

GHOST OF TSUSHIMA

Zen of Streaming: Building and Loading Ghost of Tsushima



- Adrian Bentley
- Coding Team Lead at Sucker Punch Productions



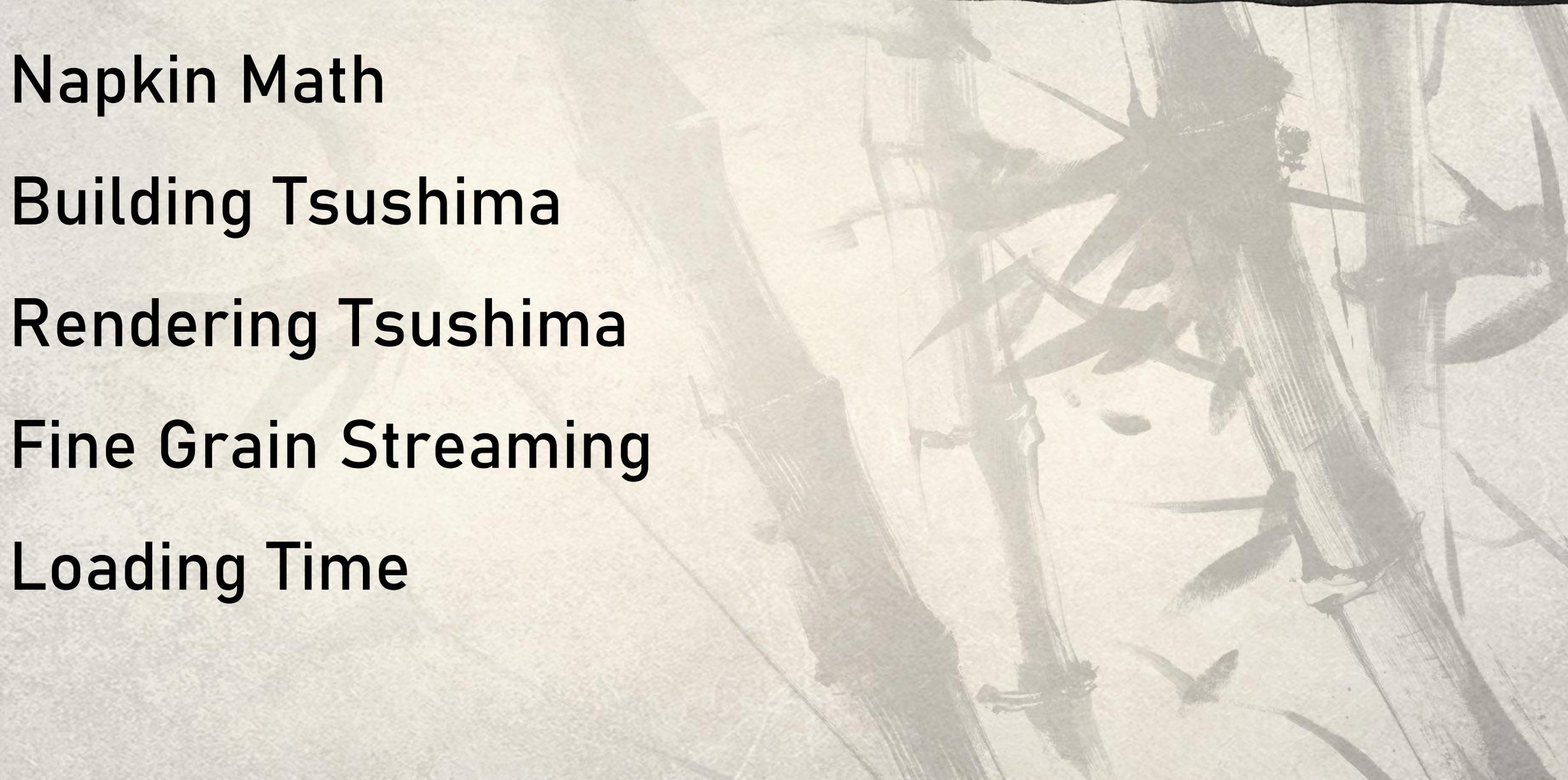
Napkin Math

Building Tsushima

Rendering Tsushima

Fine Grain Streaming

Loading Time



Simulating Wind in Ghost – Bill Rockenbeck – GDC 2021

Samurai Landscapes – Matthew Pohlmann – GDC 2021

Procedural Grass in Ghost – Eric Wohllaib – GDC 2021 Advances

Lighting & Rendering in Ghost – Jasmin Patry – SIGGRAPH 2021

NAPKIN MATH

Defining the Budgets

ONCE UPON A TIME...

inFAMOUS

SECOND SON™







inFAMOUS
**SECOND
SON**

ENGINE POSTMORTEM

By Adrian Bentley

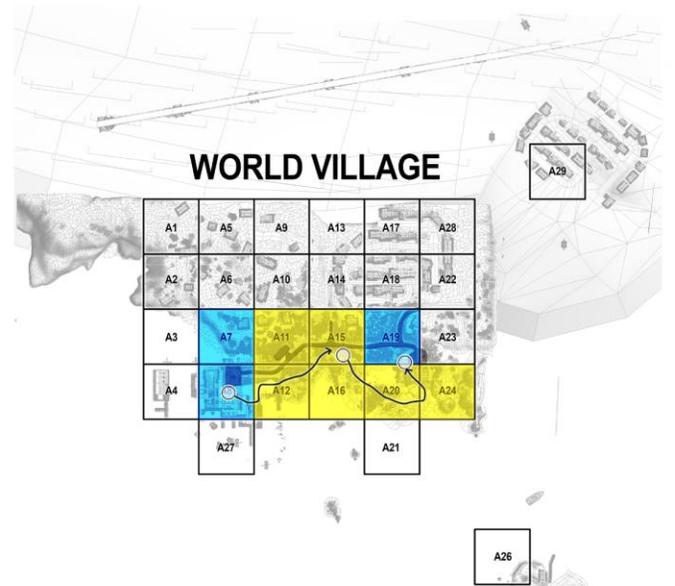
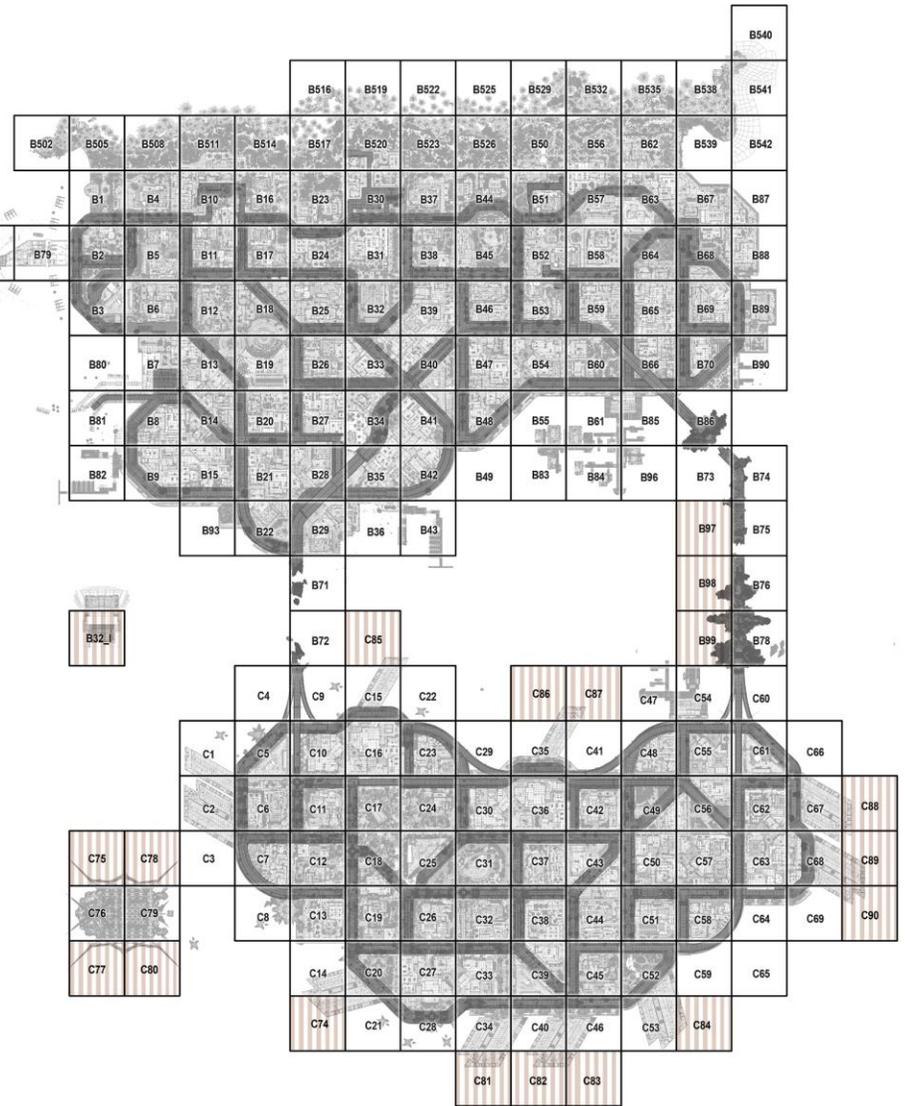
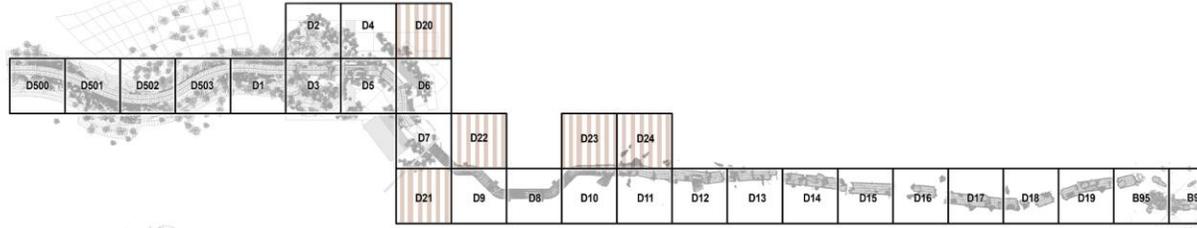
Lead Engine Programmer

MSN_VOR - VANDAL ON THE RUN

10-2-13 DESIGNER - JEFFM

- TRAVERSAL BLOCKS
- CHECKPOINT BLOCKS
- IMPOSTERLESS BLOCKS

WORLD SEAPORT



~250 100m Tiles

Pack Reports

Thu Feb 20 13:40:03 2014

Description	Game Data Size and Reports		Model Size and Reports		Texture Size and Reports	
Worlds	16M		30M		2M	
World Seaport	11.01M (4.99M free)	by size / dir / users	26.99M (3.01M free)	by size / dir / users	1.78M (224K free)	by size / dir / users
World Village	2.36M (13.64M free)	by size / dir / users	11.47M (18.53M free)	by size / dir / users	690K (1.33M free)	by size / dir / users
Common Core	57M		53M		195M	
Core Common	0 bytes (57M free)	by size / dir / users	0 bytes (53M free)	by size / dir / users	0 bytes (195M free)	by size / dir / users
Core	9M		18M		94M	
Core Seaport	0 bytes (9M free)	by size / dir / users	0 bytes (18M free)	by size / dir / users	0 bytes (94M free)	by size / dir / users
Core Village	0 bytes (9M free)	by size / dir / users	0 bytes (18M free)	by size / dir / users	0 bytes (94M free)	by size / dir / users
Core Common Subtotals	57M		53M		195M	
Env Textures	52K	by size / dir / users	0 bytes	by size / dir / users	96.9M	by size / dir / users
Env Models	1.58M	by size / dir / users	6.46M	by size / dir / users	2.43M	by size / dir / users
Env Missions	192 bytes	by size / dir / users	0 bytes	by size / dir / users	0 bytes	by size / dir / users
Subtotal for Environment	1.63M (379K free)		6.46M (1.54M free)		99.33M (682K free)	
Char Male Ped	13.61M	by size / dir / users	13.5M	by size / dir / users	24.9M	by size / dir / users
Char Female Ped	5.34M	by size / dir / users	14.54M	by size / dir / users	22.57M	by size / dir / users
Char	27.68M	by size / dir / users	386K	by size / dir / users	4.51M	by size / dir / users
Subtotal for Characters	46.62M (391K free)		28.41M (606K free)		51.97M (29K free)	
Ui	1.12M	by size / dir / users	292K	by size / dir / users	27.29M	by size / dir / users
Ui Fx	6.66M	by size / dir / users	15.23M	by size / dir / users	15.79M	by size / dir / users
Subtotal for UI/FX	7.78M (224K free)		15.51M (500K free)		43.08M (84K over)	
Misc	192 bytes	by size / dir / users	0 bytes	by size / dir / users	0 bytes	by size / dir / users
Subtotal for Misc	192 bytes		0 bytes		0 bytes	

