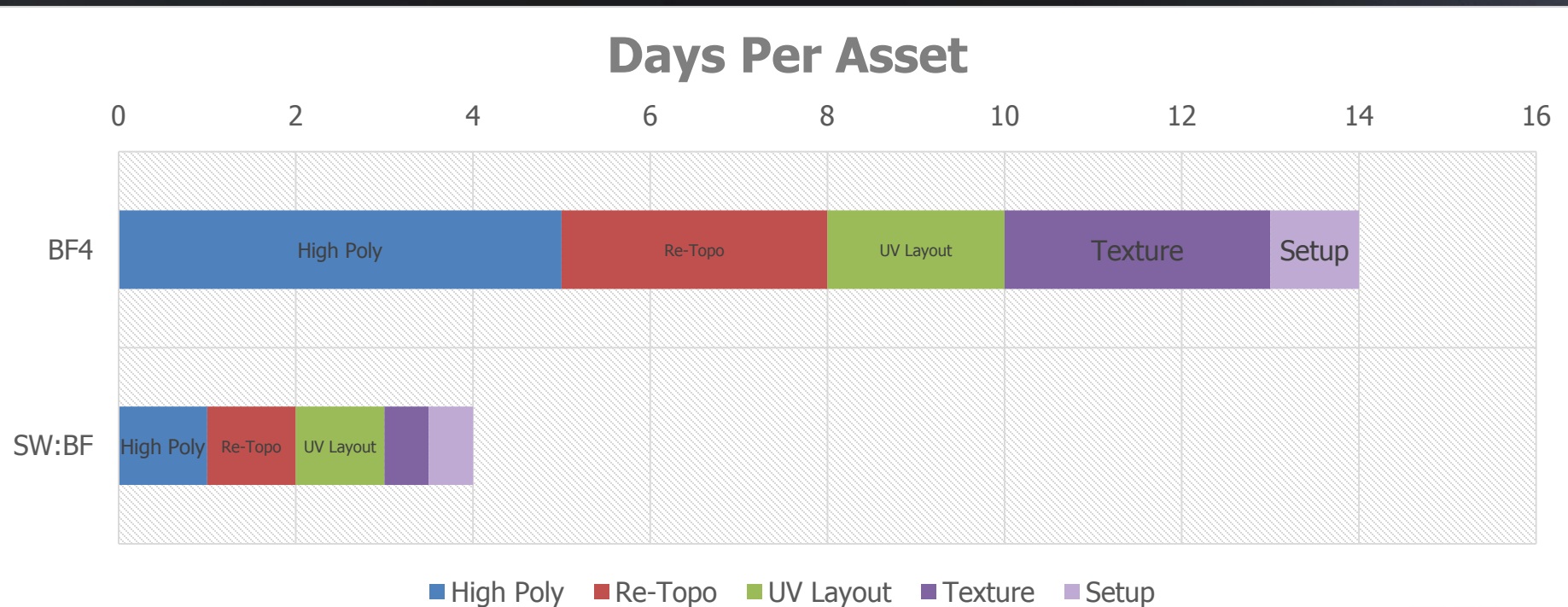
Photogrammetry workflow

- Remove vignette, keep lens distortion.
- Batch calibration to color chart.
- This entire step should be automated!
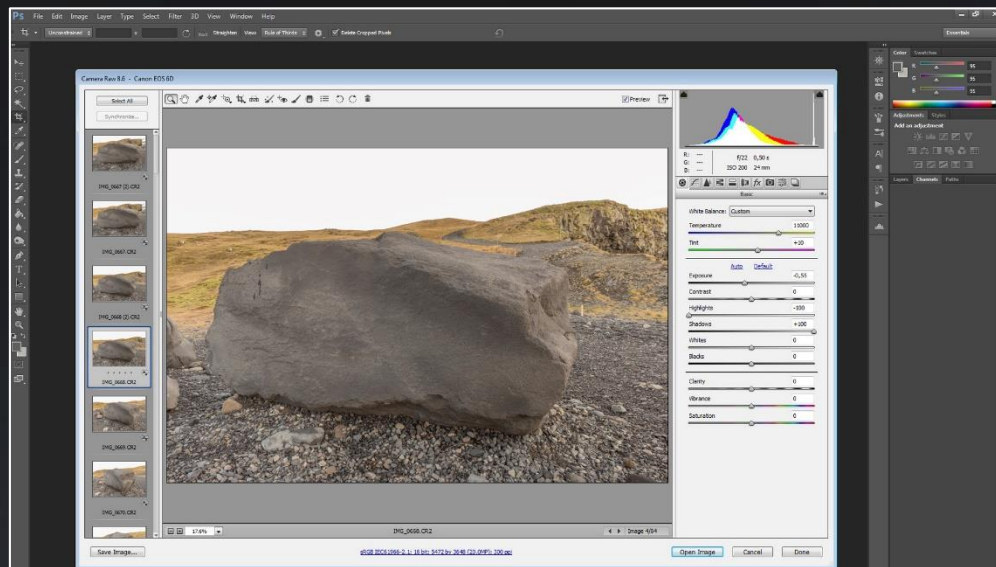
PS Camera RAW

Photogrammetry workflow

- Most assets need to be masked – very time consuming if not automatic.
- Baking at High gave most useable and manageable results for Objects.
- Baking at Ultra gave best results for Terrain – files are huge though.
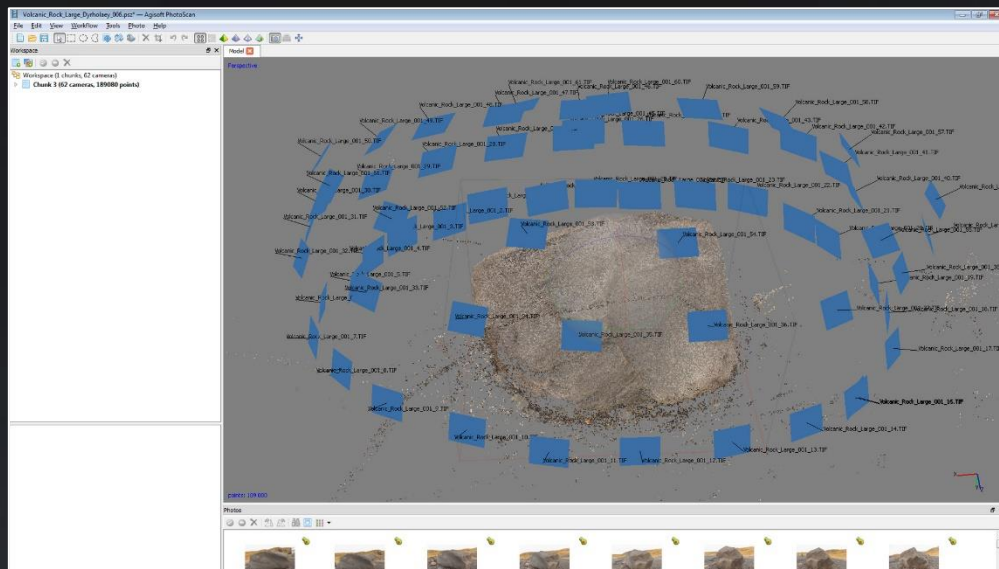
Agisoft PhotoScan

Photogrammetry workflow

- Mosaic rather than Average blend gives much sharper results.
- Shredded UVs – take to Zbrush and reproject, or bake high res.
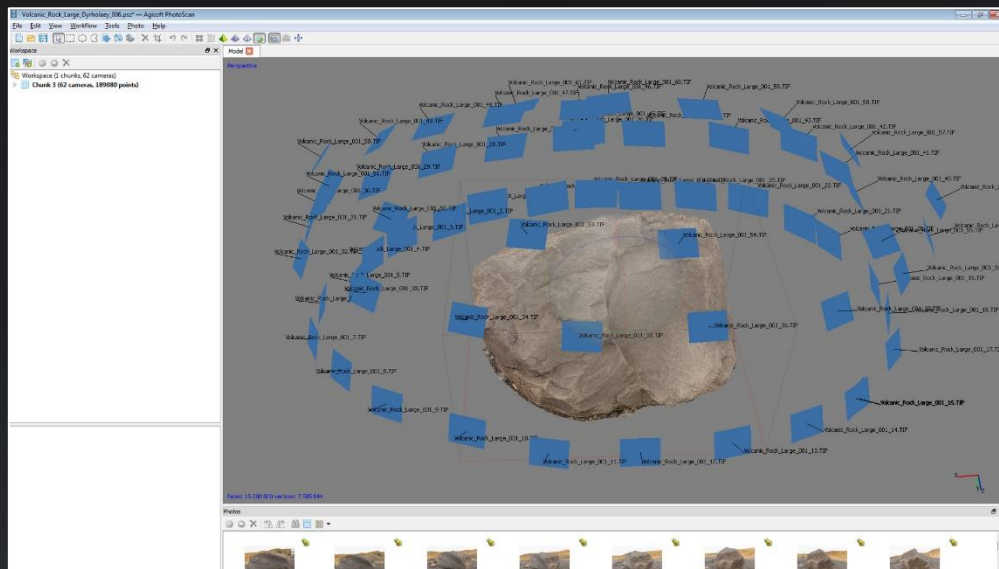
Agisoft PhotoScan

Photogrammetry workflow

- Some assets completely automated in MeshLab.
- Some needed fine tuning in Topogun – ok if very limited asset library.
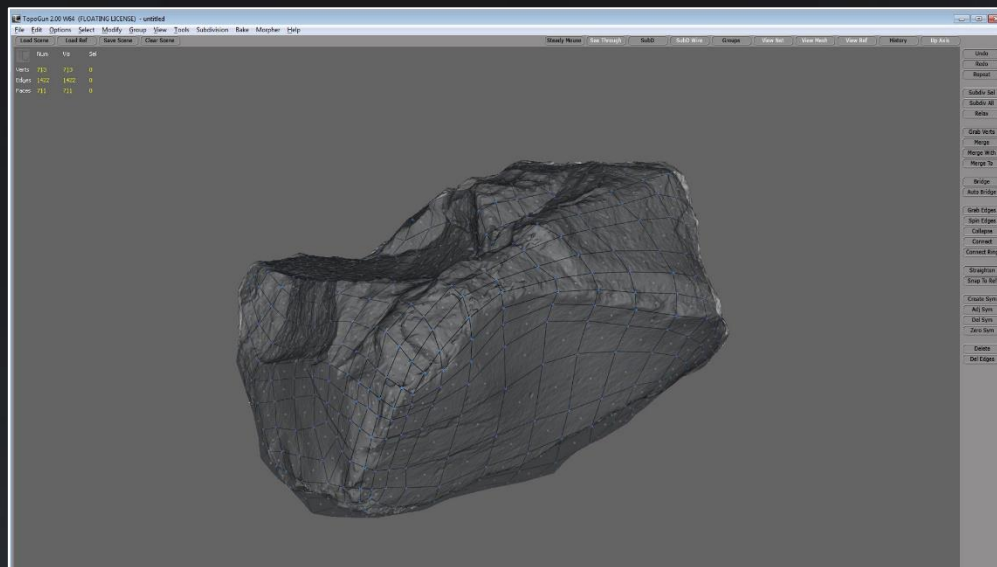
Topogun / MeshLab

