



Delta Force: Techniques From Unified Production Pipeline To Cross-Platform Runtime Support

Shangli Liang

Deputy Director of Game Client Engineering, TiMi Studio Group, Tencent Games

Yang Shi

Game Client Engineering Team Lead of Delta Force, TiMi Studio Group, Tencent Games

**MAY CONTAIN CONTENT
INAPPROPRIATE FOR CHILDREN**

Visit esrb.org for
rating information





- **Multiple game modes**
- **Performance first**
- **Big world (10km x 10km)**
- **Realistic art style**
- **Scene richness**

Same team, same time!

Delta Force: Hawk Ops

Positioning: A Next-Gen Operator-Based Tactical Shooter on PC / Console / Mobile

"We are trying to bring the top FPS experience to players on all platforms."

Challenges (PC & Mobile)

- **Big performance and architecture gap between PC and Mobile**
- **We are covering low-end devices**
- **Multiple rendering pipeline**
- **Cross-platform play means operation pace keeps same**
- **One team**

Production Pipeline

★ Asset Production

Mesh

Character

Effects

Foliage

LOD Chain

Deco Rig

GPU Particle

Speed Tree

Collision

Shader

Lit Particle

Bake AO

Texture Res

Animation

Physics Sim

Wind Field

Mip-Gen

Appearance

...

Serialization

★ Virtual Material

★ Per-platform Properties

★ Levels

★ Editing

★ Build Level

Layered

CDLOD

Actor Mapping

Streaming Assets

Lighting

Physics

Biome

Mobile Batch

Tag

HLOD

Distribution

PVS

Cross-Platform Runtime Support

Rendering

Gameplay

★ Perf Tracking

Shading

★ 3C

CPU

★ Landscape

Input Layer

Threading

Lighting

★ Feature Planning

GPU

★ GI

Shadow

★ Quality Evaluation

Memory

AO

IBL

★ Feature Planning

Hitches

Volumetrics

Network

IO

Tessellation

Weapon & Bullets

VRS





★ UI

Layout

Adaptive

Compression

Asset Production - Shared Asset

				
Characters	Weapons	Scenes	Effects	Foliage
<div>Hero Character</div> <div>NPC</div> <div><div>Boss</div><div>Other</div></div> <div>Different Shader</div>	<div>Fully shared</div> <div>Different Shader</div>	<div>Most meshes are shared</div> <div>LOD chain management</div> <div>Different shader</div> <div>Level Editing</div> <div>Performance</div> <div>Workload</div>	<div>Most are exclusive</div> <div>More features</div>	<div>All Exclusive</div> <div>PCG levels</div>

What problems arise for sharing assets across platforms?

Asset Production – Shared Asset



PC



Mobile



- **Manage different shaders in same mesh**
- **Deal with different collisions**
- **Different texture res & format**
- **Redundant platform data**
- ...

Most meshes are shared

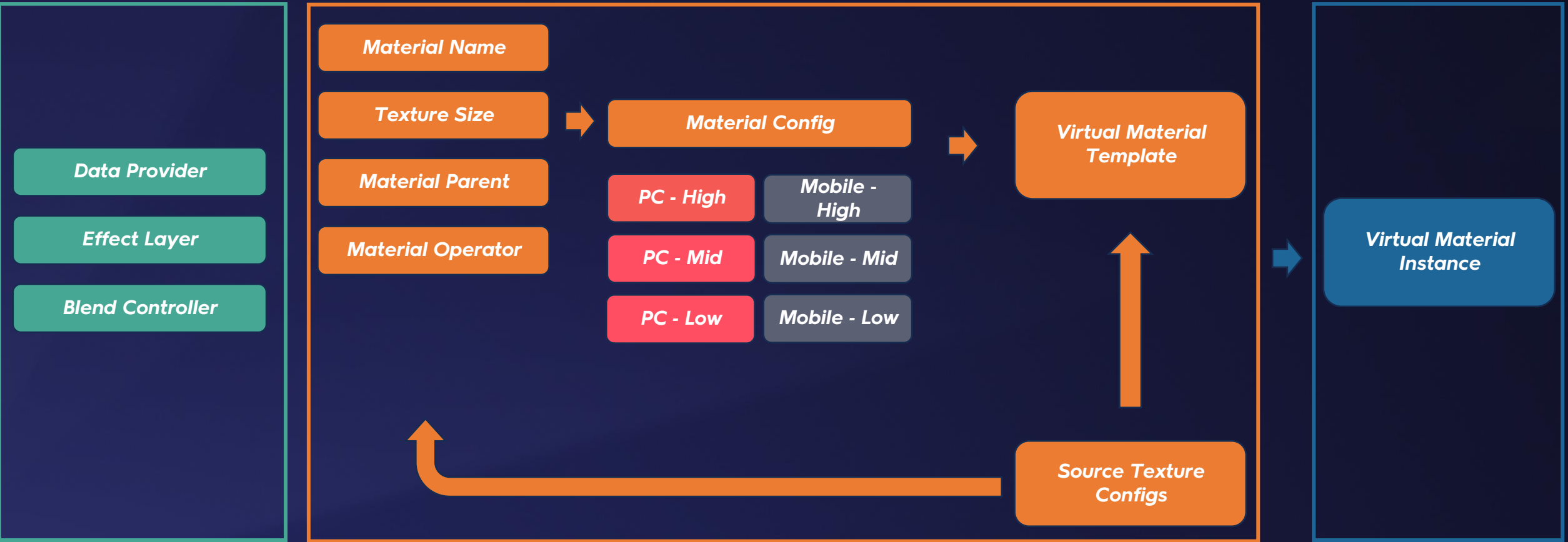
- **Decouple resources and shading**
- **Flexible Customization of resource organization rules**
- **Auto generate multi-platform and multi-quality resources**
- **Material modularization and function reuse**