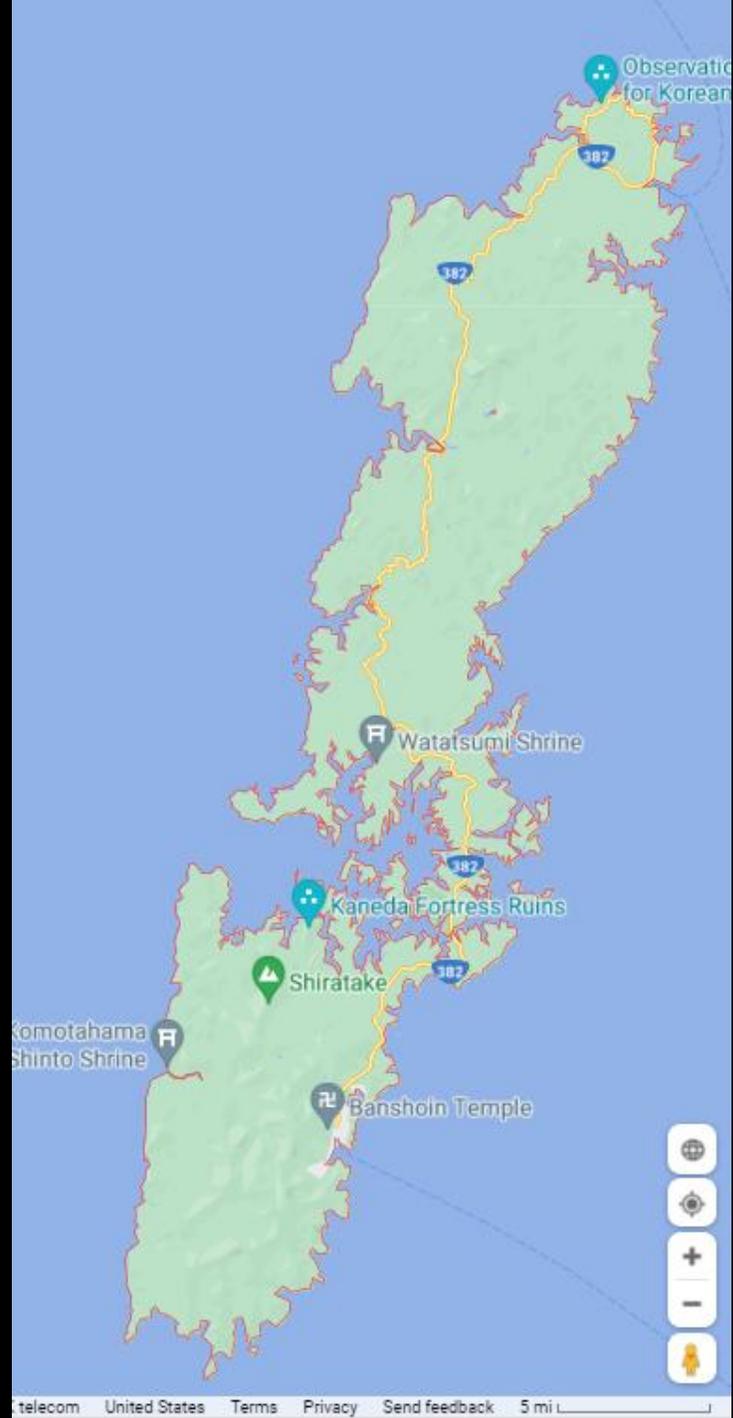
SECOND SON DISC SIZE

File Type	Size (uncompressed)
Texture	8.7G
Game+Mesh	7.4G
Cutscene	4.9G
Movies	3.4G
Music	1.4G
Dialogue	0.4G (x10)
Core	0.3G
Total	31G (24.5G compressed)
Budget	Size (uncompressed)
Textures Per Tile *	48M
Tile	22M
World TOD all tiles (~250)	616M
Total per tile	~72M



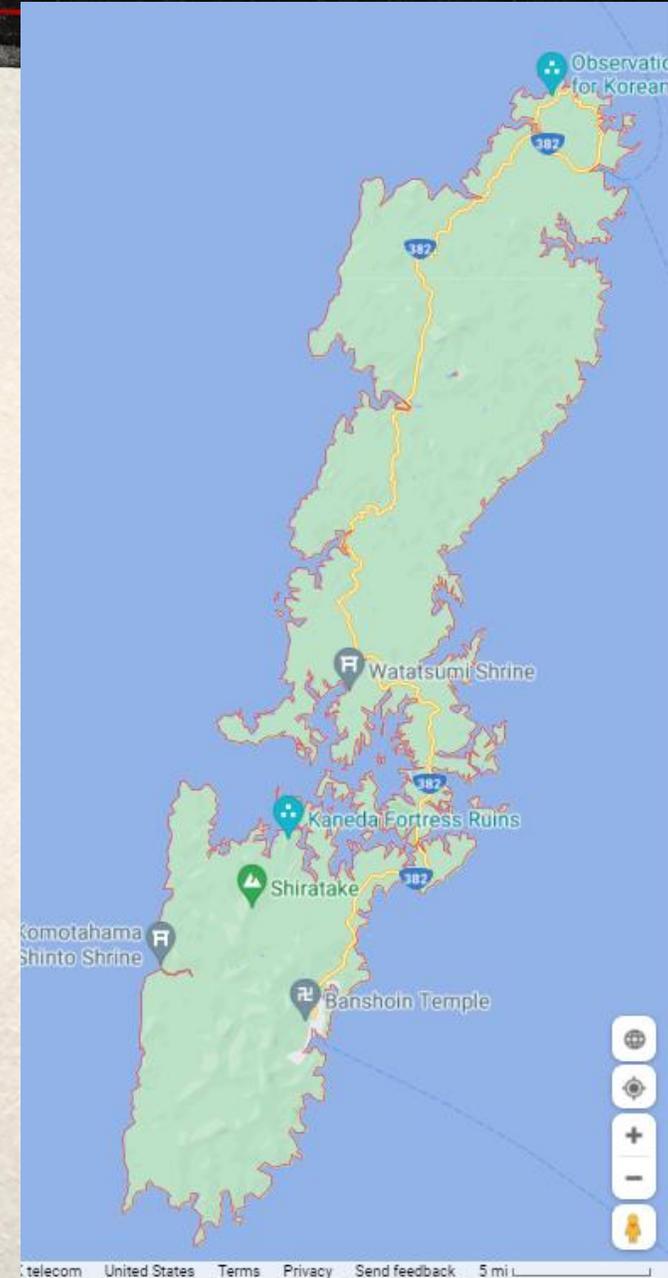






~~Tsushima ~708 km²~~

Guess ~64 km²

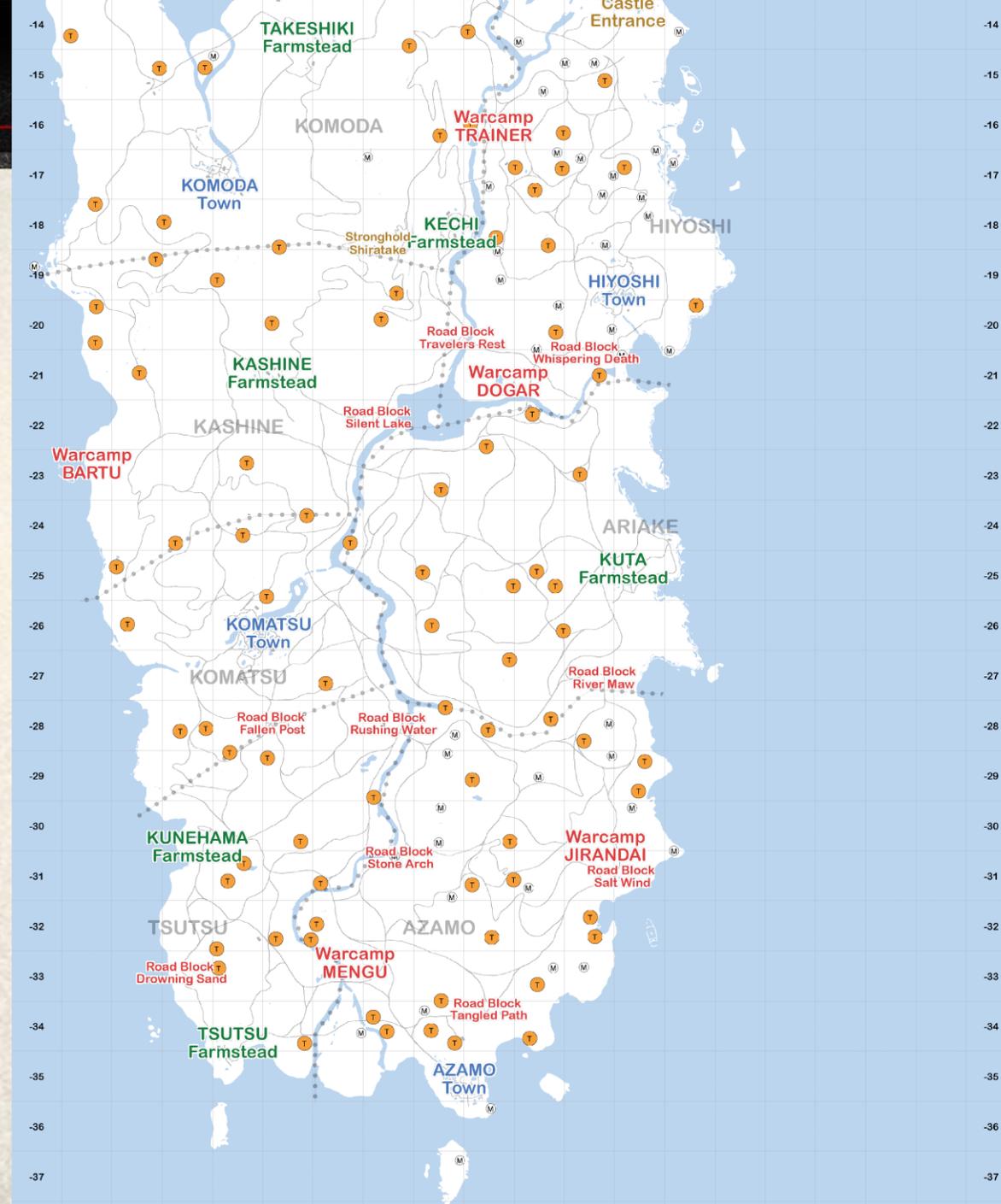


NAPKIN MATH

So 6400 100m tiles x 72M

= 460G!

Or 4M/tile





USE [R] TO LOOK AROUND

Trees in a forest = ~2500

Plus bushes, flowers, etc. 10k+ instances.

All LODs for tree = 0.6k - 1.2k = 12M total

Grass or similar stuff - 100k+/1m+

MORE ESTIMATES

Content	Tile	Optimized
Terrain	3M	2.5M
Terrain Physics	24M	0M
Vegetation	12M	1M
Lighting	0.5M	0.11M
Pathing	6M	3M

~~Merged BSPs for physics~~

~~Per lod per shader bounds for rendering~~

~~Occlusion by large buildings~~

~~Reusable city streets, side walks, buildings~~

~~Manually placed assets~~

~~Manually authored pathing~~

Aggregated packs?



Large scale/robust editing

Procedural tools

Better mission tools

Optimize instance overhead

More data into shared core

Needed fine grain streaming?