Photogrammetry workflow

- Detail textures applied later means UVs demand very little stretching.
- UV direction important for consistent detail texture alignment, sometimes may need a second UV just for detail tiling.
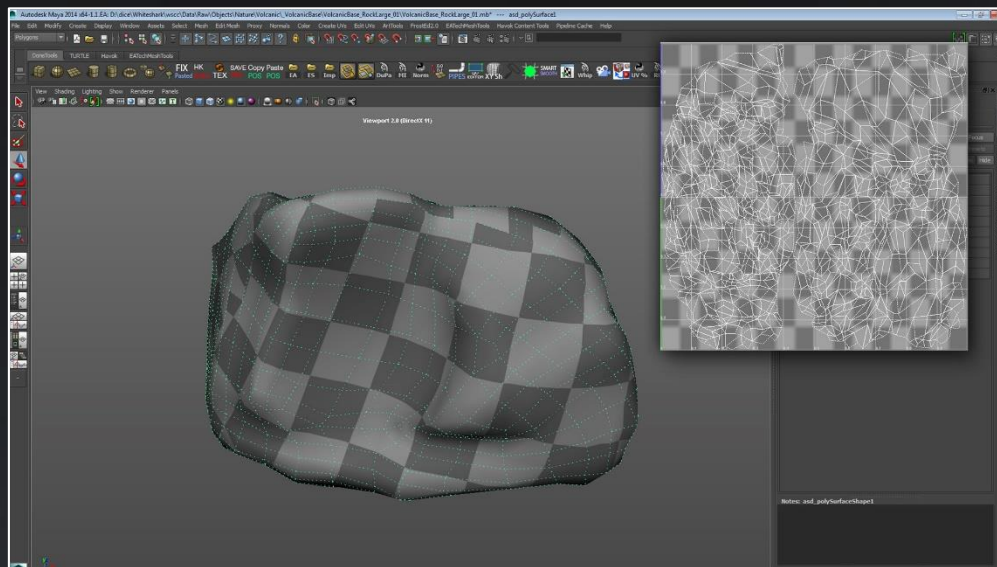
Maya / UVLayout

Photogrammetry workflow

- Color, Tangent Normals, Object Normals, Heightmap, AO, Cavity.
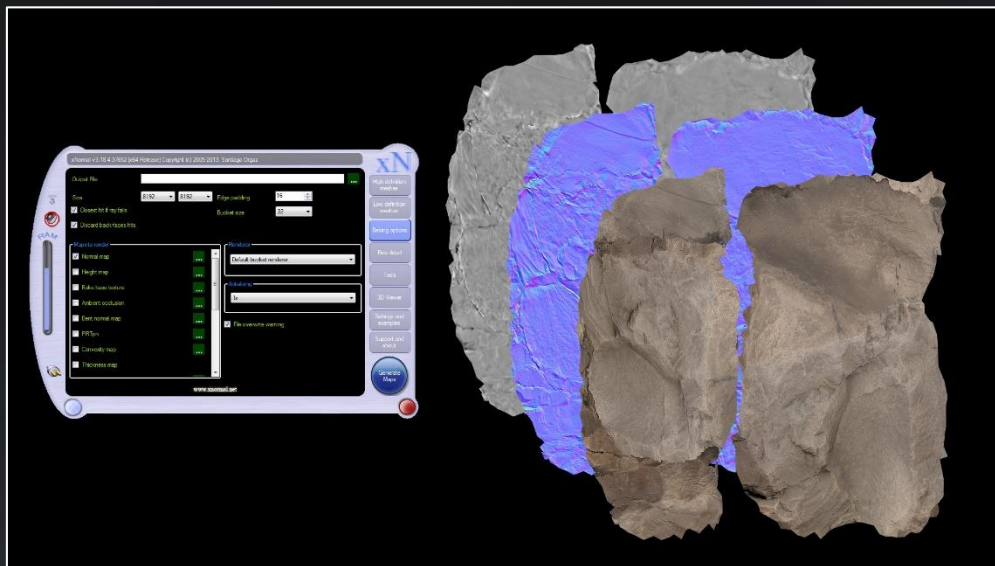
XNormal

Photogrammetry workflow

- Object Space Normals used as a mask to remove ambient lighting.
- AO or cavity map for fine tune lighting removal.
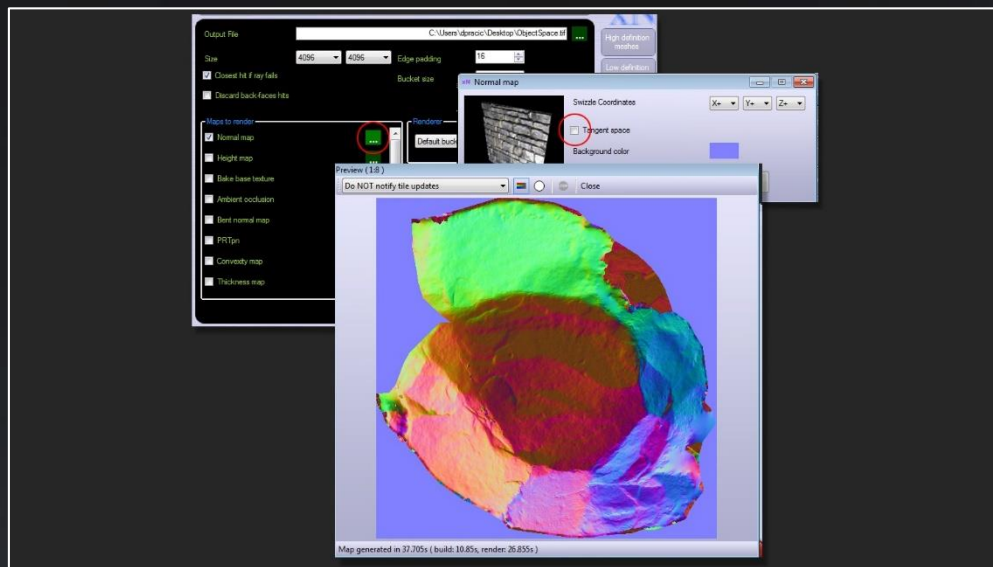
XNormal

Photogrammetry workflow
Light Removal

- 16 bit depth is your friend
- Remove as much as possible from camera raw
- Once removed, move on to ambient
- Object space normal maps ftw

Light Removal
Object Normal

- In this example, shadows are not so distinct
- Generate an object normal map
- This gives distinct colors for the different object normals
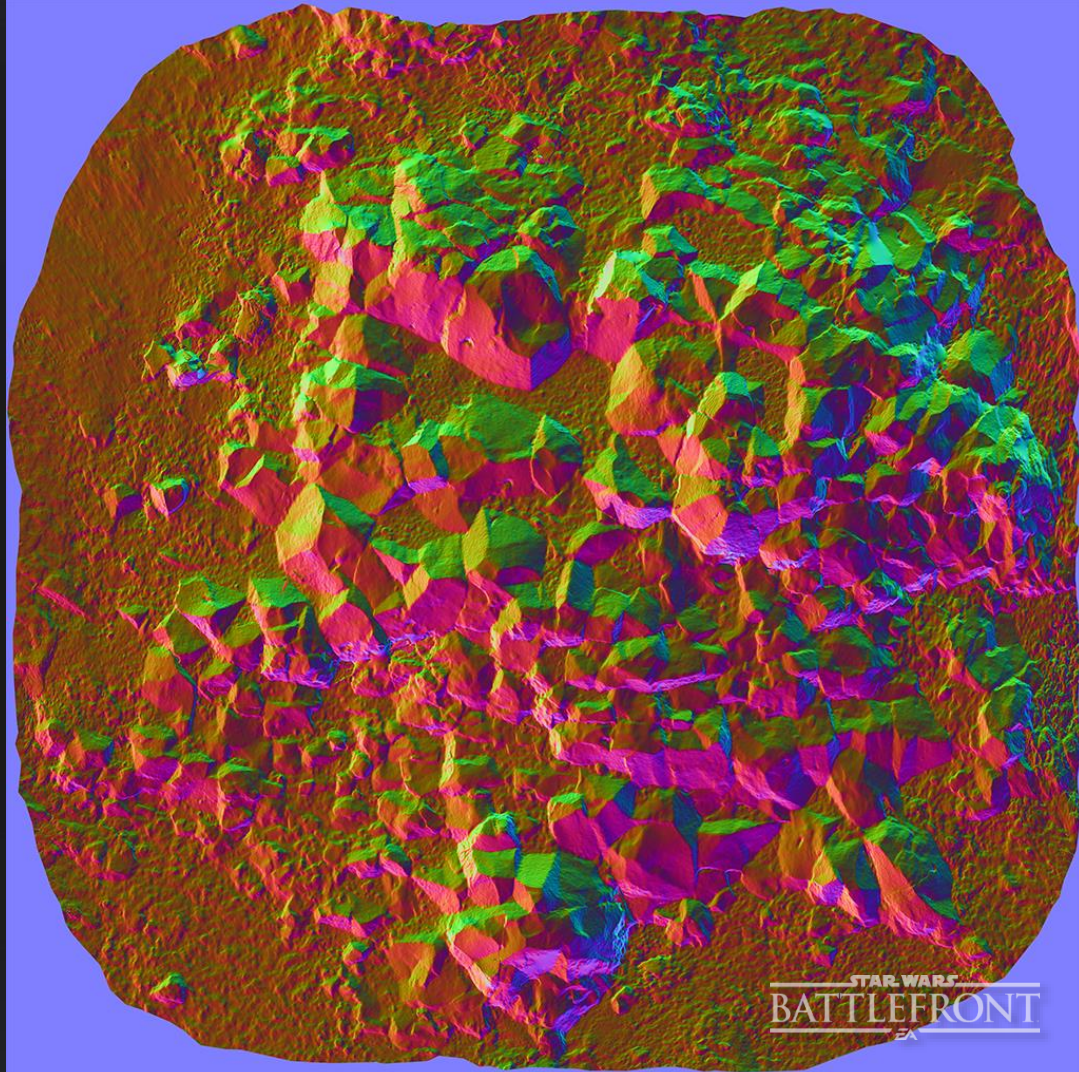
Light Removal
Object Normal

- Using Black & White mask in Photoshop you can pull out different "directions"
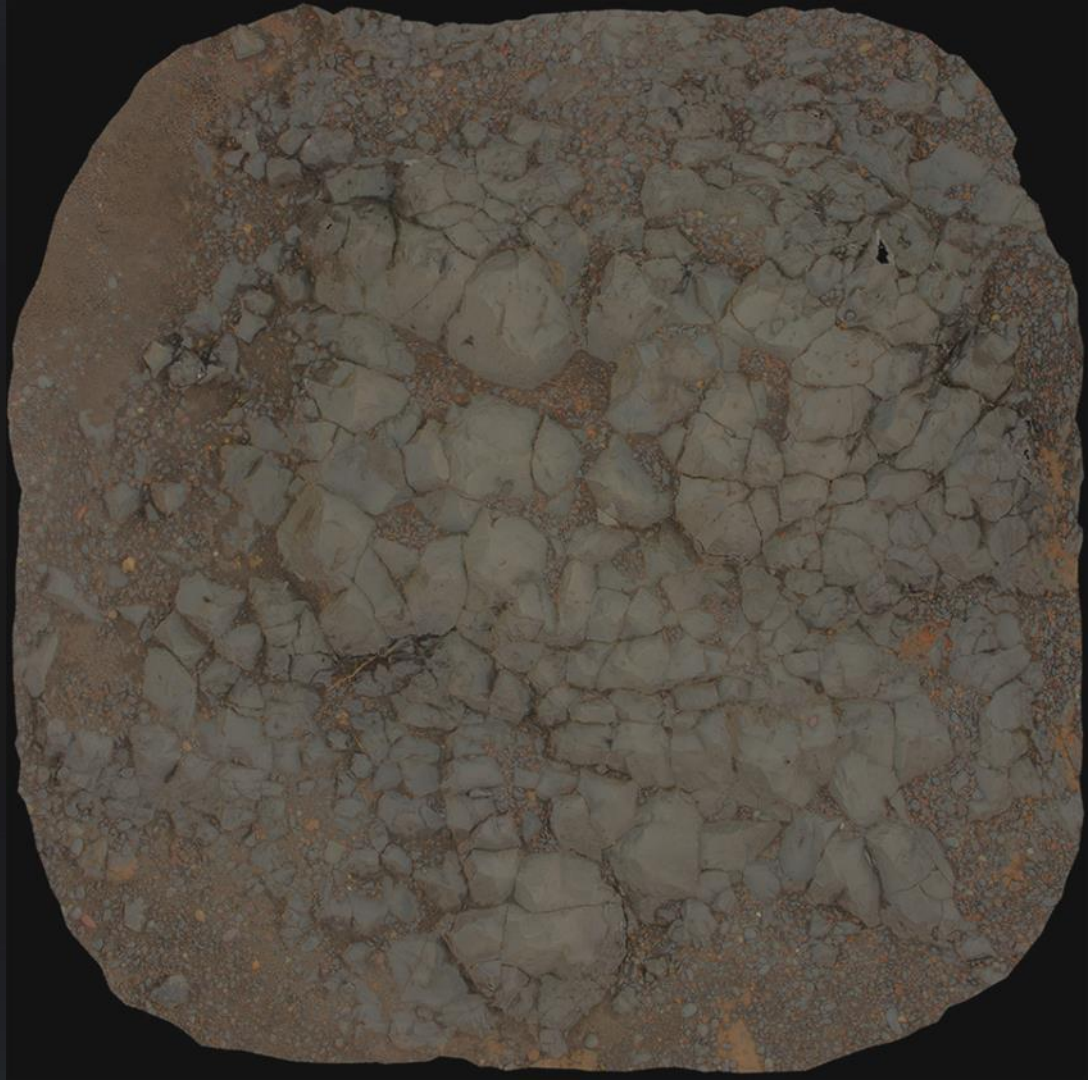
Workflow

Now we have the base "source" data...