Material Layer

Virtual Material Template

Virtual Material Instance



Asset Production – Collisions



- **Collision data are shared**
- **FPS game requires precise collision**



PC0



PC1



PC2



PC3



Mobile 0

PC4



Mobile 1

PC5



Mobile 2

Asset Production – Per platform Properties

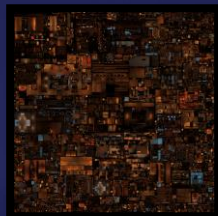
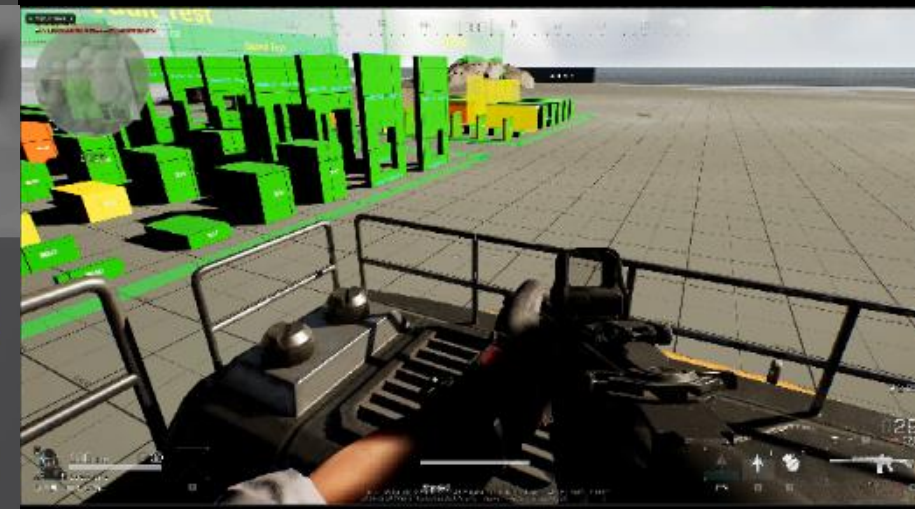
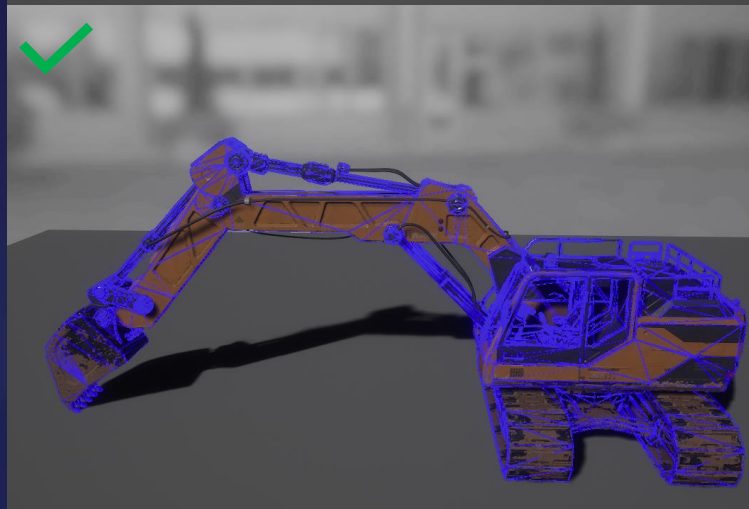


Mobile



- **Per-platform Collisions**
- **Serialization during cooking**
- **Other per-platform properties**