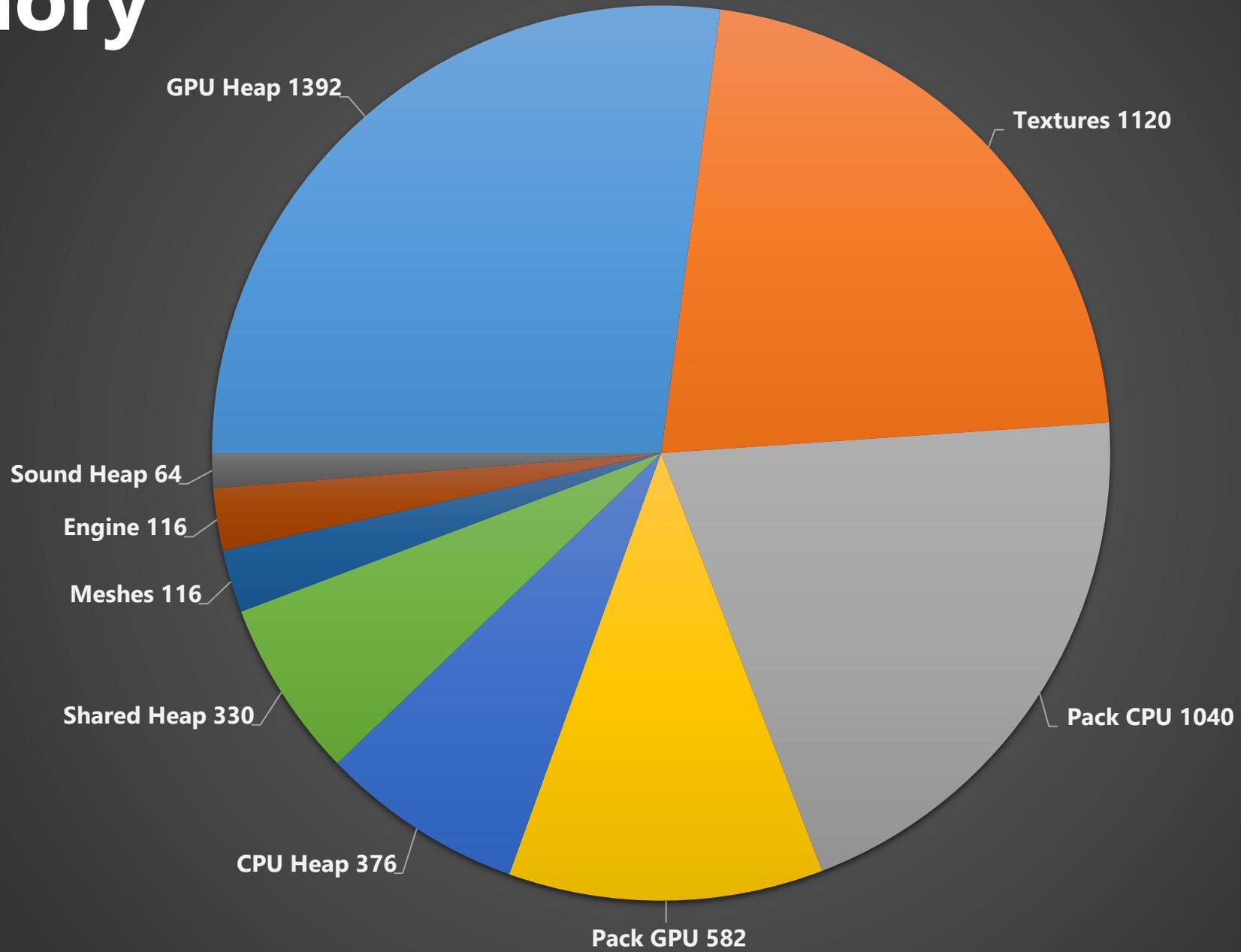
Where did we end up?



Maya	40k - 66k files	224G - 371G
Texture Bundles	54k - 60k files	230G - 610G
Wav/Loc	280k - 1m files	101G - 320G
Text	65k	80G
Movie	330	12G
Total	0.5m - 1.6m files	700G - 1.5T

File Type	Size (uncompressed)
Texture	17.9G
Game+ Mesh	2.8G
Terrain	5.2G (1214 tiles)
Streamed Mesh	0.9G
Movies	6.8G
Cutscene	2.5G
Music	2.1G
Dialogue	2G (x9 eu)
Total	56.2G (34.8G compressed)

Memory

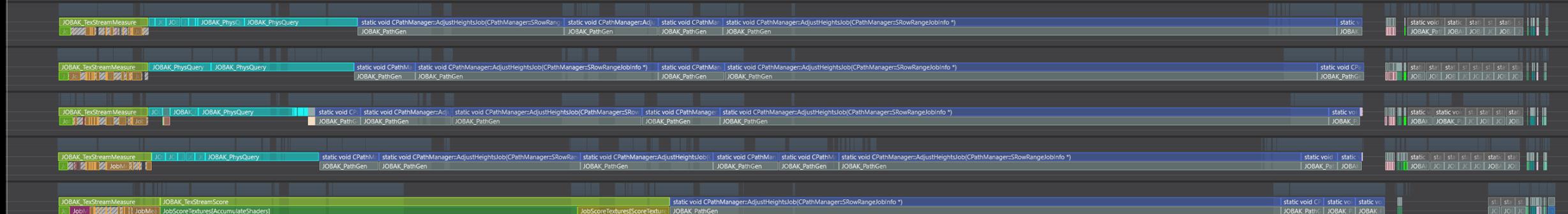
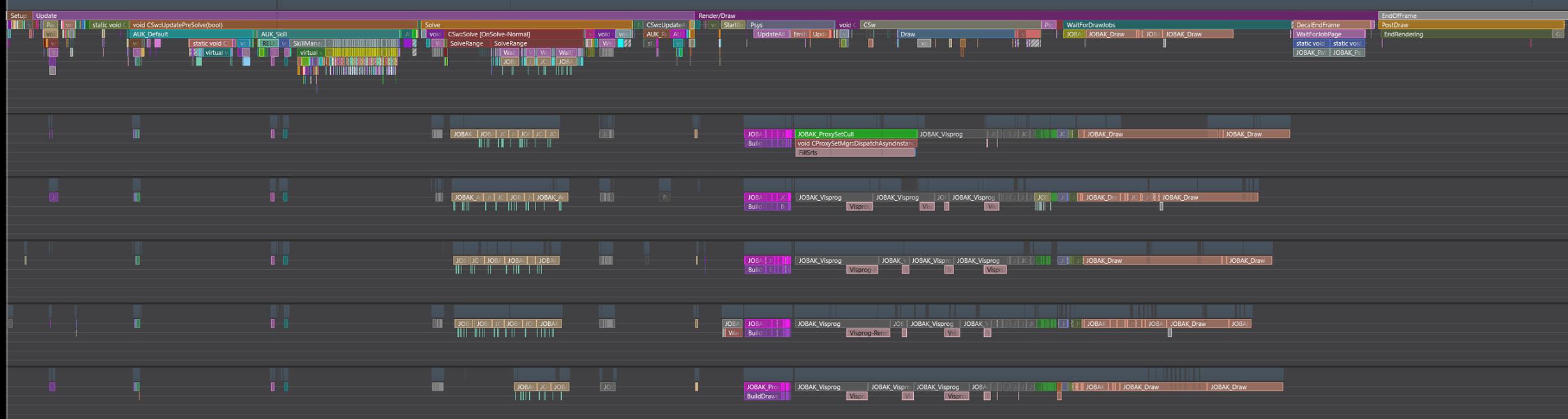
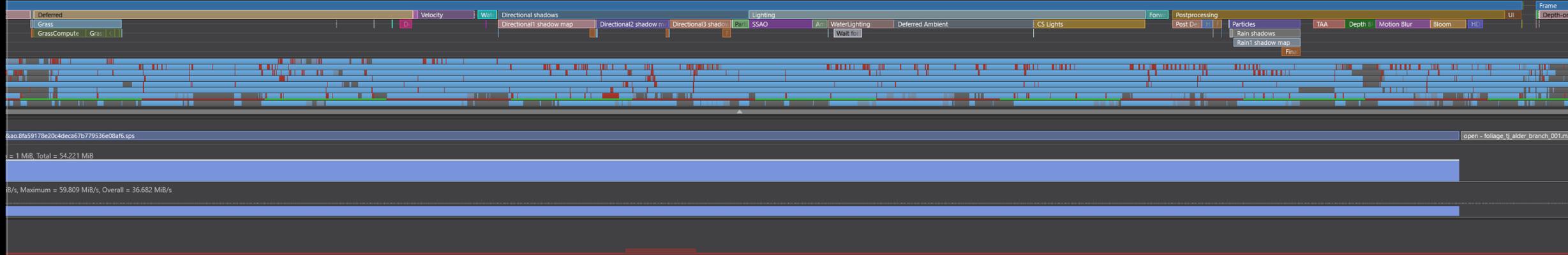


Similar structure to Second Son

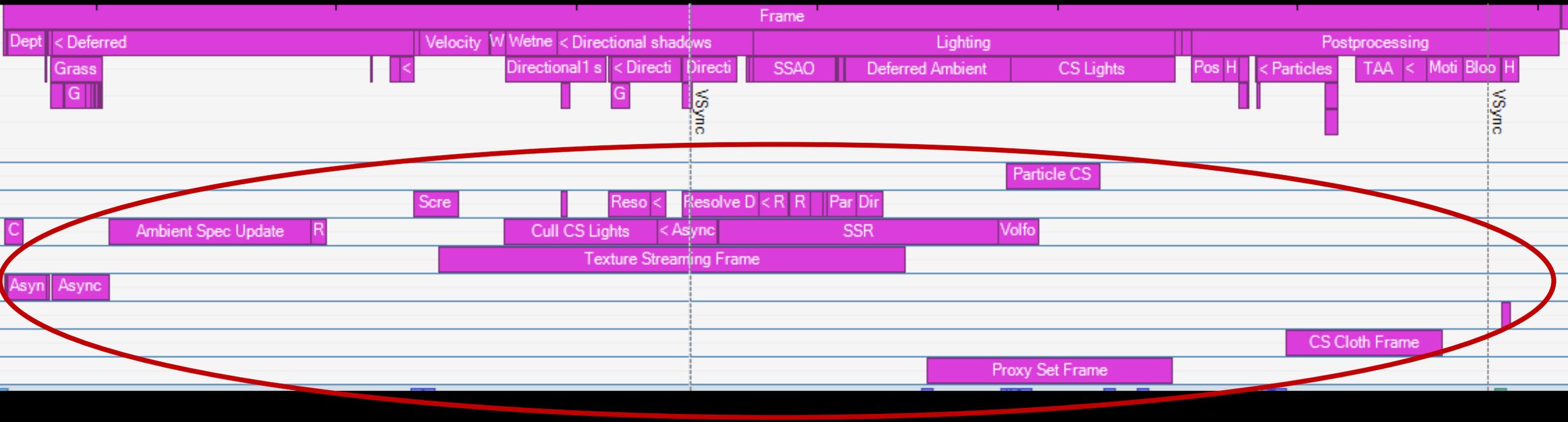
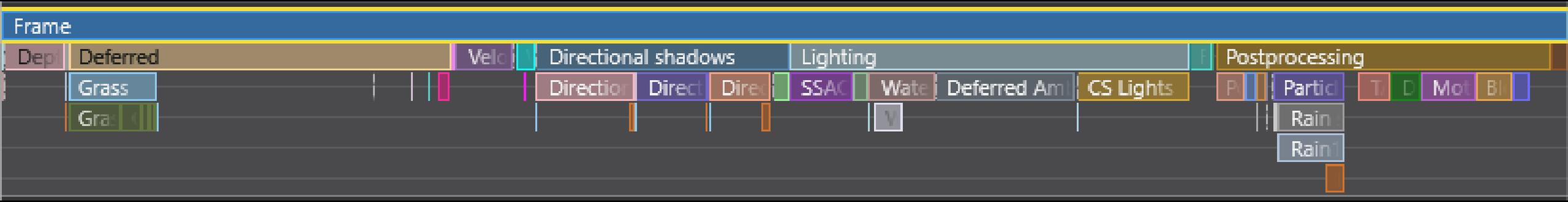
- ◆ 30hz 6.5 cores (+0.5)
- ◆ 1 main thread + 5 job threads (+5 new low priority)
- ◆ Phases – Update (AI), Solve(anim+physics), Render

More threading

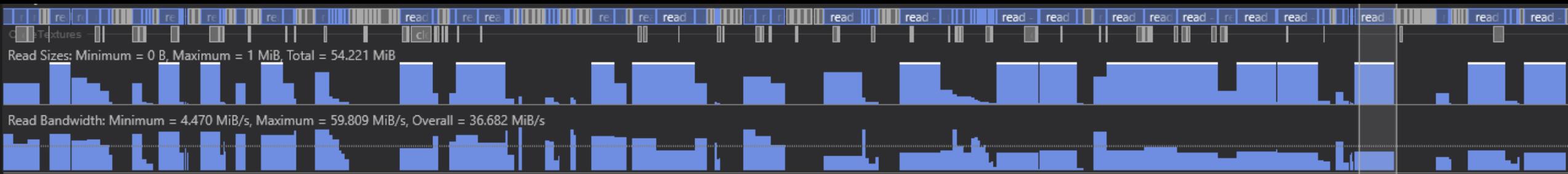
More GPU async compute



GPU Frame



File IO



Shipping a game in hard mode!