Texel Density

- Graphical representation of a pixel

- An initial pitfall for us

- The following screenshots are 1080p

- Illustrates texture pixel (texel) to screen pixel ratio

~10 texture pixels per screen pixel
(excluding mips)

[GPU Events:]   200 5.0ms
Level   0.50 / 6.0 ms
Lighting and VE 0.00 / 3.0 ms
Shadows   0.00 / 3.8 ms
FX GPU   0.00 / 3.0 ms
PostProcess   0.10 / 5.9 ms
Total   4.27 /13.0 ms

512

512

512

512

512

512

ArcticBase_Rock_Medium_02
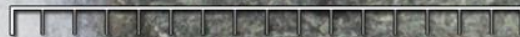
DICE

STAR WARS
BATTLEFRONT
EA

Diffuse

For Comparison: Two 4k Textures. 55mb

For Comparison: Two 1k Textures. 3.7mb

Texel Density
Takeaways

- You'd be surprised what you can get away with

- Detail maps are our friends

- Large textures have a very big streaming overhead

- We have an 800mb texture pool on Gen4

- One 8k texture is almost 90mb

Texture Arrays

- The photoscanned asset as a base is generally quite low res.
- All environmental assets use a consistent preset shader.
- Planet-specific Texture Array instance shaders add the detail on top.

Object Preset Shader
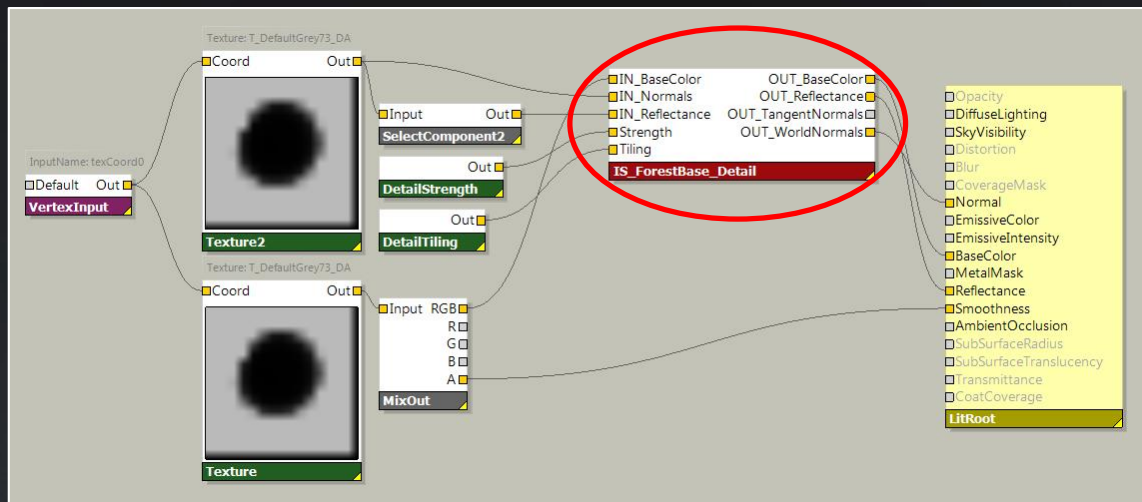
Texture Arrays

- This texture array instance shader is set up for normal, reflectance, and color.

Texture Array Instance Shader
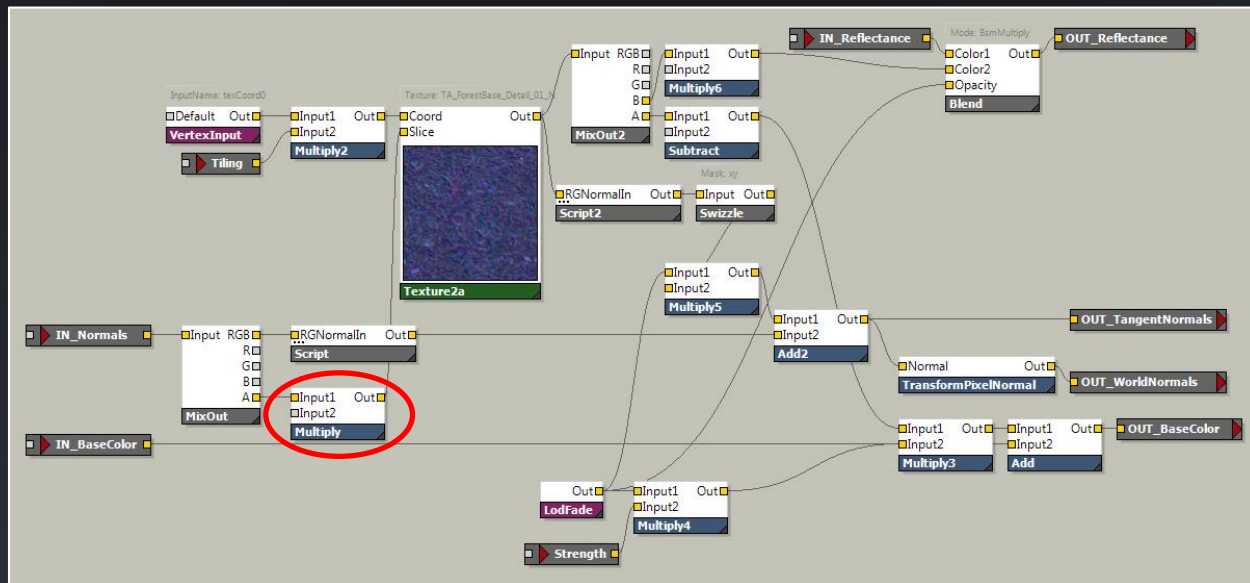
Texture Arrays

- Brightness is painted in 5 even steps (as there are 5 detail textures).
- The order of these layers must correspond to the order in the Array.

Tree Normals (Texture Array mask in Alpha)

Brightness %	Detail
0	Leaves
25	Bark
50	Burnt Bark
75	Rock
100	Moss
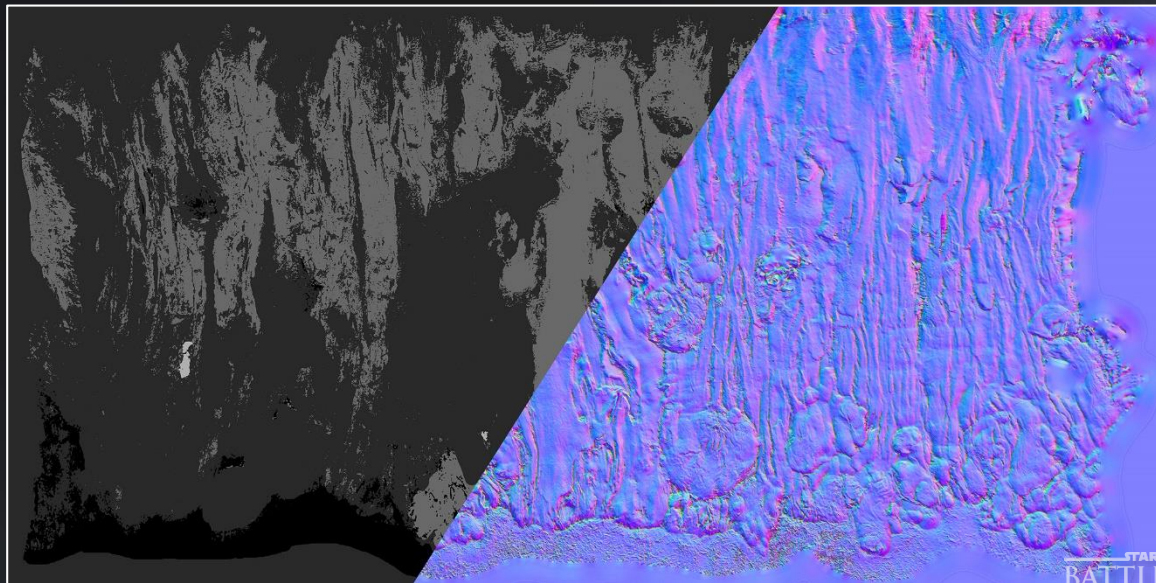
Texture Arrays

- Detail textures are fading away towards the end of LOD0 with 'LodFade'.
- LOD1 has a new simplified shader without detail textures.

Texture Array Instance Shader
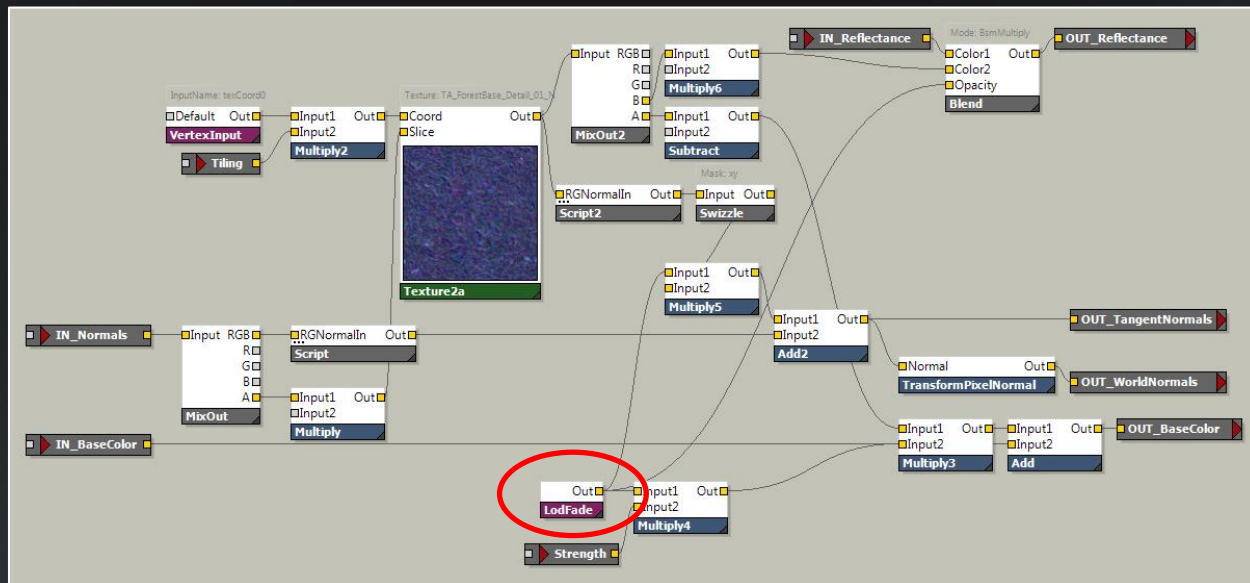
Texture Arrays

- Like most texture assets in Battlefront, we heavily pack into channels.
- The detail texture uses <u>RG</u> for normals, <u>B</u> for Reflectance, <u>A</u> for Color.