PC



Asset Production - Levels



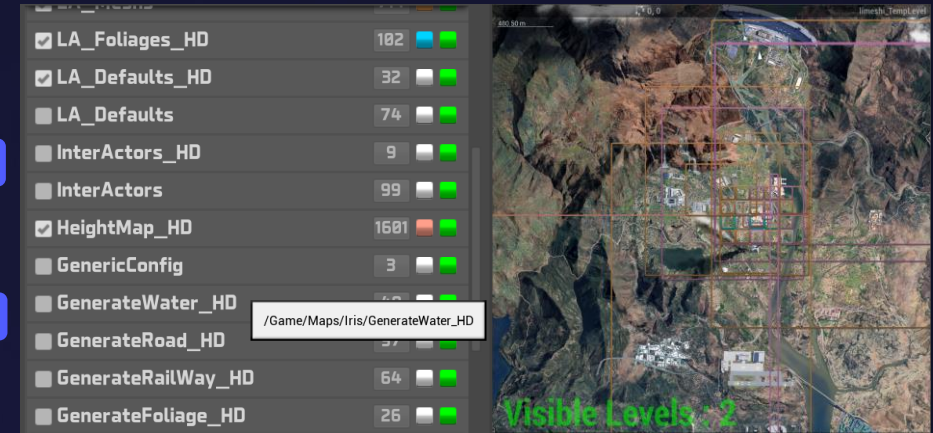
- *Create immersive scenes*
- *Different scene richness*
- *Manage different loading distance*
- *Identify possible performance breaker*
- *Perform specific optimizations for different platforms*



Asset Production – Editing Level



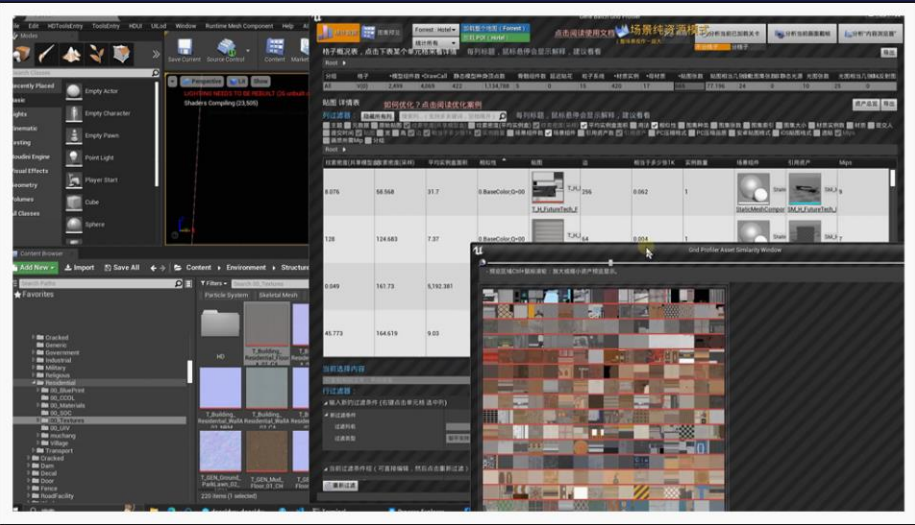
- **Pre-defined layers for shared, PC and mobile**
- **Levels are auto distributed to different groups according to bound size**
- **PCG Levels are platform exclusive**
- **Categorization management**
- **Actor mapping**



Asset Production – Grid Profiler



- **Level LOQ procedure**
- **Platform specific optimizations**
- **Multiple analyze mode (Texture usage, LODs, Platform correction, Culling screen size etc.**
- **Batching analyzation**



Texture Similarity

Mode Switch

Analyze

Over View Table

DrawCall(按SM划分)	SMC	模型	LOD数量	最小LOD	绘制Section数	实例数	体积	表面积	实例误差系数	SMC剔除屏幕大小	顶点数	当前绘制LOD	当前绘制屏幕大小	Lightmap缩放	实例化属性
0	SM_Sky	SM_SkySphere	1	0	1	1	35,184,372,088,832	6,442,450,944	0	0.015	2,208	0	5,778,458	1	CollisionProfile=BS
0	SM_Sky	SM_SkySphere	1	0	1	1	2,702	11,641	1	0.015	5,192	0	0.171	1	CollisionProfile=BS
0	SM_Sky	SM_Ball_01	1	0	1	1	0.125	1,498	1	0.015	816	0	0.104	1	CollisionProfile=BS
0	SM_Sky	SM_Cube_01	1	0	1	1	0.373	3,892	1	0.015	816	0	0.163	1	CollisionProfile=BS
0	SM_Sky	SM_Cube_02	1	0	1	1	0.125	1,498	1	0.015	816	0	0.134	1	CollisionProfile=BS
0	SM_Sky	SM_Cube_01	1	0	1	1	0.48	4,392	1	0.015	1,224	0	0.189	1	CollisionProfile=BS
0	SM_Sky	SM_Cube_03	1	0	1	1	0.48	4,392	1	0.015	1,224	0	0.205	1	CollisionProfile=BS