AWS – VPN, bandwidth, caching

Minimize sync/build

Simple/fast(ish) build not reliant on distribution

A painting of a landscape with a torii gate and a path, overlaid with the text 'BUILDING TSUSHIMA'. The scene is rendered in a painterly style with visible brushstrokes. The torii gate is a simple wooden structure with a white offering (shide) hanging from it. The path leads through a field of tall grasses and reeds towards the gate. The sky is a mix of blue and white, suggesting a bright, slightly hazy day. The overall mood is serene and contemplative.

BUILDING TSUSHIMA

Art at Scale

FLUX 1.0

File Settings Help

World

Toggle World Map

Deformers Materials

Open Stamp Library

Open Deformer Organizer

Edit Global Deformer Groups

Refresh Deformers

Additive Combine

Max Combine

Duplicate Selected

Delete Selected

Invert Selected

Lock

Unlock

Center On View

Center On Selected

View Shading Lighting Show Renderer Panels

Verts:	157209	16641	0
Edges:	341066	33024	0
Faces:	154656	16304	0
Tris:	309312	32768	0
UVs:	157209	16641	0

Backfaces: N/A
Smoothness: N/A
Instance: No
Display Layer: default
Distance From Camera: 36176.297
Selected Objects: 1

Center On Selected

Center On View

persp

1.6 fps

Attribute Editor

List Selected Focus Attributes Show TURTLE Help

terrain_LOD_Shape1 terrain_patch_h_center_ofMusdeSystem1 terrain_

mesh: terrain_LOD_Shape1

Focus Presets Show Hide

FLUX Stamp Library

Library Tools

Select Load Attributes Copy Tab

In-engine painting

GPU baked procedural growth/texturing

GPU compute occlusion/rendering

Virtual texturing

Samurai Landscapes

Building and Rendering Tsushima Island on PS4

MattP GDC 2021



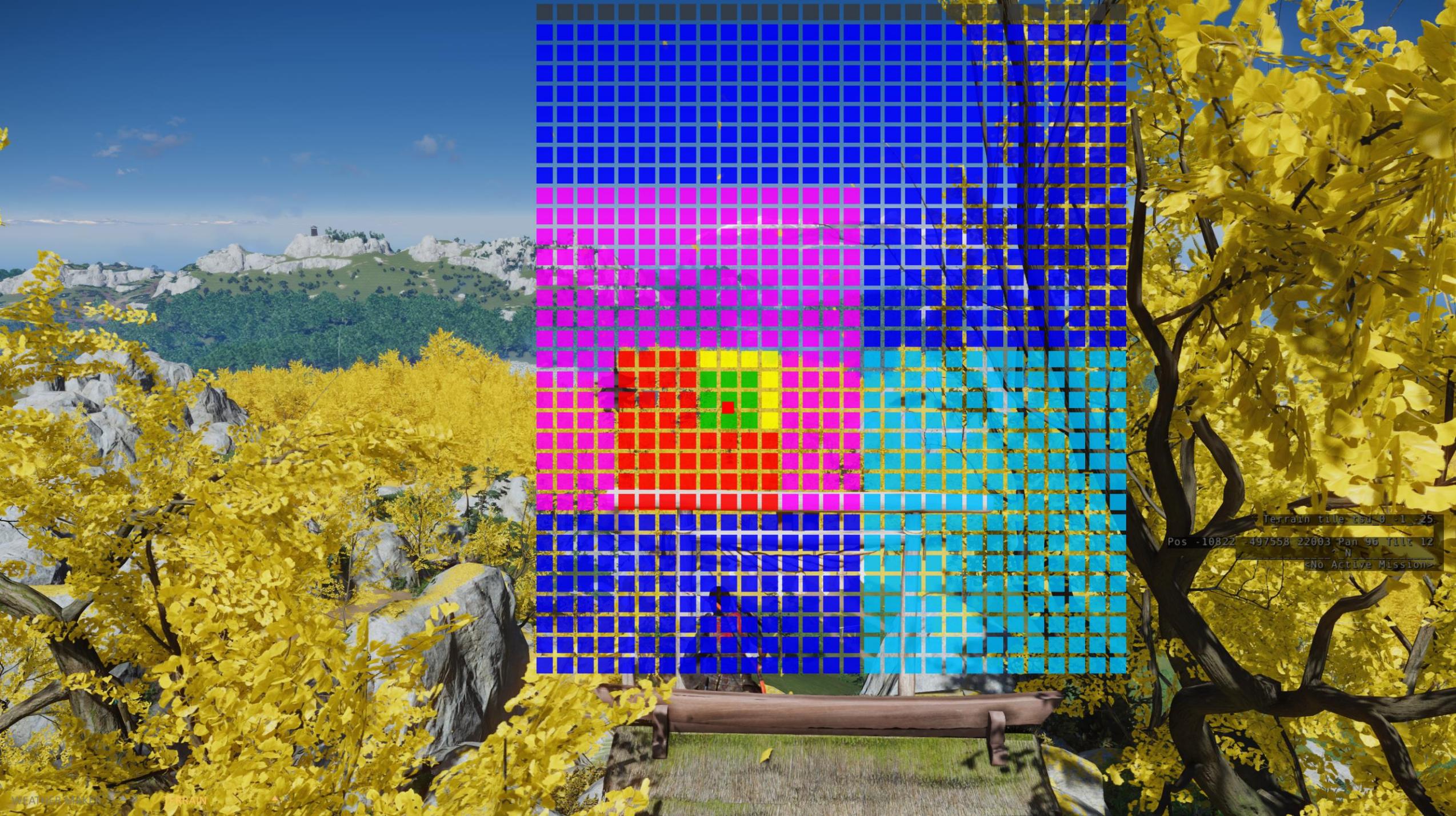


Where am I?

Tiles 200m ~6M budget (~1300)

~2.5M of 513x513 maps (~0.4m/texel)

- ◆ Height 2b (u16)
- ◆ Ambient Occlusion 0.5b (bc4)
- ◆ Biome 1b (u8)
- ◆ Texture blends 5b (rgba8 + u8)
- ◆ Grass 1.5b (u8 + bc4)
- ◆ Wetness 1b (2x bc4)
- ◆ Water flow/depth 2-3b (u8 x 2-3)



Terrain tile rsu 0 -1 -25

Pos -10822 -497558 22003 Pan 96 Hilt 12

^ N

<No Active Mission>

Copy terrain to CPU memory for physics etc.

- ◆ Height 4b - physics
- ◆ Grass 1b - AI
- ◆ Material etc. from virtual texture - sound, vfx

Careful of precision at 8km

- ◆ Lots of little problems
- ◆ $\text{pos} * \text{Inv}(\text{mat}) \rightarrow (\text{pos} - \text{posMat}) * \text{Inverse}(\text{m})$

Only rebuilds if inputs change

Flatten components into big arrays or BVH

Low memory footprint instancing

Far LODs stay up to date

```
int CompareHierarchy(Tree * tree0, Tree * tree1)
{
    CDynAry<Tree *> treeChain0, treeChain1;

    // Build parent chain for tree0 & tree1 ...

    int minCount = min(treeChain0.count(), treeChain1.count());

    for (int i = 0; i < minCount; ++i) {
        int n = CompareConstructionOrder(treeChain0.Tail(i), treeChain1.Tail(i));
        if (n) return d;
    }

    return arypPrim0.C() - arypPrim1.C(); // Favor shortest parent chain
}

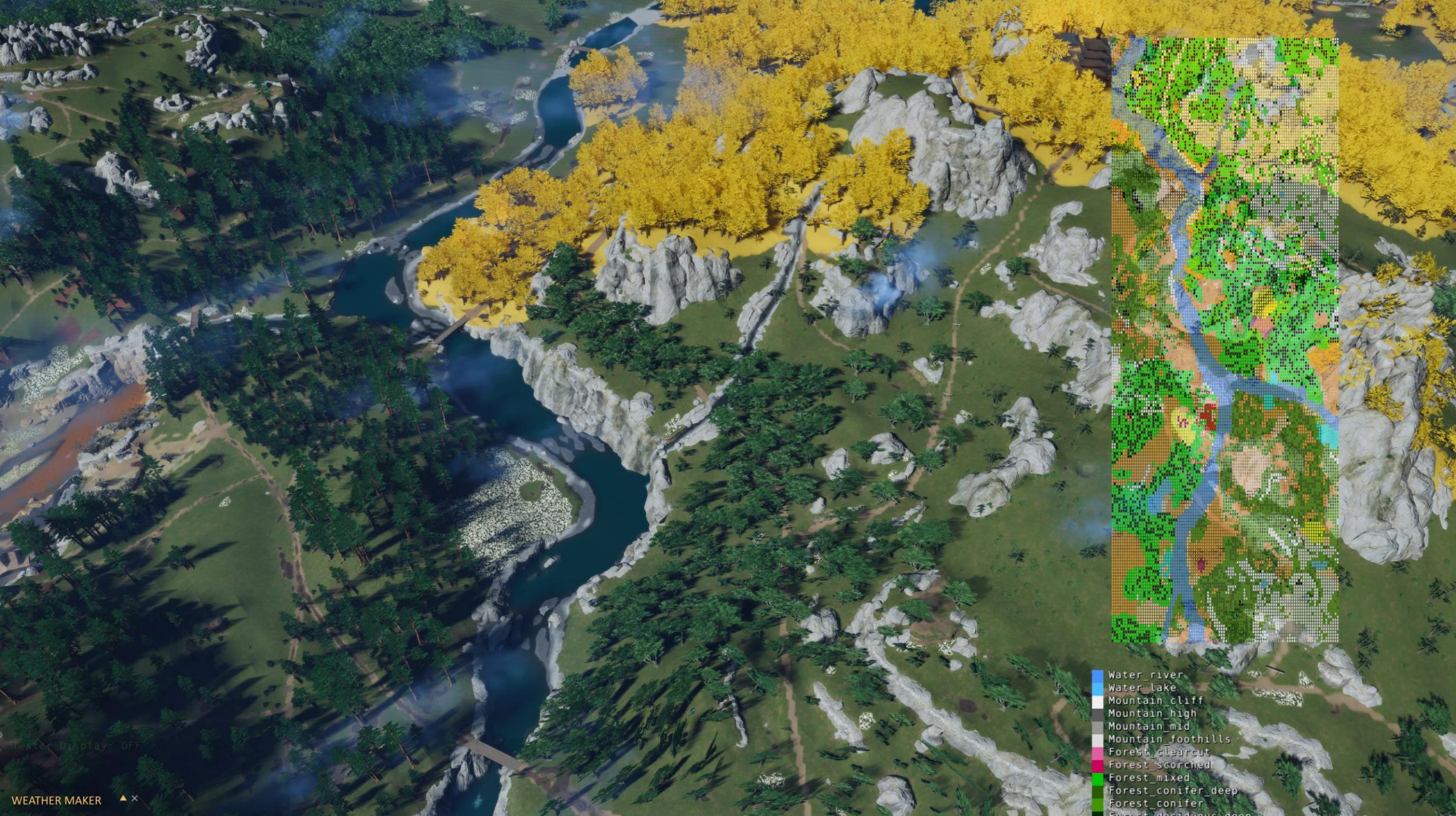
// Any elements sorted this way can use index ranges to refer to a group
```

```
void Merge(Tree * parent, const char * path, Filter filter) {
    if (IsCached(path)) {
        DuplicateTree(path, parent); // Reuse read & filtered tree
        return;
    }

    Tree * tree = TreeReadFile(path); // Read file

    // Recurse on nested file references
    for (auto * childTree = ChildrenToRecurse(parent)) {
        Merge(childTree, PathToMerge(child), filter);
    }

    FilterTree(tree, filter); // Reduce tree complexity
    CacheFile(path, tree); // Memoize the file contents
}
```