Detail Texture within the Texture Array
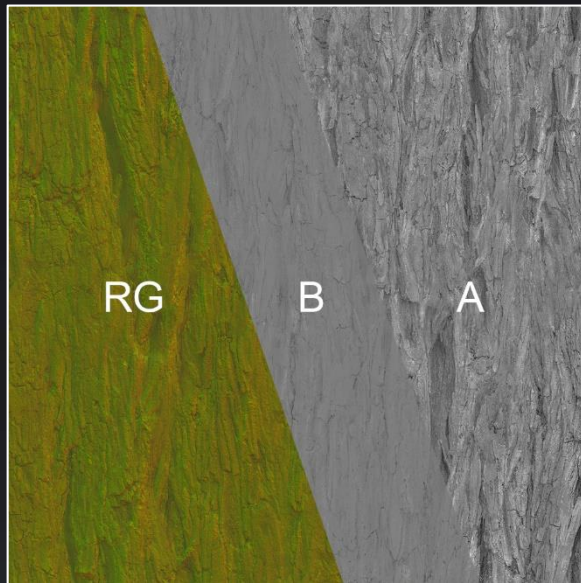
Texture Arrays

- Unique PhotoScan base is surface blurred, while keeping important details.

- Drastically reduces texture memory.

Texture Arrays

- Texture Array tiling detail on top fill in the details without overpowering the base.

- Drastically increases texture resolution.
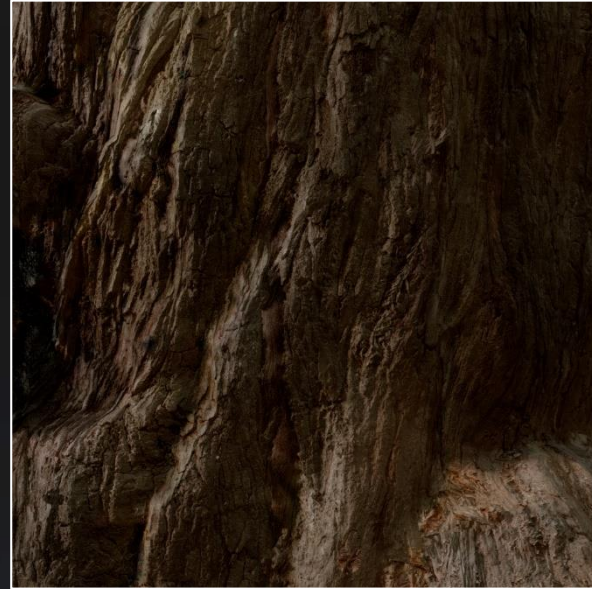
Photoscan vs. Everything Else

How do we ensure all assets fit together consistently?

"Old" Asset Creation   Photogrammetry   "New" Asset Creation

QUALITY   TECH   EFFORT

QUALITY   TECH + TECHNIQUES   EFFORT

NEEDED   QUALITY

TECH   EFFORT

TECH + TECHNIQUES   EFFORT

?

DICE

STAR WARS
BATTLEFRONT
EA
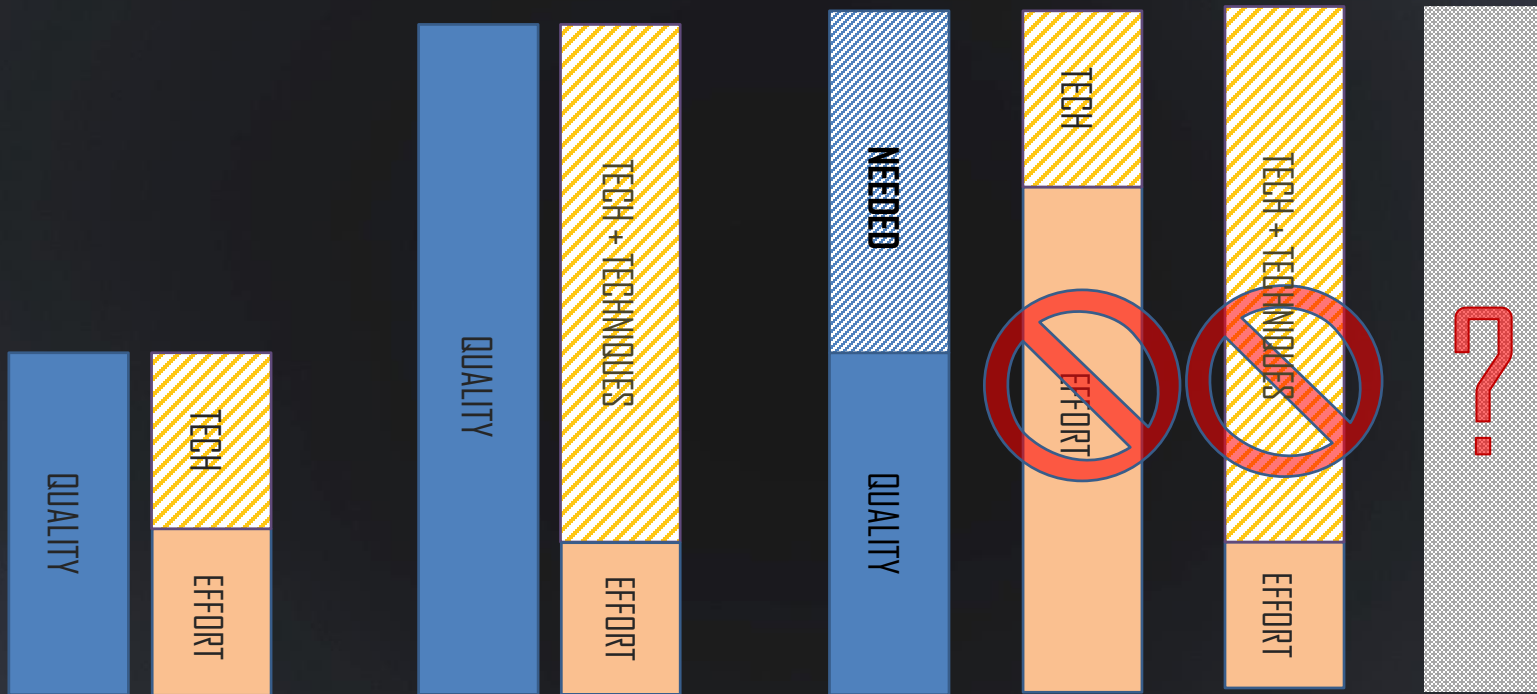
PBR
Quick Thoughts

- Helps us bridge that gap

- Lessons from Criterion, Guerilla Games, Dontnod etc

- Trying to get everyone to switch together can backfire.

- Some people will adapt instantly. Let them lead!

- For PBR, there isn't really a middle road.

PBR

- Assets generally fit well but still needed to be balanced against the scene.
- Finding and trusting true PBR values at times was a challenge
- Constant reinforcement helps prevent inconsistencies
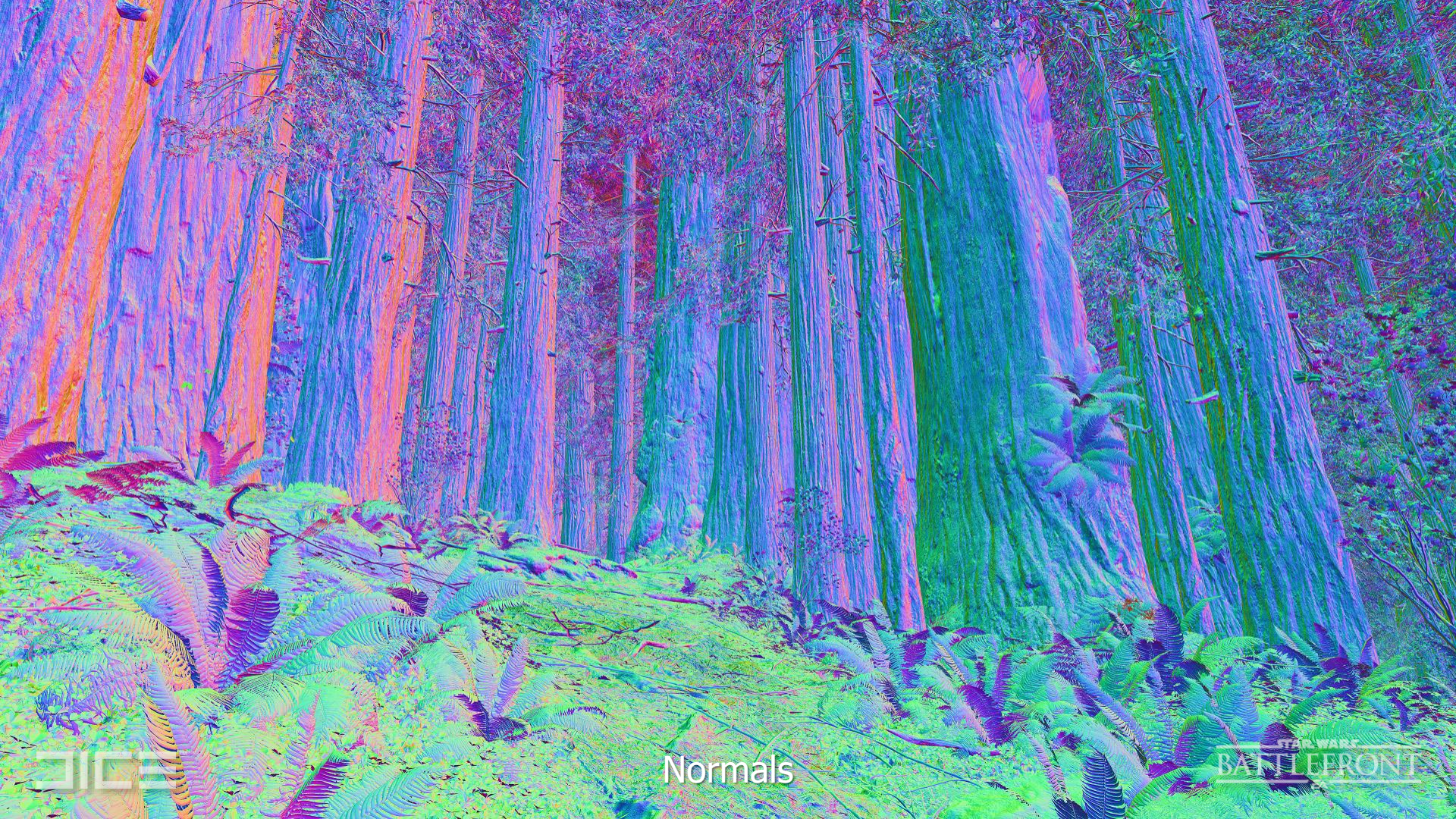
BaseColor