Detail Filter

Statistics

Reason & Advice

Asset Production – Runtime Level



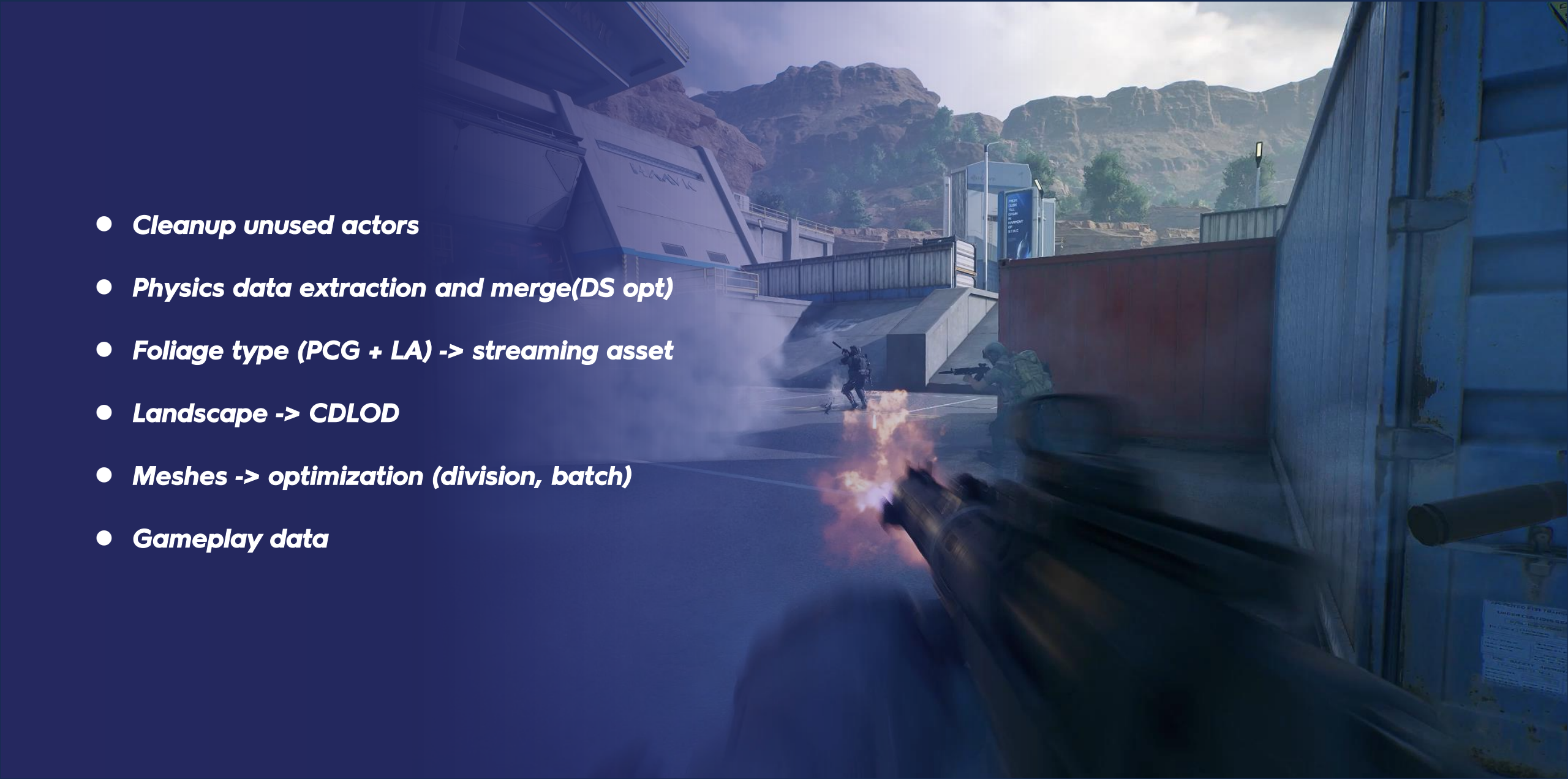
- **Performance**
- **View distance**
- **Memory – loading distance**
- **IO – streaming request**
- **Heating on mobile**

Platform	HLOD	Big	Mid	Small
PC	512m +	384 – 512m	256 – 384m	128 – 256m
Mobile	384m +	256 – 384m	128 – 192m	64 – 128m
Scoping	Visibility vs. Loading			



Same assets, same levels, different runtime with robust performance!