- Water_river
- Water_lake
- Mountain_cliff
- Mountain_high
- Mountain_mid
- Mountain_foothills
- Forest_clearcut
- Forest_scorched
- Forest_mixed
- Forest_conifer_deep
- Forest_conifer
- Forest_deciduous_deep

Tests: On play: OFF

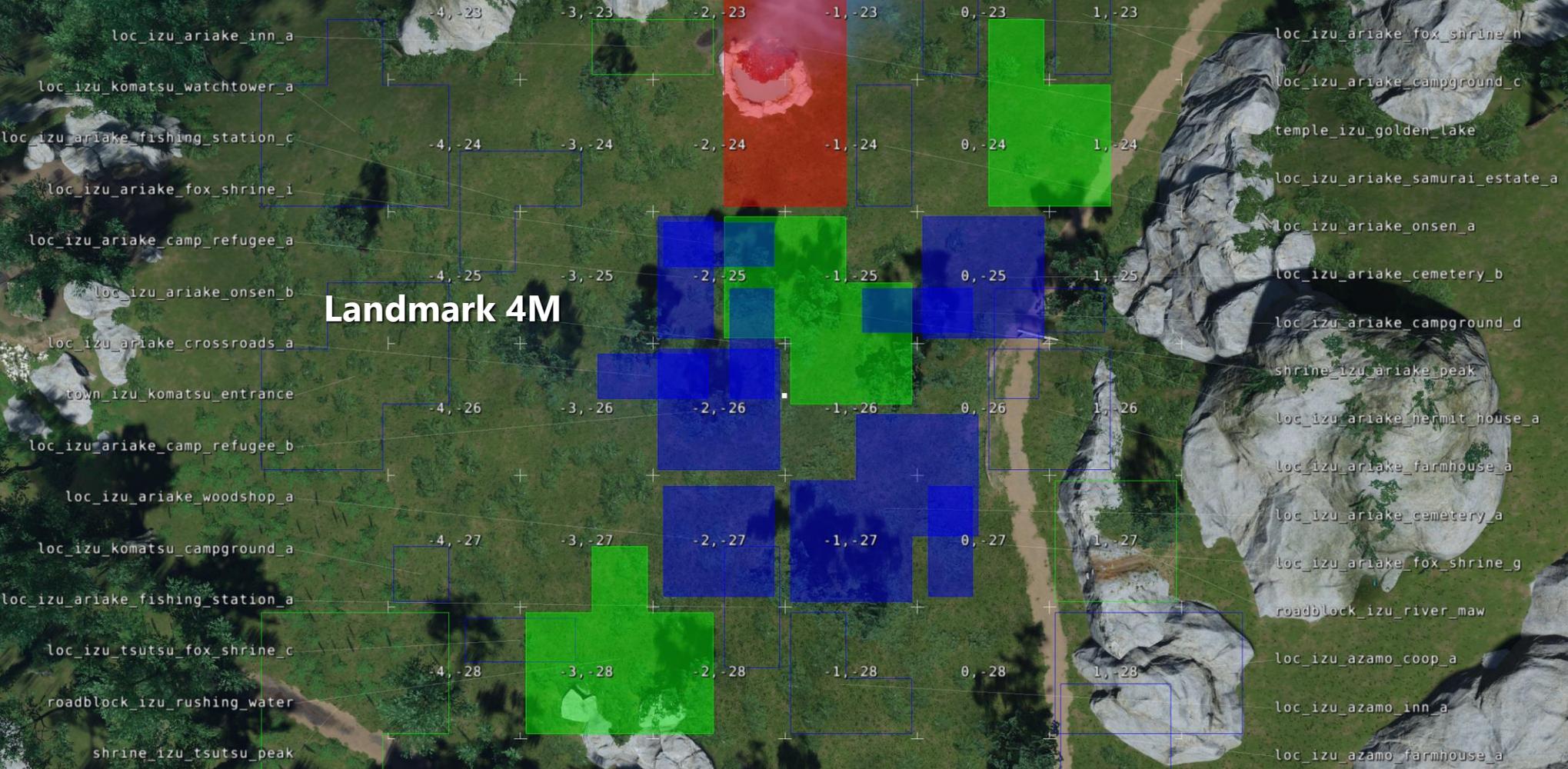
WEATHER MAKER ▲ ×

Header + Table of Contents (cached)
Virtual memory mapped pages
A few reads plus pointer patching

City 44M

Village 24M

Landmark 4M



City
Village
Landmark

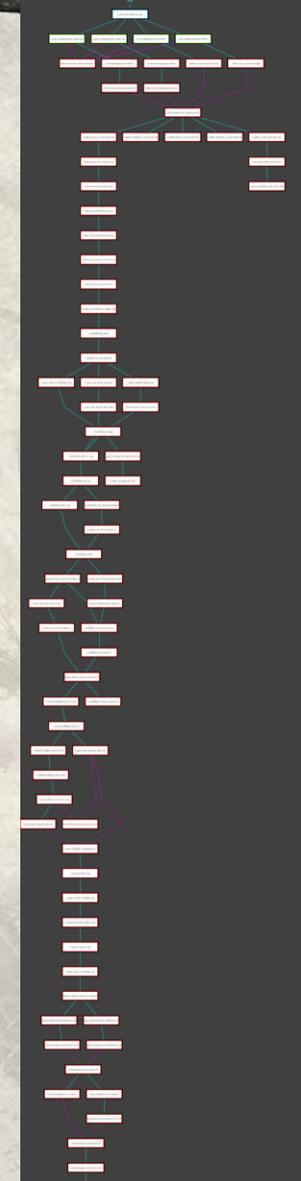
Many loaded at a time – each 0.25-8M

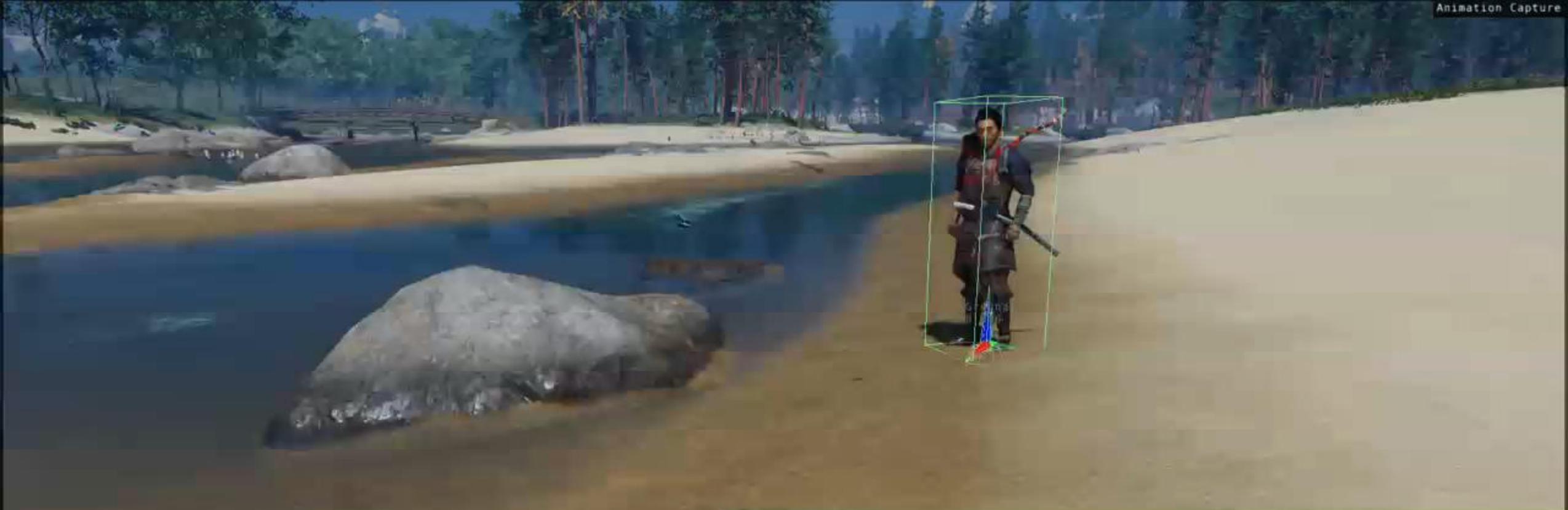
Task Graph with Components

- ◆ Deterministic testing
- ◆ More designer agency vs complex script

In-game editing/debugging

- ◆ Hints to procedural systems
- ◆ Remotable Dev UI
- ◆ Previs NPC animations
- ◆ History debugging





HERO - CAPTURE

Edit Options

Export Show: **hero (0x101b4ef8a0)** Lock actors Set Start Play Set End Prev Next Game: Resume

BlindFace		anim				
HoldProp1				anim_combat_fold_sword		
FullBody	anim_hero_idle_v2.idle	anim	anim_1	anim_hero	anim_hero_combat_ki	anim_hero_combat
JumpOut				anim_hero		
LandIn						
Additive	anim_hero_idle_additive.idle_1		anim_1		anim_hero_combat_ki	
BalanceAdditive			anim_1			
AimPartial			anim_1			
TorsoPartial	anim_hero_torsolook					
HeadPartial	anim_hero_headlook					anim_hero_headlook
Expression	anim_hero_expressionneutral		anim_1	anim_hero_mis	anim	anim_hero_expressionneutral
InPlay						



Changes can conflict!

- ◆ Snap assets to terrain during compile
- ◆ Cross game references compile time checks
- ◆ Sanity checking linear sequences

Followed Mission: m_rb_izu_stone_arch
Priority: Passive
PreferRespawns: no
Eligible Respawns:
57m m_rb_izu_stone_arch:t_rb_kill:RESPAWN
89m m_rb_izu_stone_arch:t_rb_kill:respawn_01
187m m_rb_izu_stone_arch:t_rb_kill:respawn_02
510m m_am_mengu::RESPAWN_postLiberate
752m m_am_jirandai::RESPAWN_postLiberate
* = preferred respawn point

No Companion(s)