Normals

Reflectance

Smoothness

SubSurfaceTranslucency

Shadow

Ambient Occlusion

Light + Indirect
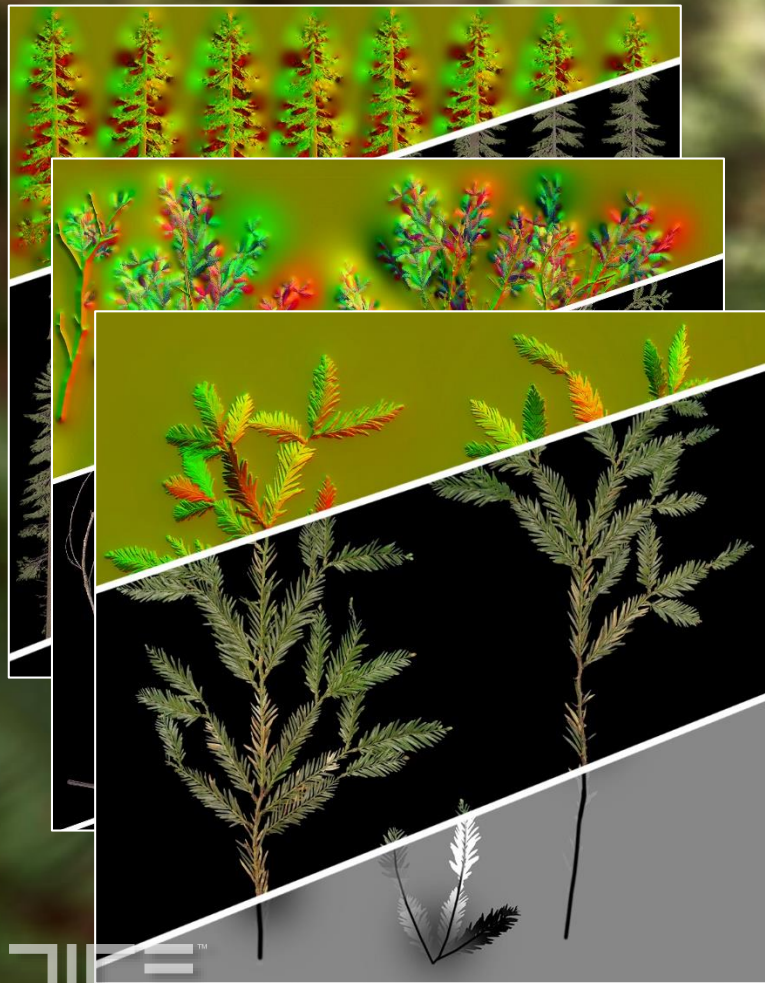
Vegetation

- Reaching 60fps – lets make a dense forest!
- Difficult to get good results from scanned vegetation – complex and noisy, resulting in horrible scans.
- Limited vegetation needs meant creating by hand was an option.
- Based on photos of vegetation from location captured on blue board.
- Integrating hand-built vegetation into a PhotoScanned world was tough to reach comparable quality!
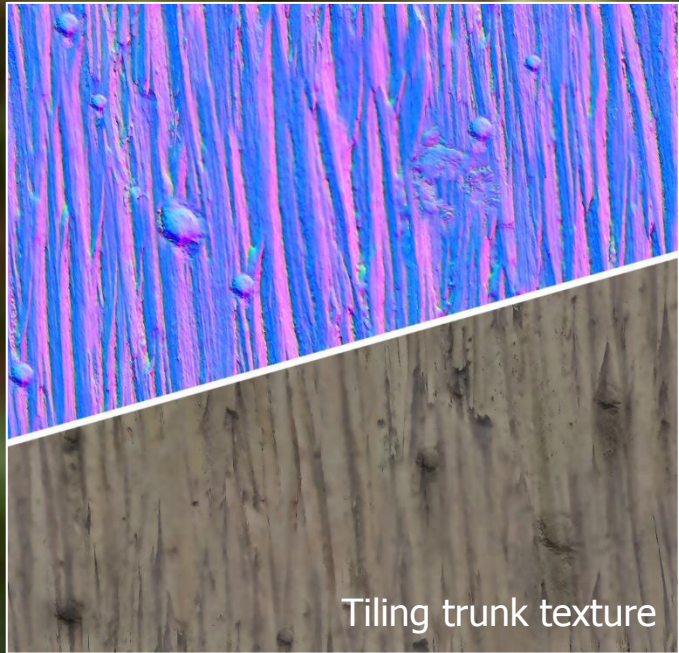
Vegetation

- All vegetation share the same textures
- There are textures created separately for close up, mid, and far distances.
- There is a lot of texture packing within vegetation assets to reduce complexity of shaders.
- MipMaps tweaked for visual control.
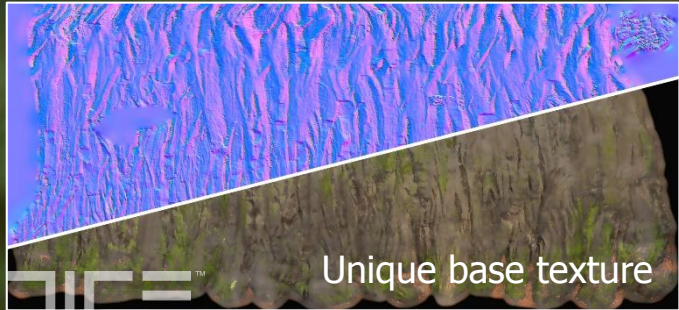- Normals created by broadly modelling around leaves and branches Maya.

Vegetation

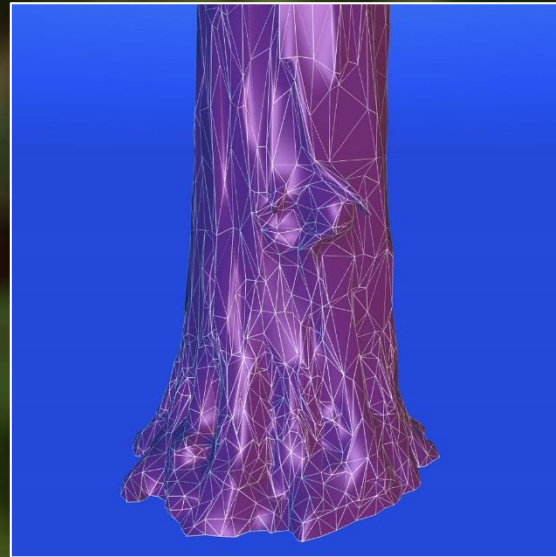Tiling trunk texture

Unique base texture

- Unique PhotoScanned base blends to tiling texture that extends upwards.
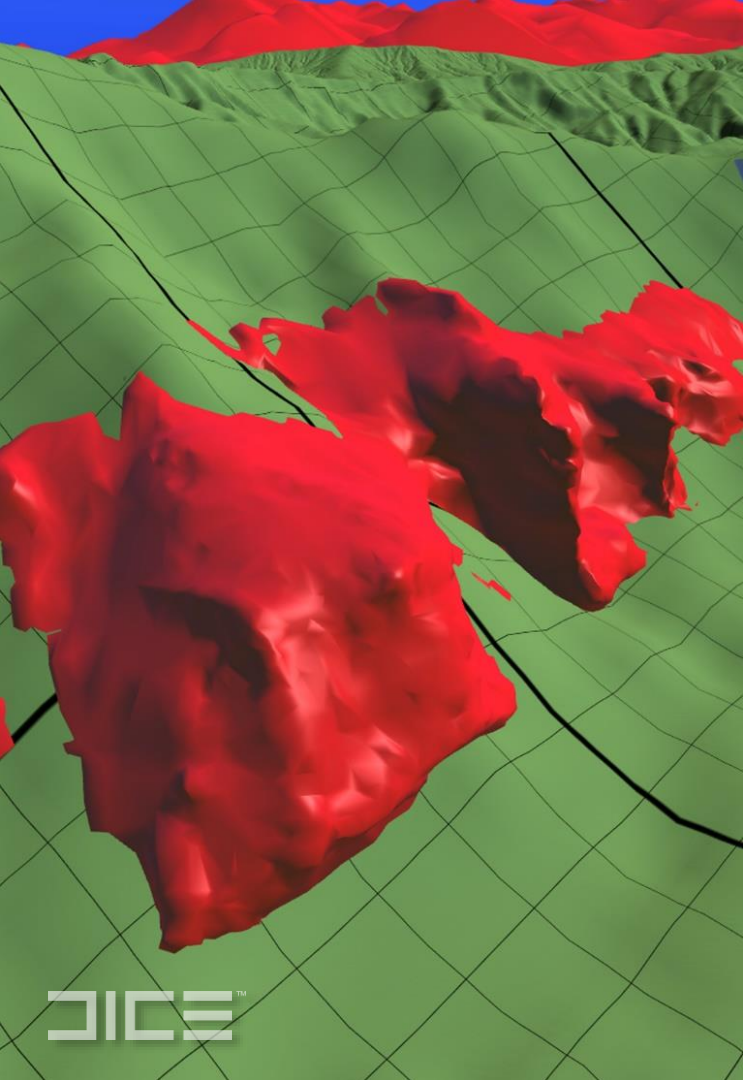- Tree base is generally evenly tessellated for best displacement results.

STAR WARS
BATTLEFRONT
EA

Terrain/asset blending

- A lot of work put in to reduce the visible disconnect between assets and terrain.

- Most assets have terrain radiosity projected on to them.

- Texture Arrays world projected and details match.

- Terrain color and normals applied to most assets.

Terrain/asset blending

- A lot of work put in to reduce the visible disconnect between assets and terrain.

- Most assets have terrain radiosity projected on to them.

- Texture Arrays world projected and details match.

- Terrain color and normals applied to most assets.

Mesh Displacement

- Great quality boost for close up detail.
- Object displacement stopped at ~7 meters.
- An ongoing balance between quality and performance.

Mesh Displacement OFF

Tessellation
Heightfield Displacement

- Ability to define displacement shape (via mask)
- One of our "Must Have" features
- Critical for ATAT footprints
- Terrain Decal Component
- Combined with detail displacement shader
- (More on this a little later)