- *Cleanup unused actors*
- *Physics data extraction and merge(DS opt)*
- *Foliage type (PCG + LA) -> streaming asset*
- *Landscape -> CDLOD*
- *Meshes -> optimization (division, batch)*
- *Gameplay data*



Asset Production – Runtime Levels

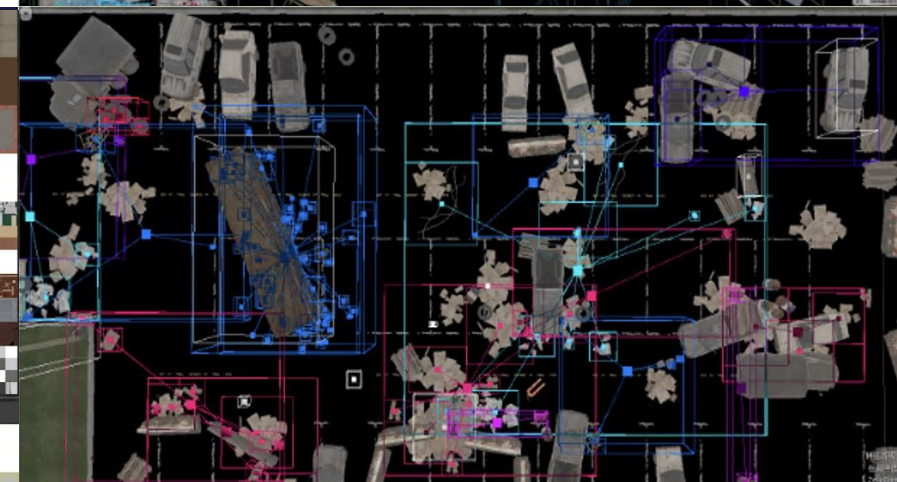
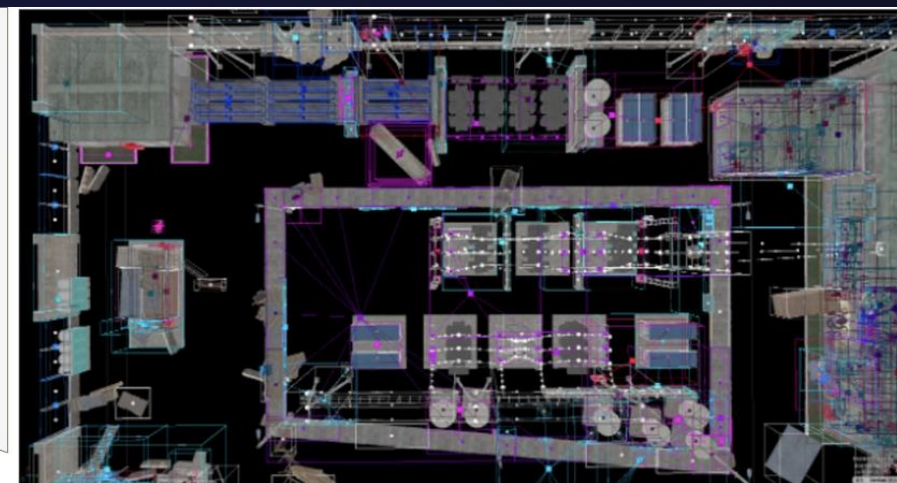
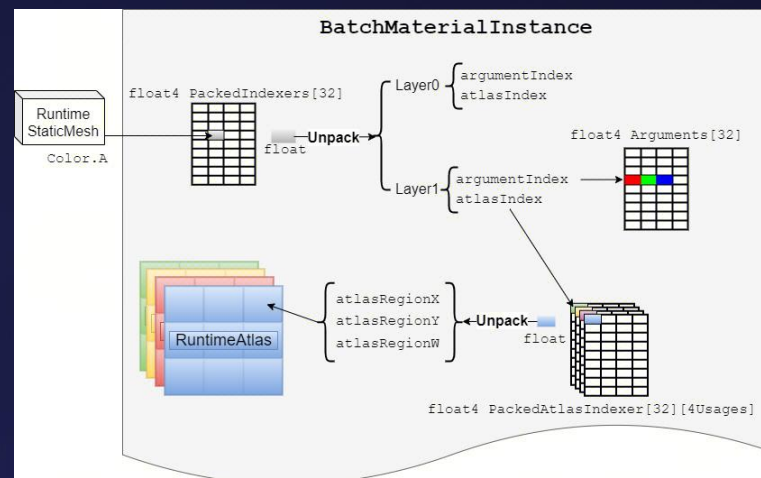


● Feasibility

- Material
- Material Params (1 float4)
- Texture Format
- Lightmap
- Shadow Cast
- IBL