Closest Fast Travel Spawn:
196m herodest_izu_azamo_onsen_b

Breadcrumb
Standing

Breadcrumb
Standing

No Task Info

Breadcrumb
Standing

Breadcrumb
Standing

Breadcrumb
Standing



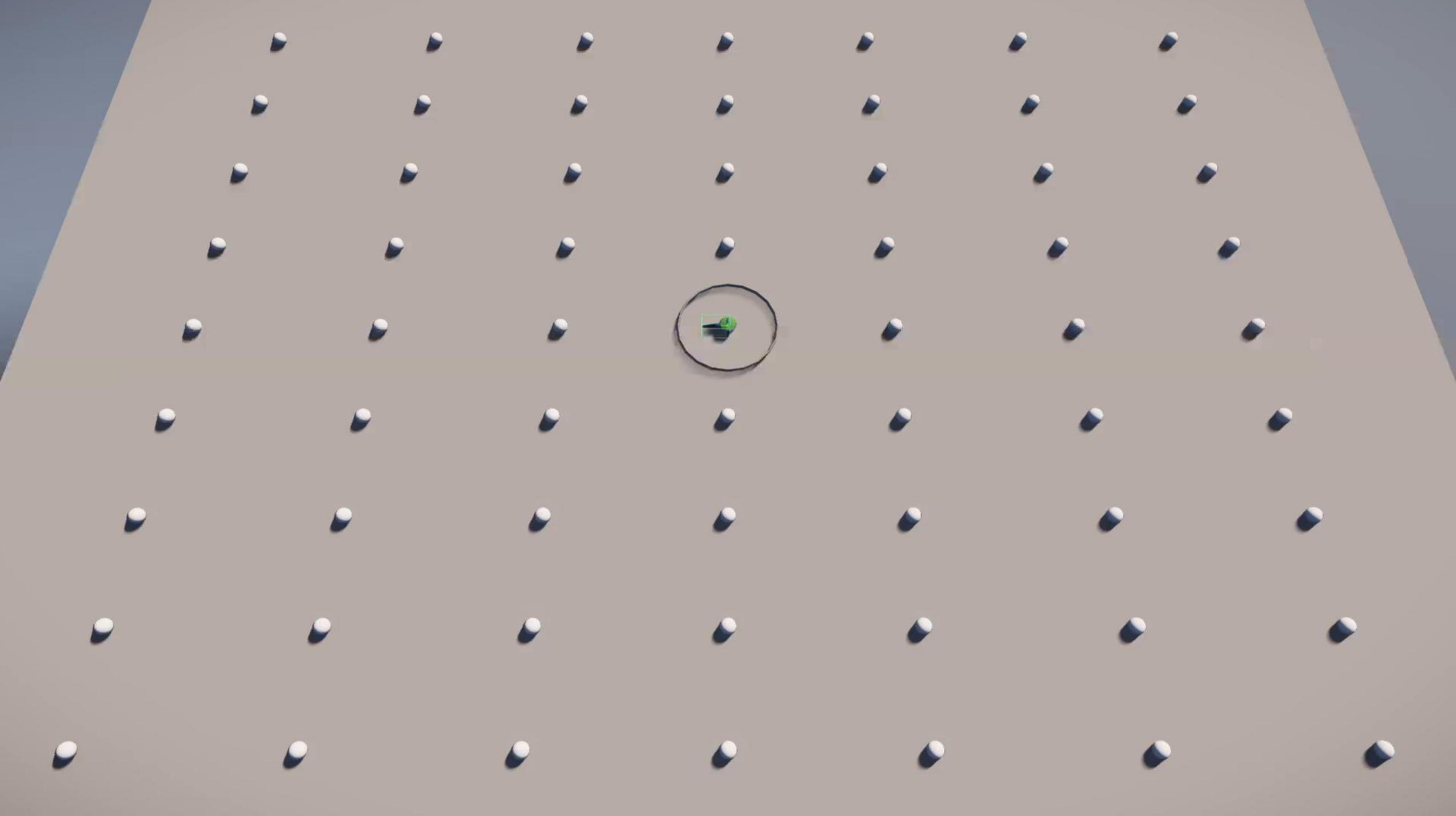
“The fastest code is code that doesn't run”

Doesn't exist > Asleep > Optimized > Naive

Doesn't exist - Spatial Streaming

Asleep - Static Distance Heap

Optimized - Visibility, Texture streaming



```
void CDistanceHeap::Update(Point posView) {
    float travelView = Dist(posView, m_posView);
    m_posView = posView;

    m_travelDistance += travelView;

    // When travel distance * 1.0001f when it goes over a limit ...

    // Remeasure and categorize any items that may be close enough
    for (;;) {
        if (m_heap[0].m_score >= m_travelDistance)
            break;

        float distance = DistanceToItem(m_heap[0].m_item, posView);

        TryUpdateActiveness(m_heap[0].m_item, distance <= 0);

        m_heap[0].m_score = m_travelDistance + Abs(distance);
        ReheapifyElement(0);
    }
}
```

Problem cases

- ◆ Tons of moving objects
- ◆ Camera cuts/teleports

Used in particle emitters, volumes...

animated objects, sound emitters, interactive objects...

TO BAKE OR NOT TO BAKE

GH  ST



Compile time > Runtime

- ◆ Unless dependencies are hard!
- ◆ Or it's huge... e.g. per TOD per tile baked data.

Expensive but smallish data good

- ◆ Procedural blends, placement, etc.
- ◆ Portals for pathing and AI cover spots
- ◆ Baked nightly



WHAT ABOUT BIG STUFF?

Content	Tile	Optimized
Terrain	3M	2.5M
Terrain Physics	24M	0M
Vegetation	12M	1M
Lighting	0.5M	0.11M
Pathing	6M	3M

ISS – 100% authored navigation mesh

- ◆ Too much work, Offline bakes are hard

Ghost – Live 20cm subgrids (4 512x512 per tile)

- ◆ Subgrids wake if all overlapping terrain/regions available
- ◆ Cast thousands of vertical rays against terrain & physics
- ◆ Buildings generate grids from custom meshes
- ◆ Slope rejection, disjoint set for connectivity
- ◆ Hierarchical A*
- ◆ Amortized over time.



tsu_0_2_22 (2632,6)
tsu_0_2_22 (3003,6)



Ladder

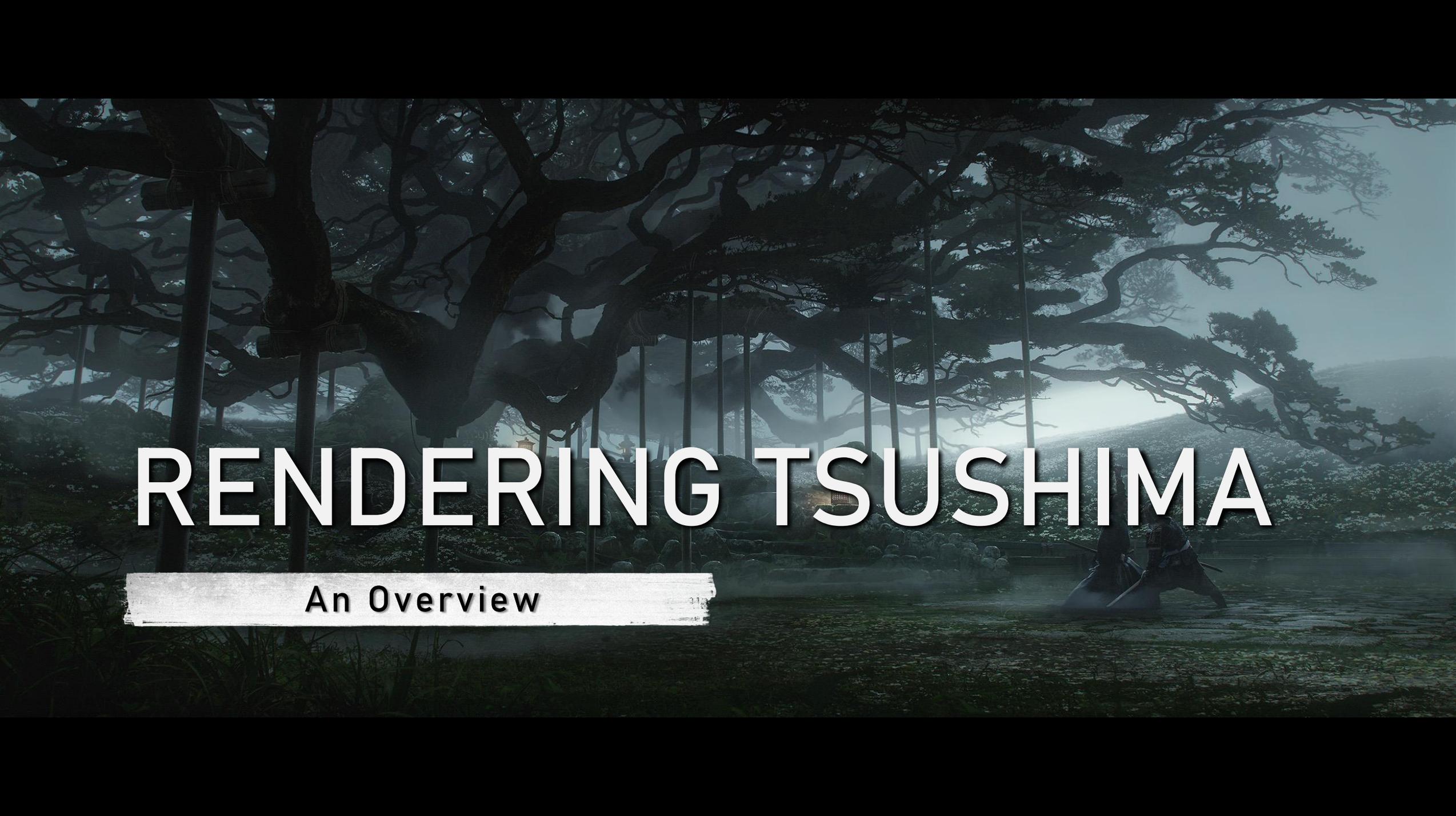
mult

building_inn_a dl.xml|pm_106 (2642,1)
tsu_0_17 -28 (2767,1)

Networking in a streaming world is hard

The background of the title screen is a dark, atmospheric scene. At the top center, a close-up of a woman's face with white makeup and dark hair is visible. Below her, a large, glowing red vertical beam of light descends from the sky, illuminating a group of samurai warriors in the distance. The ground is covered in red petals or leaves, and the overall color palette is dominated by dark reds and blacks.

GH[○]OST OF TSUSHIMA
LEGGENDS



RENDERING TSUSHIMA

An Overview

Everything is a mesh

Deferred (fat) mostly one pass

- ◆ Forward+ for skin etc.

Dynamic view BSP for CPU occlusion

Static TOD w/ indirect tetrahedral mesh

- ◆ Sampled on vertices or per mesh



Deferred with Z-equals for cutout

- ◆ Saved 3+ ms (see Doom 2016)
- ◆ Tile categorization lighting/decals

Height map terrain rendering

GPU occlusion

Real time TOD

Height map + normals +more

- ◆ 513x513 grid for simpler sampling

Rendered with 9-index buffers

- ◆ 64x64 quad grid
- ◆ Quad-tree with neighbor fixup
- ◆ See "Terrain Rendering in Frostbite"

Blends & Virtual Texture

- ◆ Expensive blending + many decals
- ◆ See MattP talk for more details

Compute + indirect draw

- ◆ Single 24b-32b per instance (all LODs/shaders)
- ◆ 10k instances ~0.3M (vs 12M)
- ◆ ~10M for all far LOD in world

```
struct SProxySetInstance
{
    FLOAT3 m_pos; // Object-space position of instance
    UINT1 m_eulPack; // Packed Euler Rotation (10-11-11)
    UINT1 m_vecScalePack; // Packed Scale (10-10-10)
    UINT1 m_bRand : 8; // Random byte
    UINT1 m_iTileid : 7; // Index of tileid of this instance (or 128 if none)
    UINT1 m_nProbeMask : 2; // Probe mask override for instance
    UINT1 m_iGroup : 15; // Which group we're in
};
```

Occlusion with last frame's depth (CPU & GPU)

- ◆ Per-triangle culling terrain culling

Far LODs drop with stochastic max distance

More details in Samurai Landscape talk...

Shader Swaps

RONIN



WANDERER



GHOST



Grass maps per tile for type/height

- ◆ Runtime placed blades and instanced meshes
- ◆ Leverages GPU compute rendering

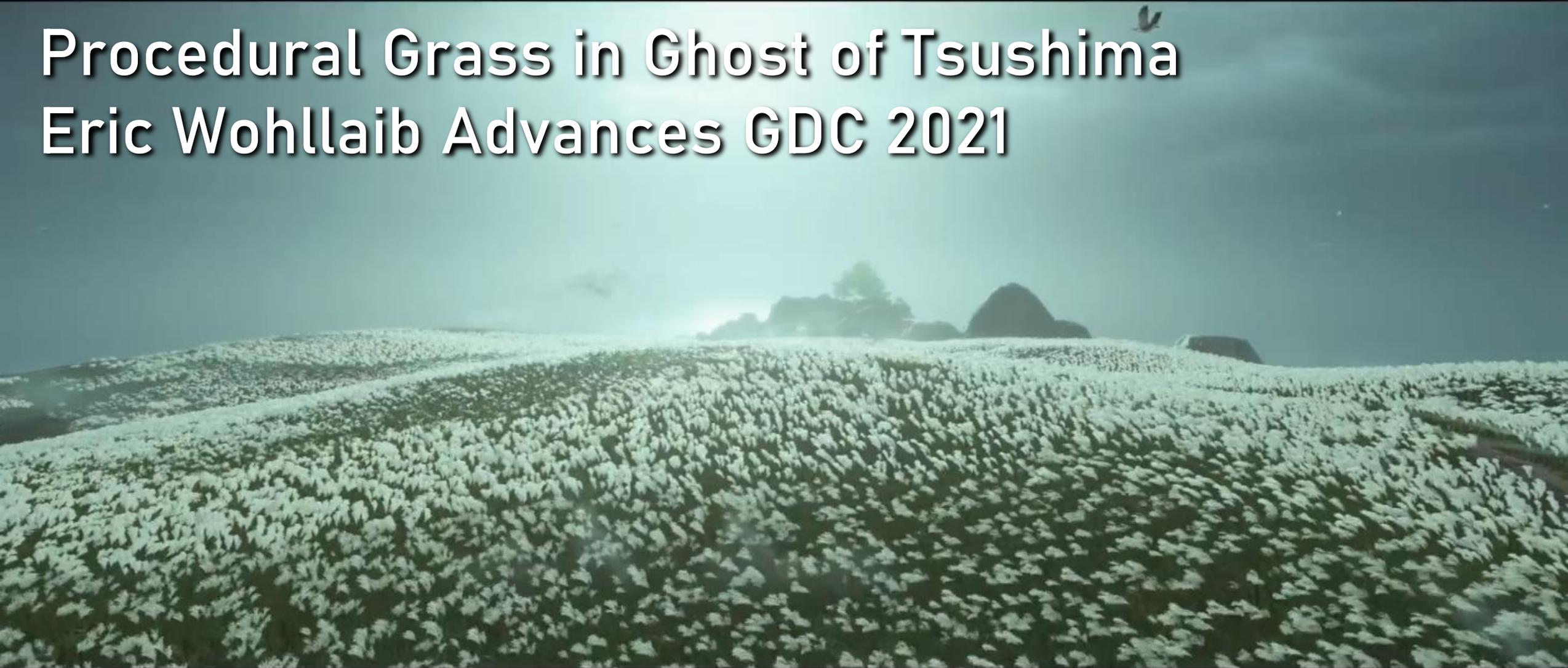
Stochastic grass shadowing

Fallback LOD texture blend on terrain

Wind and character interaction

Procedural Grass in Ghost of Tsushima

Eric Wohllaib Advances GDC 2021



No super-powers... but lots of leaves

Wind as a core gameplay feature

Particles

- ◆ Terrain, water access and collision.
- ◆ Animated meshes, reality bubble, and more