STAR WARS
BATTLEFRONT
EA

Tessellation and Photogrammetry

Heightfield Displacement

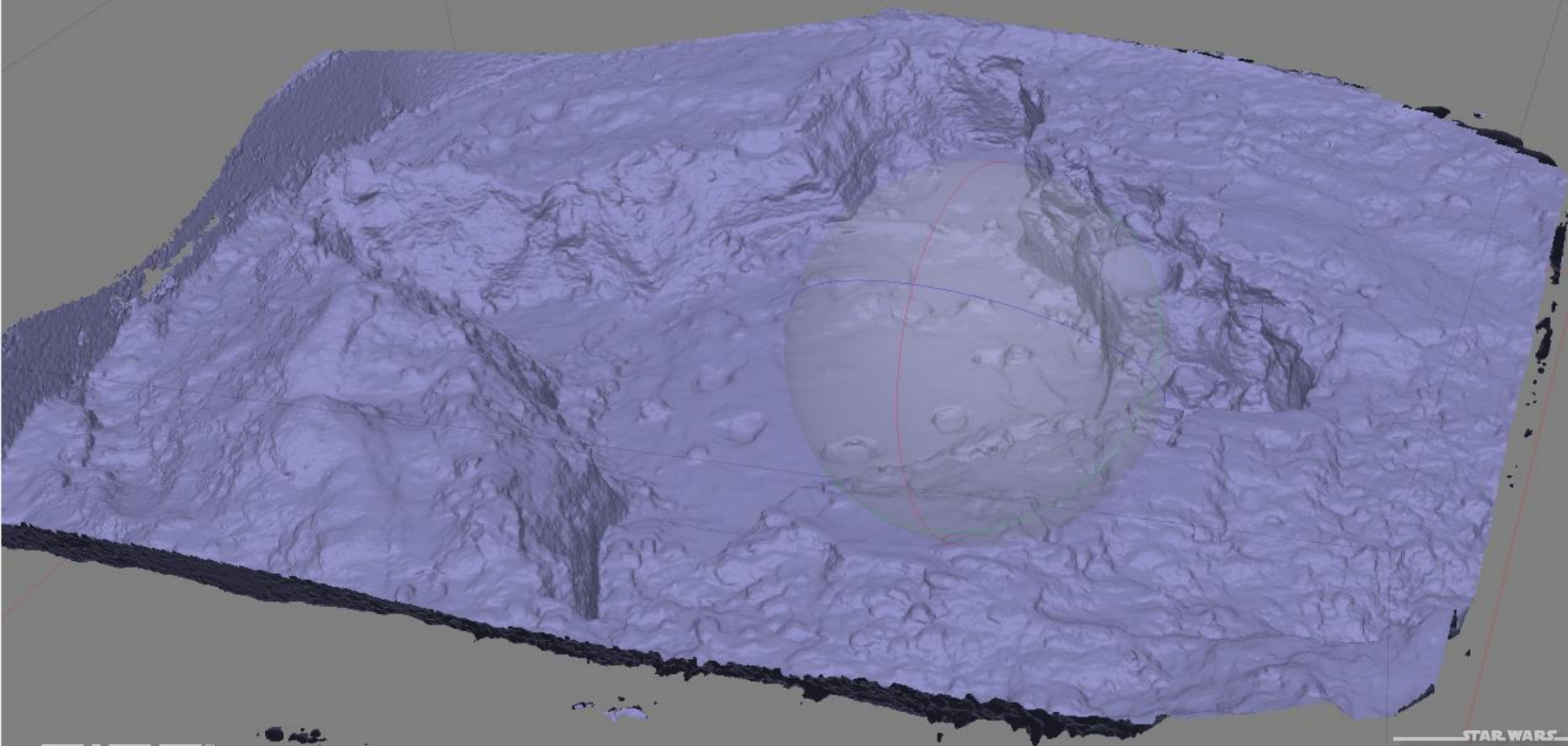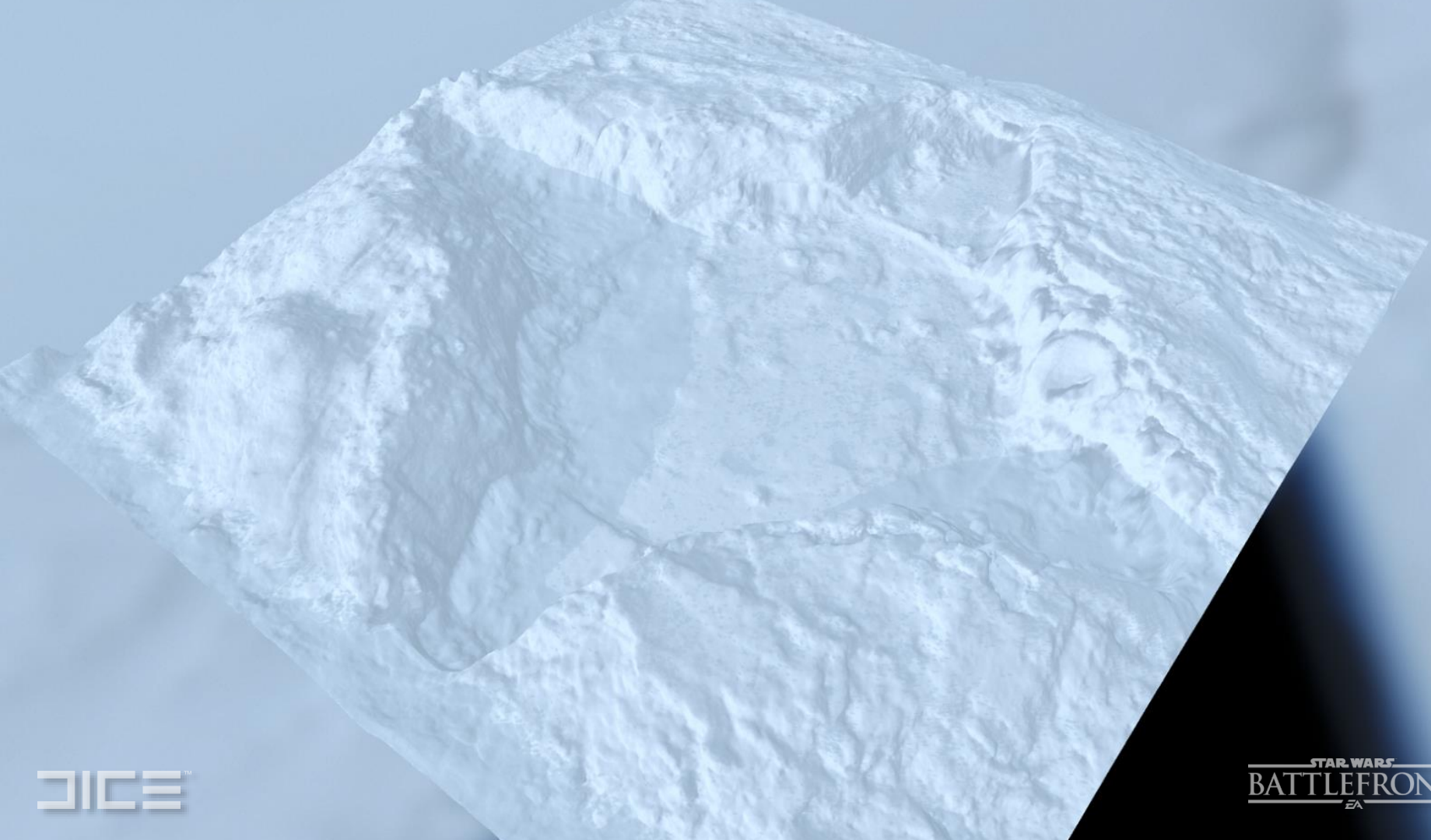- Quality vs. Speed ratio is extremely high
- Feel free to get creative

Terrain
Speaking of terrain.

DICE™

STAR WARS
BATTLEFRONT
EA

Terrain

Final big chunk of our "photogrammetry" puzzle.

Terrain

- 8x8km with a 1x1km playable space.
- Huge background terrain meshes.
- Terrain textures captured from real world.
- Heightmaps created from photoscanning for terrain tessellation.
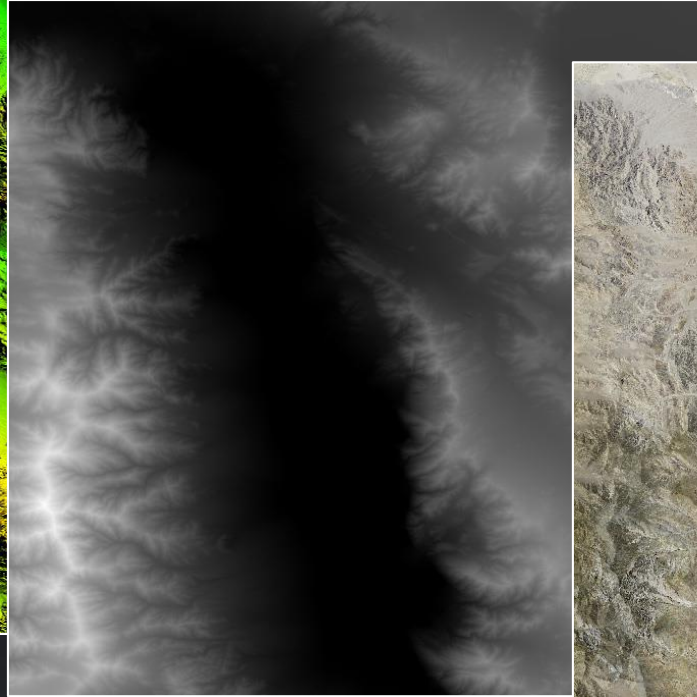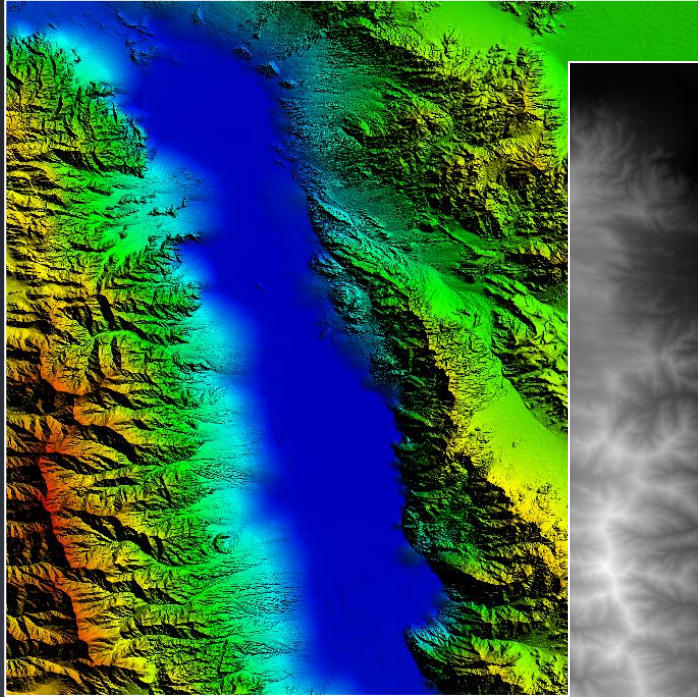- Layers limited to control draw calls.

Moving (mostly) away from WorldMachine..
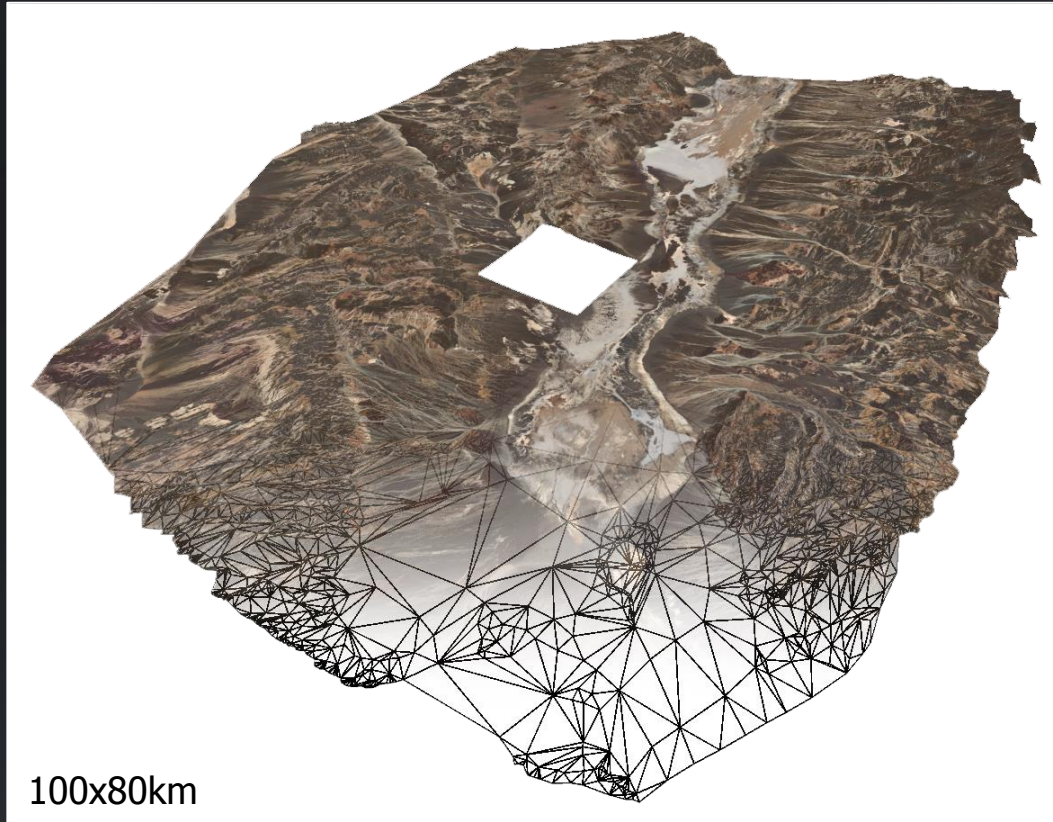
..to real-world topography data.