● Rationality

- Vertex Count
- Texture Resolution
- Distance



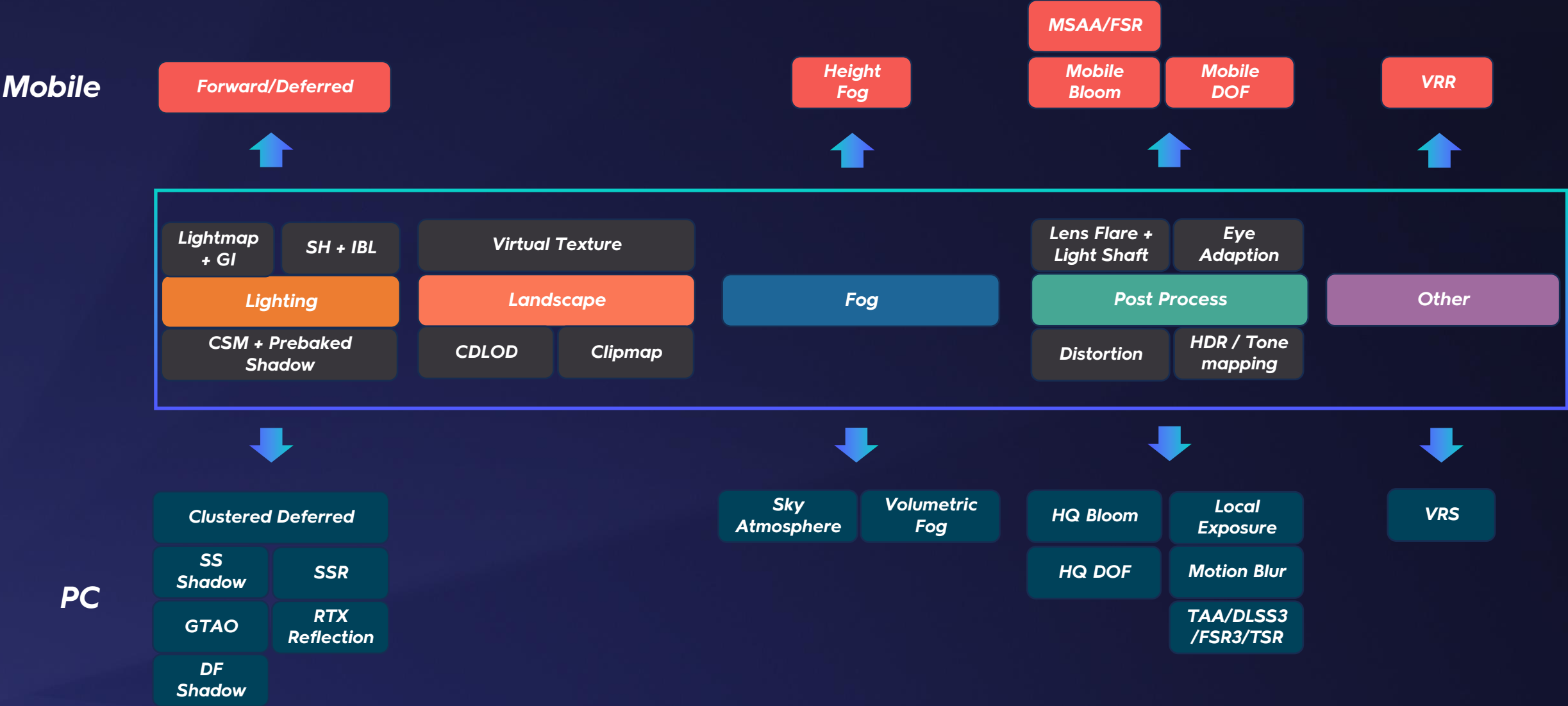
Draw call reduced significantly while disk space redundant is acceptable