Tons of GPU Cloth

- ◆ Every character, flag, doorway
- ◆ Collisions, constraints, layering, etc.

Blowing from the West: Simulation and Particles in Ghost of Tsushima

Bill Rockenbeck GDC 2021





GPU relighting

- ◆ Nearest 16 cubemaps self-shadowed
- ◆ Terrain probes 16x16x3 per tile
- ◆ Streamed tetmesh data for buildings

Disc ~44m whole world + 2M per city or so

Runtime BC6 compression

- ◆ <https://github.com/knarkowicz/GPURealTimeBC6H>





Thin walls - interior visibility mask

Real-time Clouds

Volumetric fog

Scattering aligned color space

Lots of async compute

Lighting & Rendering Ghost of Tsushima

JasminP SIGGRAPH 2021



Screen-space Shadows

- ◆ Drop small shadowed items for perf
- ◆ Cards for tree shadows

Profile Guided Optimization

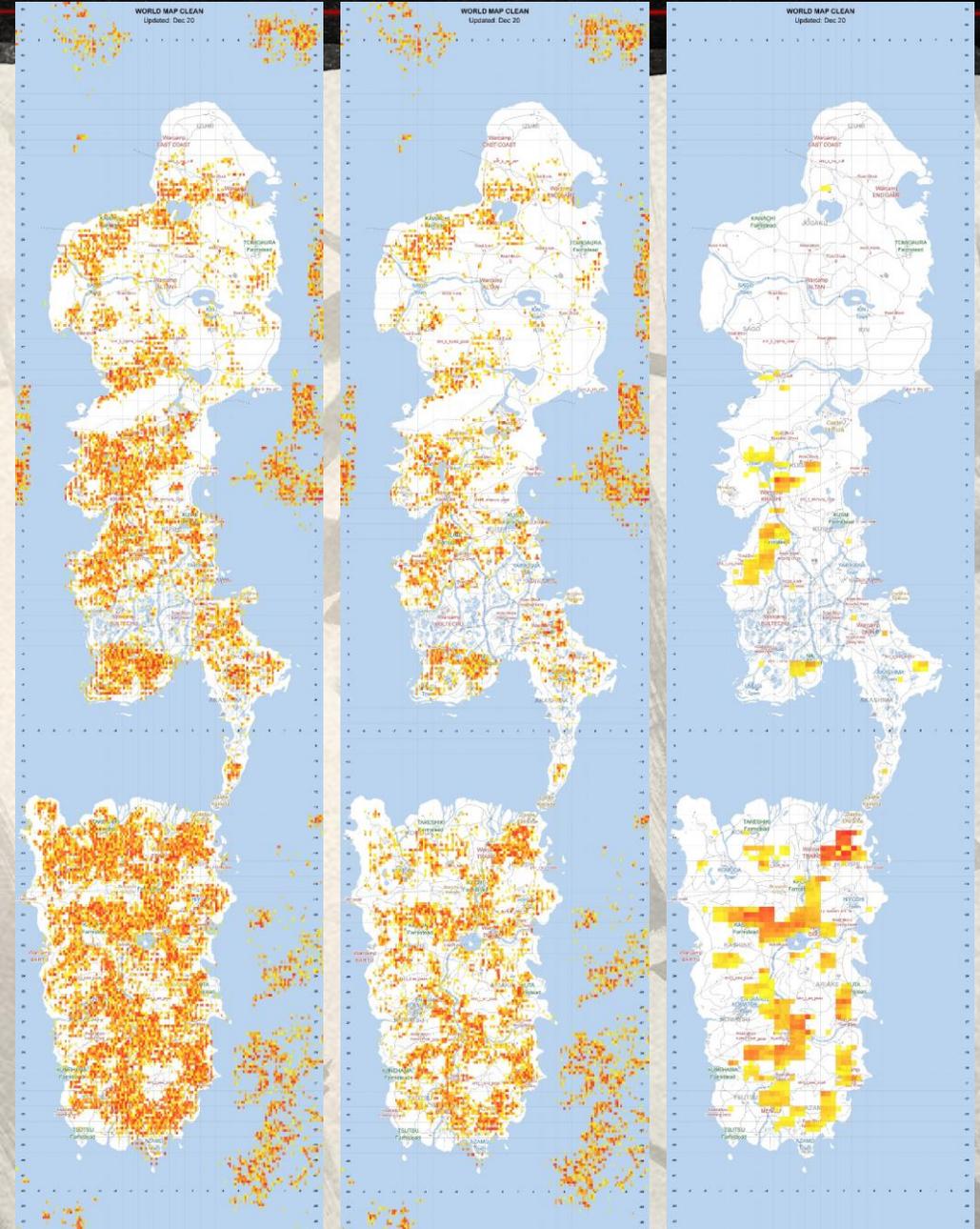
Lots of Scalarized loops

- ◆ Drop texture atlas for particles
- ◆ Terrain texture sampling
- ◆ Shared blend values



2018 – tests way over
2020 – nearing budget
Ship – solid

Solid team effort





GH  ST OF TSUSHIMA



TEXTURE STREAMING

A Tale of Development

“It’ll be ok to put everything in Core.”

“This is just a forward-looking experiment.”

Narrator – “It wasn’t...”

E3 2018 blocked by memory issues

Mildly panicked switch

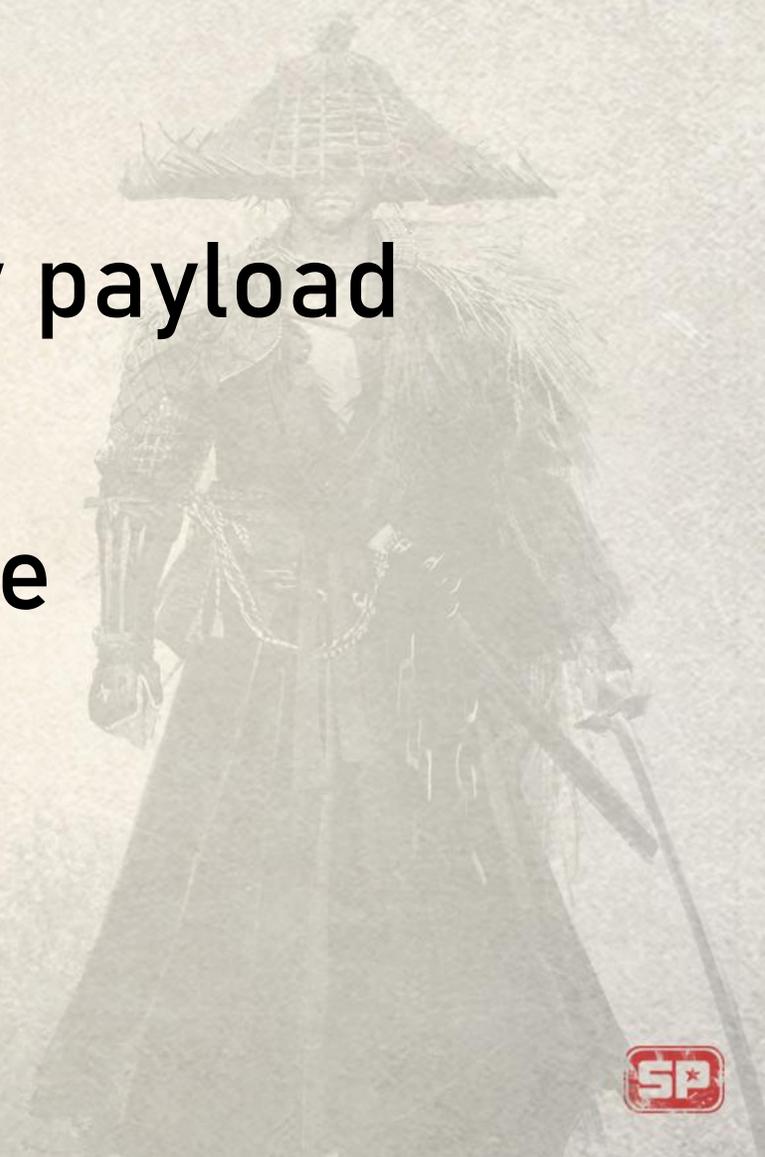
Defragmentation

Textures are metadata plus binary payload

- ◆ Single read for any mip range

Single manifest loaded at boot time

- ◆ All metadata and mips smaller than 64



Texture Budget Ok
Mesh Budget Ok

Budget	Texture		Mesh	
	Want	Have	Want	Have
None	128.2M	128.2M	0	2.62M
Environment	431.7M	431.7M	20.06M	53.06M
Terrain	122.3M	140.55M	0	0
Character	1.3M	1.3M	5.5M	5.5M
Forced				
PolycutPre	0	0	0	0
Polycut	0	0	0	0
ForceMisc	5.47M	5.47M	0	0
Ui	247.47M	247.47M	0	0
Fx	82.68M	82.68M	0	0
SpecProbe	9.67M	9.67M	0	0
Engine	3.73M	3.73M	0	0
UiHighPri	436K	436K	0	0
Total	1.06	1.02G	25.56M	61.18M



Single manifest loaded at boot time

- ◆ All metadata and mips smaller than 64

```
struct STexMeshManifest
{
    struct STextureLowMips
    {
        CString m_file; // File name
        SMd5 m_md5Sources; // Md5 sources for this texture (for diffing)
        STextureData m_textd; // Dimension, format, etc.
        u64 m_offsetMips; // Index of data for first mip
        u64 m_offsetLowMips; // Index of data for non-streamed mips
    };

    CDynAry<STextureLowMips> m_textures; // Textures
    CDynAry<u8> m_lowmipBytes; // Bytes for all low mips

    CDynAry<SMeshGroupInfo> m_meshes; // Meshes
};
```

Most objects don't move

- ◆ Non-dependent store doesn't block

Push all data to array when it changes

- ◆ Prep for threading now nearly memcpy
- ◆ Branch for rare compact SIMD data

Profile and be careful of overhead!