Processing the data

Processing the data
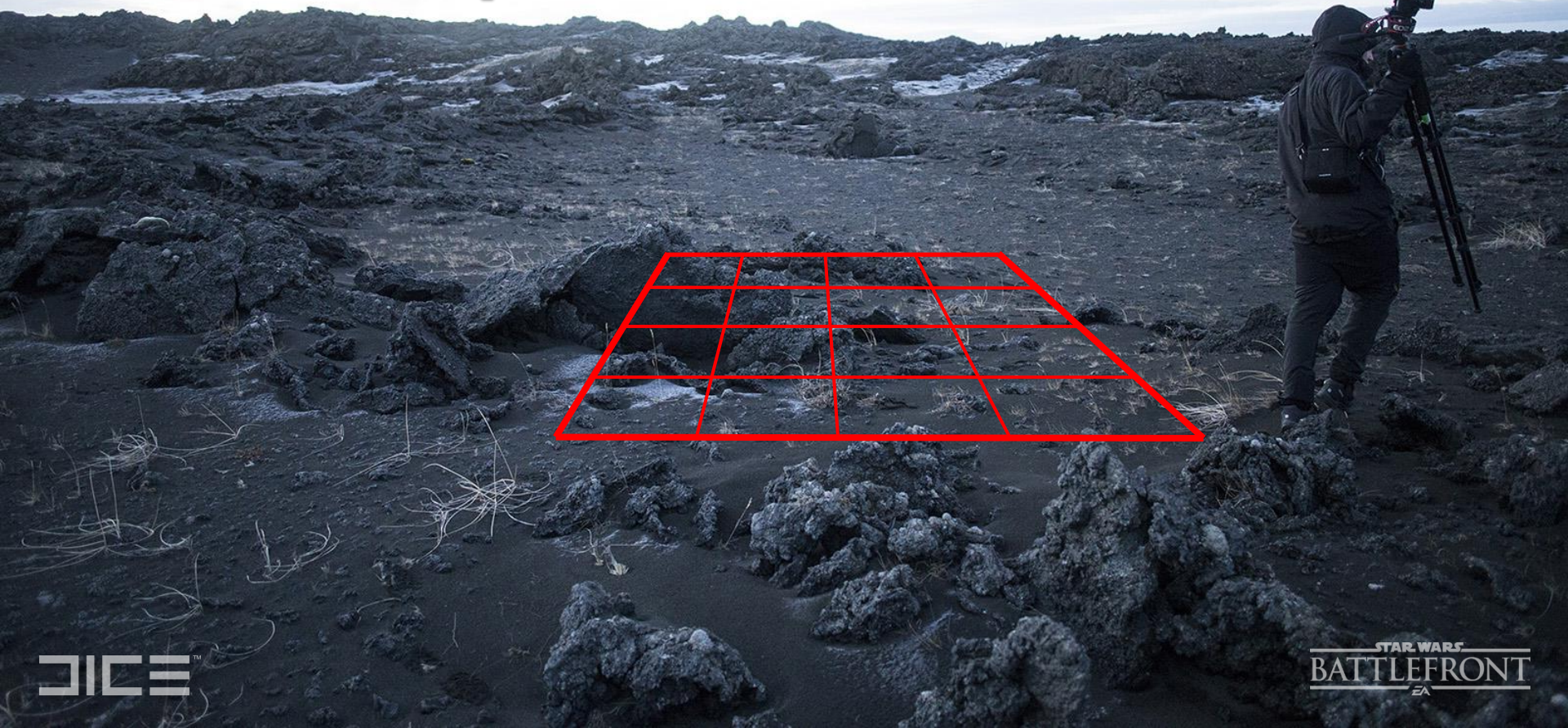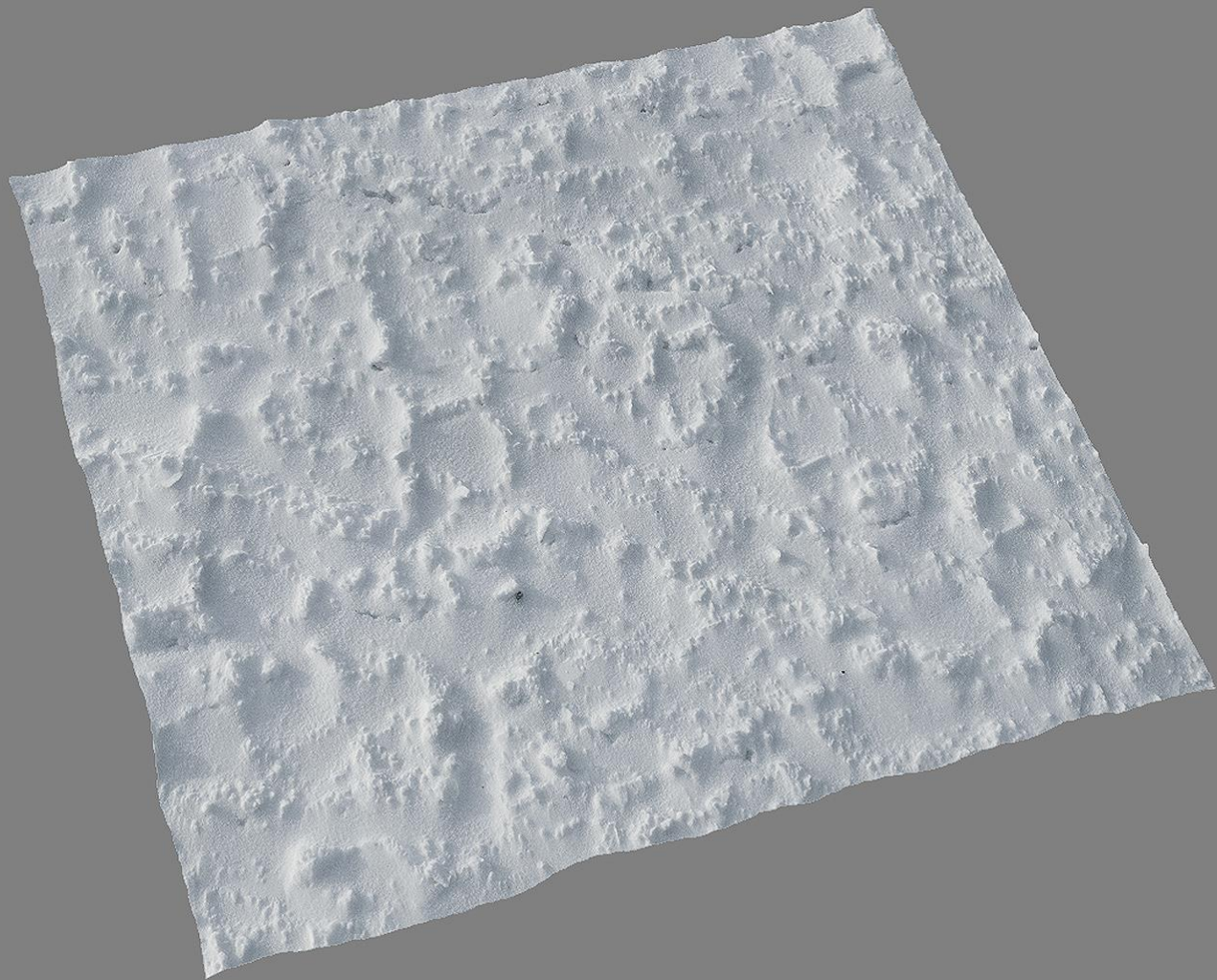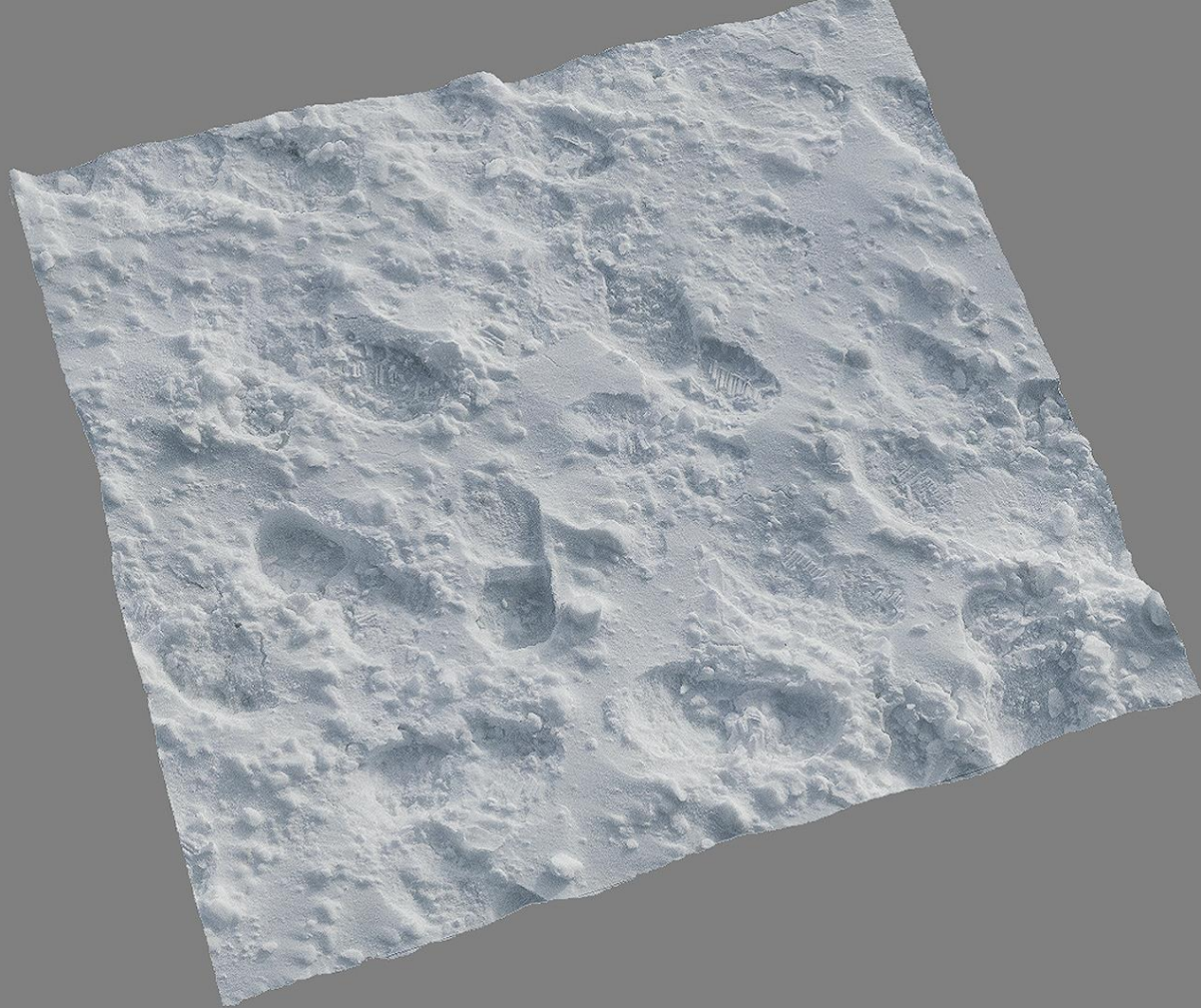
100x80km