- ◆ Copy several MB from many threads a bad idea
- ◆ Atomic increment initially dwarfed copy cost


```
inline int4 NCeilLog(float4 vec) {
    int4 vecIeee = int4(vec); // bit-wise cast to int
    vecIeee += int4(int(((1 << CBIT_FltFrac) - 1) - (EXP_FltBias << CBIT_FltFrac)));
    return vecIeee >> CBIT_FltFrac;
}

void ProcessShaderGroup(...) {
    float4 dist = SqrtFast(DistSqr(localBounds, localViewPosition)); // SqrtFast = x * _mm_rsqrt_ps(x);
    float4 distTex = max(shdgroup.m_distMin, float4(penaltyRatio) * max(dist - float4(biasView), float4(minView)));
    float4 perspectiveRatio = float4(ratioFov) * _mm_rcp_ps(distTex);

    // Also measure distance to prefetch positions if prefetching...

    // Calculate pixels covered by shader's uv density
    // (uv_dist / cam_dist) * (half_screen_res / tan(fov/2))

    float4 textureAreaRatio = perspectiveRatio * perspectiveRatio;
    float4 screenCoverage = shdgroup.m_meshAreas * textureAreaRatio;

    // Approximate pixel count and conservative log for mip (^8 for fixed point fraction)

    float4 pixelFixedPoint = float4(halfScreenRes) * shdgroup.m_uvSize * perspectiveRatio;
    pixelFixedPoint *= pixelFixedPoint;
    pixelFixedPoint *= pixelFixedPoint;
    pixelFixedPoint *= pixelFixedPoint;

    int4 logPixel = min(int4(logClamp), NCeilLog(pixelFixedPoint));
    for (int i = 0; i < 4; ++i) {
        SShaderStat * stat = &aryShdstat[shdgroup.m_shdids[i]];
        stat->m_score += screenCoverage[i];
        stat->m_logPixels = max(stat->m_logPixels, s16(pixelFixedPoint[i]));
    }
}
```

Particle atlas – drop it, scalarize and use
bindless

Virtual texture caching update

- ◆ Always update 1/36th of 1 slice per frame

UVs are complex (degenerate, weird, etc.)

Tried various UV Density heuristics

- ◆ Average of 90% of triangle area
- ◆ Maximum of 90% of triangle area
- ◆ Min max dimensions with aniso (90% of tri area)
- ◆ Max of log bucketed area (80% area and clamp)

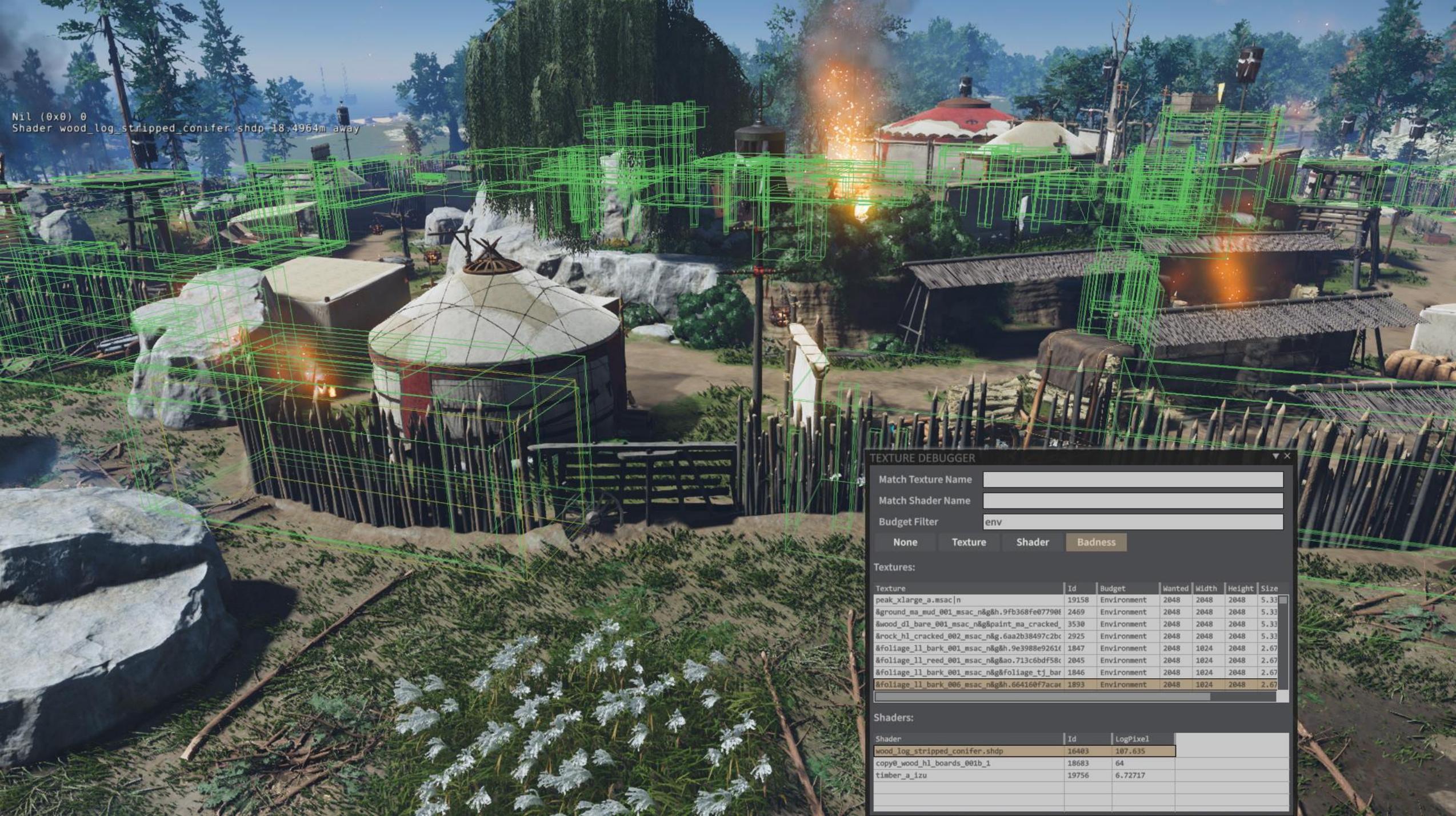
Approximate for procedural UVs

Shader UV scale is a pain

1G texture budget, squish when over

```
void UpdateOverbudget() {
    bool overbudget = m_sizeWanted > sizeTotal - sizeReserved;
    if (overbudget) {
        // increase min distance and reduce bias, then scale distance
        m_penalty = min(64, m_penalty + 1); // clamp at high multiplier
        m_cooldown = 1;
    }
    else if (m_nPenalty > 0) {
        // If no reads requested or not recently overbudget lower penalty
        if (!any_reads_desired)
            m_cooldown = 0;
        if (m_cooldown == 0)
            m_penalty -= 1;
    }
}
```

Nil (0x0) 0
Shader wood_log_stripped_conifer.shdp 18.4964m away



TEXTURE DEBUGGER

Match Texture Name

Match Shader Name

Budget Filter

None Texture Shader **Badness**

Textures:

Texture	Id	Budget	Wanted	Width	Height	Size
peak_xlarge_a.msac n	19158	Environment	2048	2048	2048	5.33
&ground_ma_mud_001_msac_n&g&h.9fb368fe07790e	2469	Environment	2048	2048	2048	5.33
&wood_d1_bare_001_msac_n&g&paint_ma_cracked_	3530	Environment	2048	2048	2048	5.33
&rock_h1_cracked_002_msac_n&g.6aa2b38497c2bc	2925	Environment	2048	2048	2048	5.33
&foliage_l1_bark_001_msac_n&g&h.9e3988e9261f	1847	Environment	2048	1024	2048	2.67
&foliage_l1_reed_001_msac_n&g&ao.713c6bdf58c	2045	Environment	2048	1024	2048	2.67
&foliage_l1_bark_001_msac_n&g&foliage_tj_bar	1846	Environment	2048	1024	2048	2.67
&foliage_l1_bark_006_msac_n&g&h.664160f7aca	1893	Environment	2048	1024	2048	2.67

Shaders:

Shader	Id	LogPixel
wood_log_stripped_conifer.shdp	16483	107.635
copy0_wood_h1_boards_001b_1	18683	64
t1mber_a_izu	19756	6.72717

Necessary and Hard

- ◆ Character existence
- ◆ Nearest next cutscene
- ◆ Every camera cut (e.g. within/across cutscenes)

Characters locked by default at 1k

- ◆ Cutscenes and photo mode can unlock them

Growth: Grid then bottom-up merge

- ◆ Group bounds at 64 grid (or 5m cells)
- ◆ Merge until goal reached (e.g. goal $\sqrt{\text{count} - 4} + 4$)

Terrain & grass – bounds around $\frac{1}{4}$ of tile

- ◆ Min camera distance clamp

Similar

- ◆ Also squishy!
- ◆ Manifest stores headers, single read for any LOD range
- ◆ Parallel measure code

Complex to suppress LOD drawing

Virtual memory with 64K pages

```
inline float DistanceFromNlod(NLOD nlod) {
    float fraction = (nlod / float(NLOD_Max));
    float distance = fraction * fraction * NLOD_MAX_DISTANCE; // Up to 12.8km
    return (nlod < NLOD_Max) ? distance : FLT_MAX;
}

struct SMeshGroupInfo {
    struct SMeshInfo {
        U64 m_hashStream; // Unique 'name' of mesh within the streaming group
        U32 m_offsetIndices; // Offset of the indices
        NLOD m_nlodMic; // LOD range this mesh supports
        NLOD m_nlodMac; // ...
        CFixAry<int, 16> m_offsets; // Offsets within the memory layout
    };

    struct SLodInfo {
        NLOD m_nlodMac; // LOD distance we support (any distance greater)
        int m_offset; // How much we need to have read for LOD
    };

    CString m_filename; // File we'll read mesh data from
    SMd5 m_md5Sources; // Md5 sources for this texture (for diffing)
    u32 m_offsetData; // Offset of mesh data
    U32 m_sizeData; // Size of data

    CDynAry<SMeshInfo> m_meshinfos;
    CFixAry<SLodInfo, 8> m_lodinfos;
};
```

A samurai in traditional dark armor with red accents stands with his back to the camera in a dimly lit, atmospheric room. He has a katana tucked into his belt. The room features a wooden floor, a table with lit candles, and various samurai-related items. Large, stylized Japanese characters are visible on the wall behind him. A large white text overlay is centered on the screen.

CONCLUSION

Loading...

Cold Boot ~44s

Fast Travel ~8s

Death < 5s

- ◆ Delayed load to show user tips screen

VISION AND CONSTRAINTS



Don't load so many textures!

- ◆ Reduce textures by 2.2x when warping ($\sim 1/4^{\text{th}}$ the memory)
- ◆ Unless into a cutscene...

Minimize reads per file

- ◆ Aggregate small assets

Minimize size

- ◆ Merge at compile time
- ◆ Optimize, then optimize some more

Lean on GPU compute

Pull a few tricks

More virtual memory & GPU compute

More fine streaming (anims, sounds)

- ◆ Need to make problems squishy

Improved tools, build time, etc.

Solve complex UVs

THANKS

GH[↑]ST



WE ARE HIRING!

Want to work on problems like these?

We're looking for programmers!

Tools, Graphics, Gameplay, Camera, Audio, Engine

Junior or Seniors welcome!

<https://jobs.suckerpunch.com/>

GH  ST OF TSUSHIMA



EXTRAS

For your enjoyment



```
t_remind_stance_stone_ronin: Running? #f
t_train_stance_stone_mongol: Nil
t_train_stance_water: Nil
t_train_stance_wind: Nil
t_train_stance_moon: Nil
```



```
[x1] engine :: update pre-solve :: Unable to find animation Nil in context horse
[x2] Cloth sim does not support non-uniform scale on object banner_bamboo_game_a_ja.xml|CLOTH [billro]
[x92] Animation tu_hero_duel_lite_intro_samurai.xml|anim_samurai|play cannot find IK handle /l_hand_ikHandle/ on horse.xml|c_root (m_dl_umugi_sprig|roster_E01)
[x21] Attacker with archk=Horse does not have specified intensity value!
```

Similar to Tribes model

- UDP with sequence numbers
- Messages (unreliable) and reliable messages (unordered)
- Snapshots of a thing with dirty bits (unordered reliable)

Streaming lifetime vs ownership

- Host arbitrates whole pack ownership
- Generation numbers so people know what to accept
- Authority trading either enforced or vote-based

Users crashing in Jan/2021 with no recent patch

Best guess:

1. Some machine created a bogus empty snapshot.
2. Transmitted it to everyone.
3. Snapshot persisted across a warp (shouldn't happen).
4. Crash wouldn't happen until later on.
5. This allows it to continue to spread!

Ghost in the machine!

1. Force repro by corrupting a snapshot.
2. Don't keep across warps.
3. Force crash if send/receive bad snapshots.

Exciting times. 😊