Texture Capture

Terrain Tessellation

- Great quality boost for close up detail on terrain
- Major upgrade for us. Can never go back!
- Terrain displacement stopped at ~15 meters.
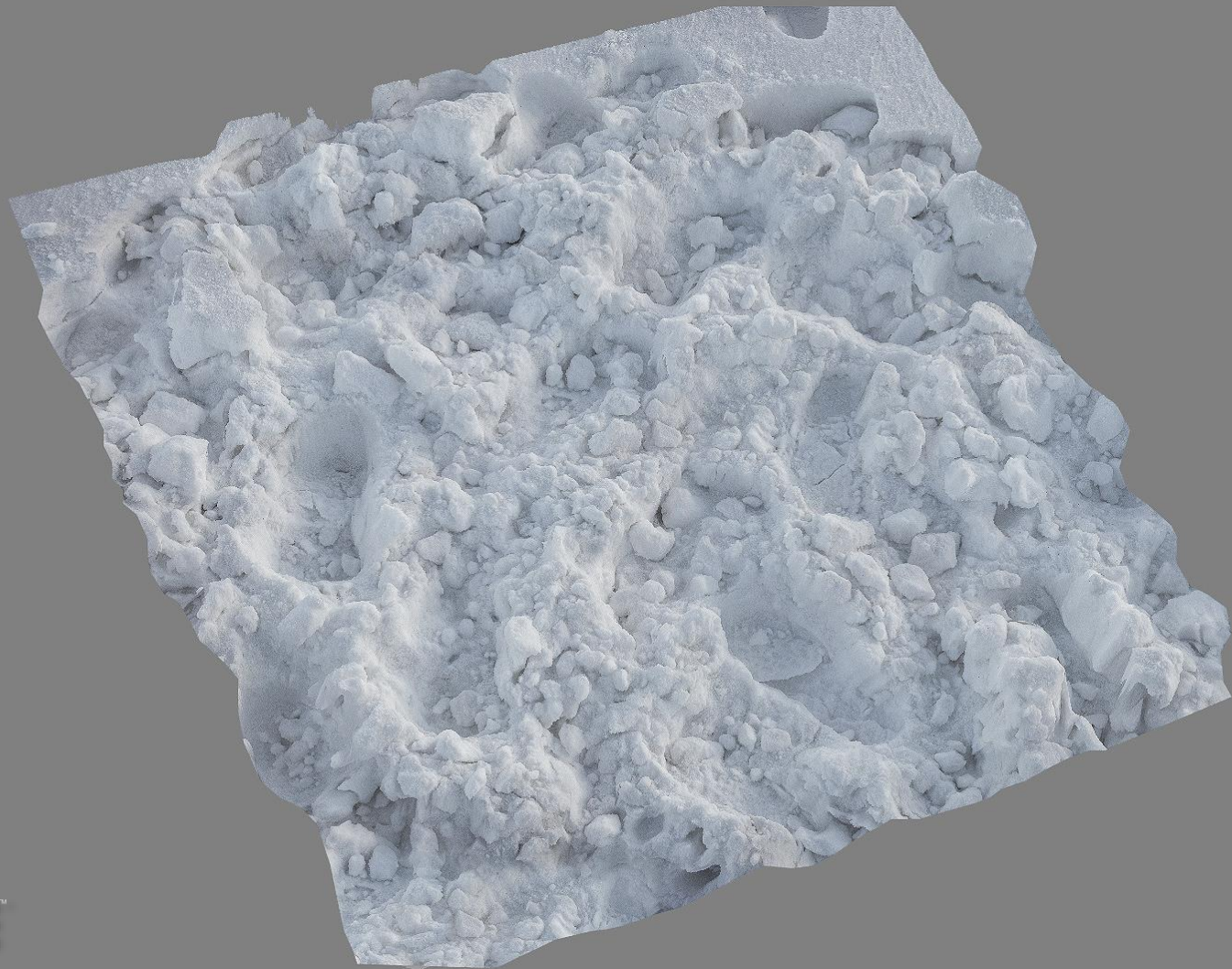
DICE

Terrain Tessellation OFF
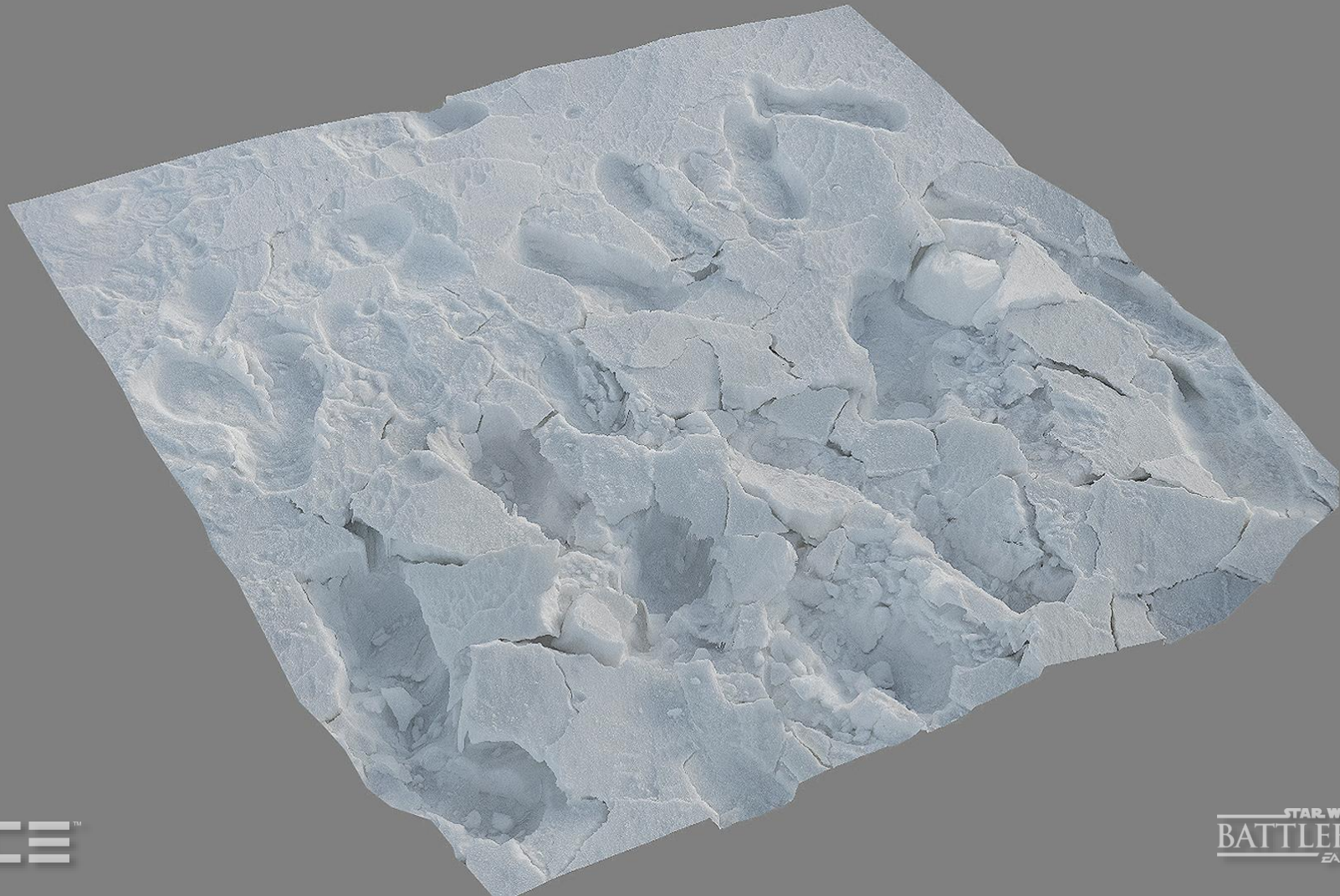
STAR WARS
BATTLEFRONT
EA

Terrain Tessellation ON

Best practices - Terrain

- We aimed for 3x3 meter patches
- 5x5 photo grid ensures enough overlap
- Capture only 1 defined element at a time
- Low ISO and high F-Stop for sharp image
- Avoid noisy surfaces such as grass

Level Construction Kits

- Content and documentation packages.

- Provides fast level iteration.

- Shifts focus to building levels rather than content during Production.

- Allows us to more quickly understand performance and keep track of it.

Level Construction Kits

- 8x8km highly detailed terrain.

- Ready to paint terrain materials.

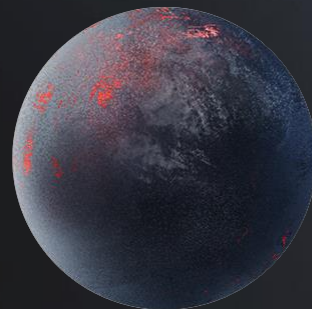- Easily accessible limited library of content.

- Lighting presets based on the films.
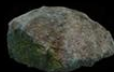
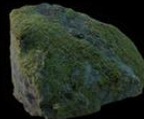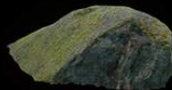Endor   Hoth   Tatooine   Sullust

Endor

DICE

STAR WARS
BATTLEFRONT
EA

Endor
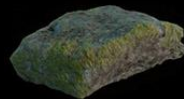
Hoth

Hoth

Tatooine
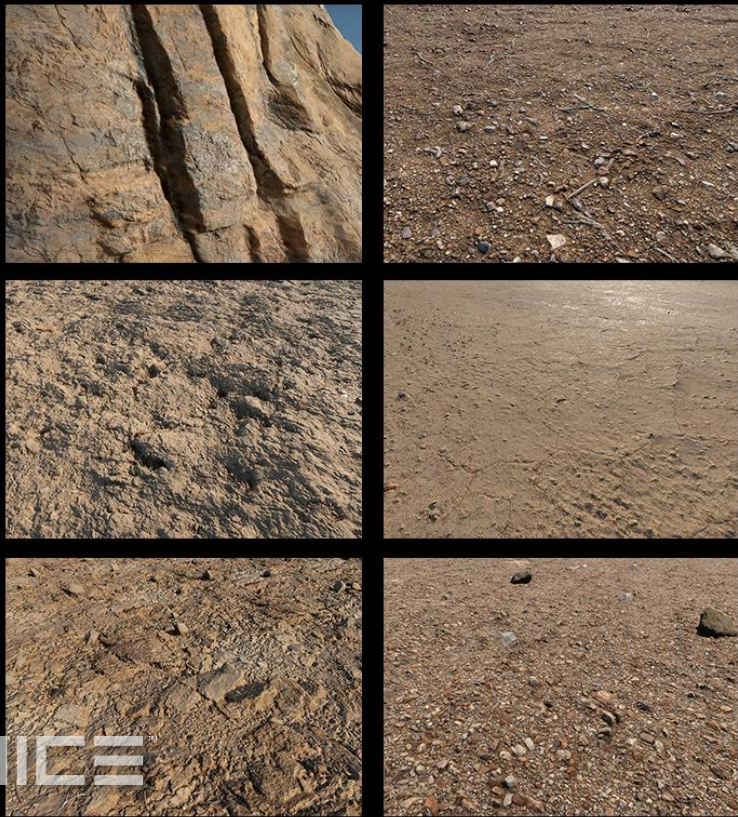
Sullust

DICE™

STAR WARS
BATTLEFRONT
EA

Sullust
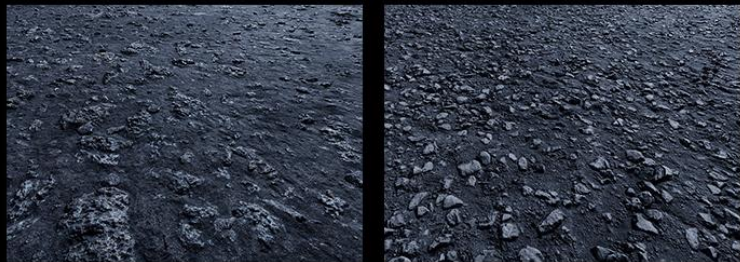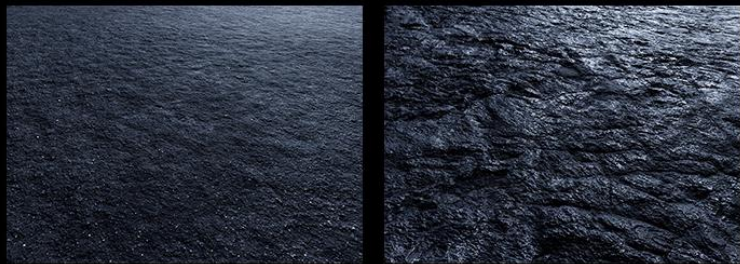
Key learnings and takeaways

- Solid processes and workflows in place before Production was a big win.

- Well planned and structured Photogrammetry trips results in clean and manageable assets.

- Photogrammetry is not a silver bullet – it was only one part of achieving the final results in game.

- Consistency is key when working with scanned assets – it was crucial to have artists own and be responsible for their locations/planets.

- Working towards PBR was an ongoing process to success.

- Much of our creative process was incredibly repetitive, much of which can and should be automated.

- There was way too much manual labor, with levels being built by brute force without the help of workflow-improving tools.

STAR WARS
BATTLEFRONT
EA

Special Thanks

John Troive - Senior Environment Artist
Pontus Ryman - Senior Environment Artist
Petter Skold - Senior Environment Artist
Tim McLeod - Environment Artist
Simon Barsky - Environment Artist
Darko Pracic - Environment Artist
Daniel Cambrand - Environment Artist
Oscar Carlen - Senior Lighting Artist
Madina Chionidi - Character Artist
Björn Arvidsson - Character Artist
Robert Kihl - Rendering Engineer

... and of course, the entire Battlefront team at DICE!

Thanks!

Andrew Hamilton – Lead Environment Artist
Andrew.Hamilton@dice.se

Kenneth Brown – Technical Art Director
Kenneth.Brown@dice.se