- Rendering
- Gameplay
- Performance



- Rendering
- Gameplay
- Performance





Runtime - Lighting



Runtime – Volumetric GI



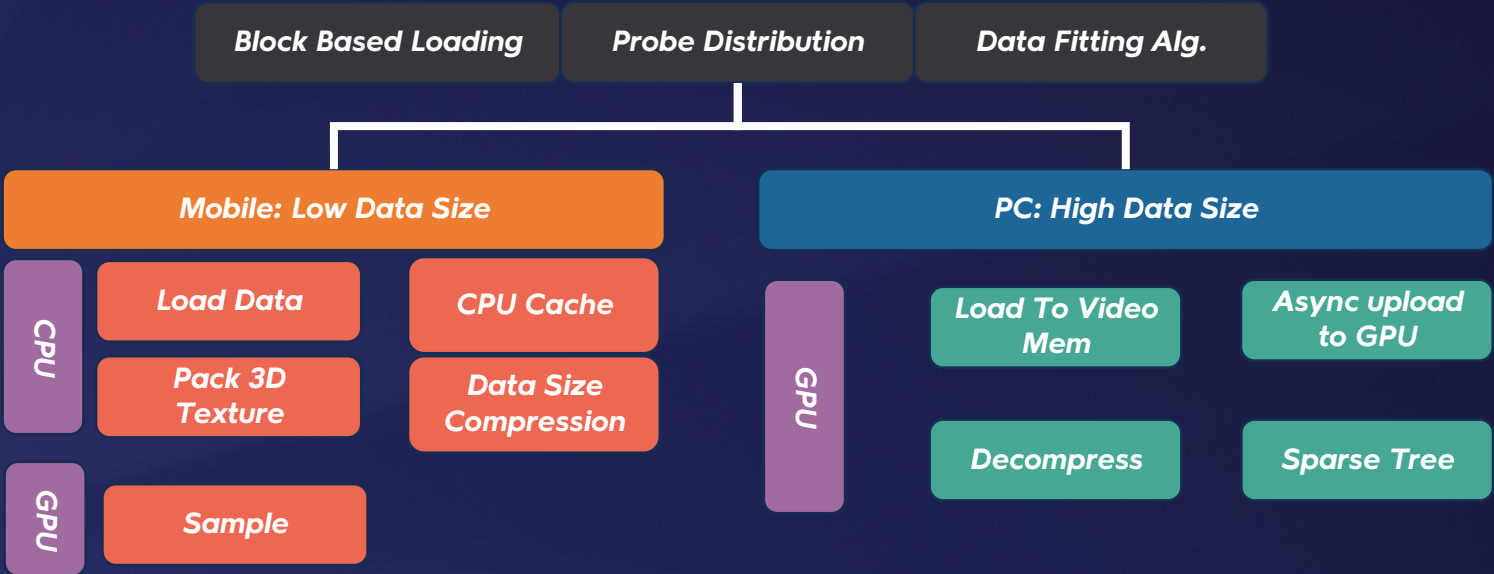
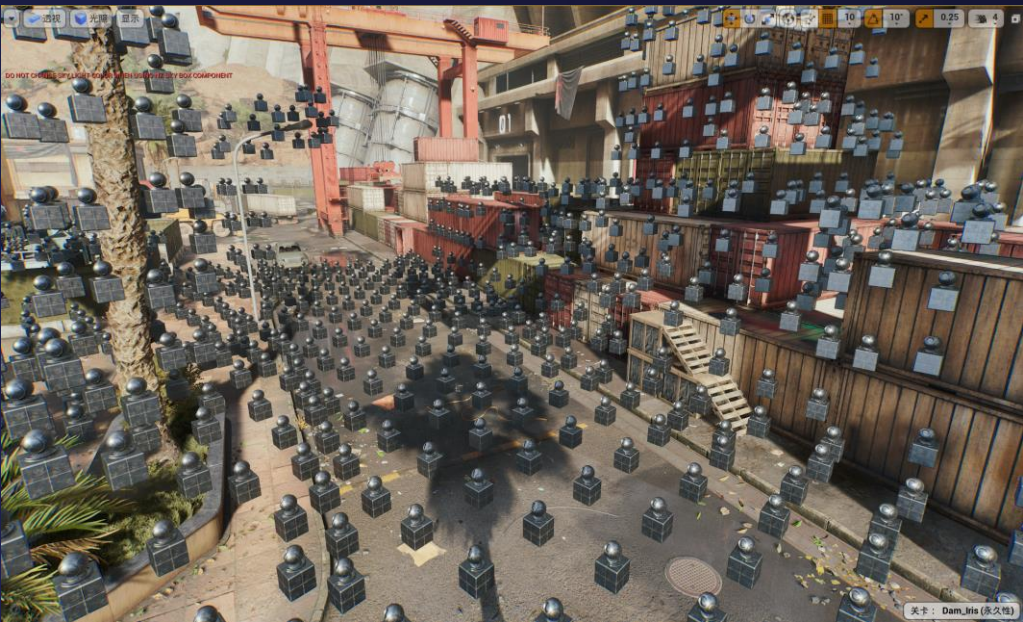
Different Problem

Mobile:

- Package Size
- Heating

PC:

- GI Precision
- Distance
- Video Mem
- Runtime Cost



$$L_s = \left(I_s - \sum_n W_{s,n} B(f_{s,n}) \right)^2 + \frac{\lambda}{2} \sum_n |W_{s,n}|^2$$
$$L_s = I_s^2 - 2I_s \sum_n W_{s,n} B(f_{s,n}) + \left(\sum_n W_{s,n} B(f_{s,n}) \right)^2 + \frac{\lambda}{2} \sum_n |W_{s,n}|^2$$
$$L = \sum_s L_s$$
$$\nabla_{W_{s,nx}} = -2I_s B(f_{s,nx}) + 2 \left[\sum_n W_{s,n} B(f_{s,n}) \right] B(f_{s,nx}) + \lambda W_{s,nx}$$
$$\nabla_{W_{s,nx}}^2 = 2B^2(f_{s,nx}) + \lambda$$
$$\nabla_{B_{f_{s,nx}}}^p = -2I_s W_{s,nx} + \left(\sum_n W_{s,n} B(f_{s,n}) \right) W_{s,nx}$$
$$\nabla_{B_{f_{s,nx}}} = \nabla_{B_T} = \sum_p \left[-2I_p W_p + \left(\sum_n W_{p,s,n} B(f_{p,s,n}) \right) W_p \right]$$
$$\nabla_{B_T}^2 = 2 \sum_p W_p^2$$



- *TOD Sequence manager*
- *Share the same parent sequence*
- *Platform specific sub sequence*



Advanced Cross-Platform Landscape Rendering



- **10KM Large World**
- **32 Materials on Mobile, 256 Materials on PC**
- **Weight based Material ID**
- **Adaptive Virtual Texture Driven**
- **Runtime VT Compression**
- **3 Materials / 500+ Texels Per-Meter Mobile**
- **Cliff Rendering with VT**
- **High-Performance Vista Detail Rendering**
- **Per-Meter Biome Info: Tint, Wetness, AO**
- **Massive Decals**
- **Software Tessellation**
- **All achieved on Mobile!**



Advanced Cross-Platform Landscape Rendering



Adaptive Dynamic Texture Array



Dynamic
Update Slice(s)

85.3MB → 10.6MB

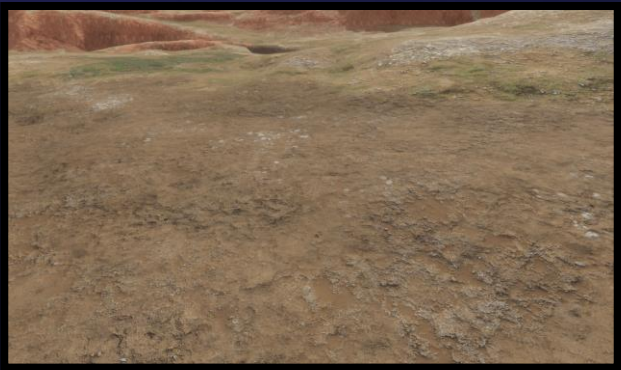
10K Texture → 640x5 Texture Array
133.3MB → 1.95MB

Reduce to 1% Mem Cost

Stochastic Tri-Planar with VT on Mobile



Clipmap Textue Streaming
based Wetness



High-Perf Vista Improvement



380k Quality
with 80k Performance

GDC

-



Platform Features

Specific platform features

- Mouse/Joystick/Mobile Touch Sensitivity
- Gyroscope/Force Feedback
- HUD
- ...

Flexible Features

Different features across platform

- Animation Feature Details
- Character Appearance Details
- Physically correct movement
- Weapon Bullet trajectory Detail
- Detailed Sound & Effects

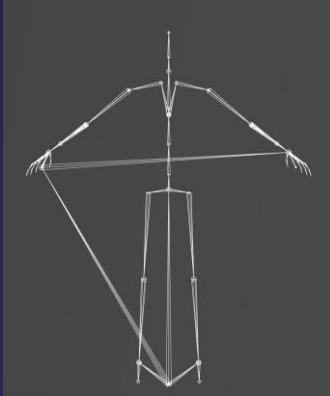
Core Features

Foundation For AAA FPS on all platform

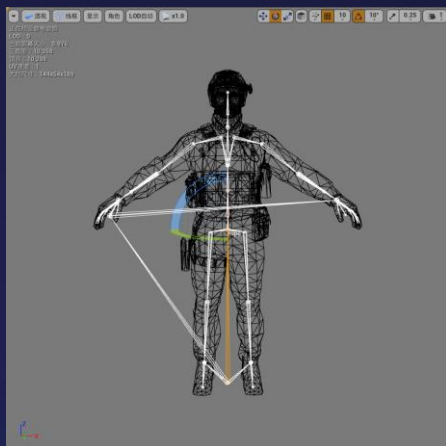
- Gameplay Framework
- Smooth Movement and net code
- Realistic Weapon Shooting
- ...

- Assets production pipeline base on same Skeleton and Animations
- Mesh has additional Bones and LOD on PC
- Additional parts for details on PC

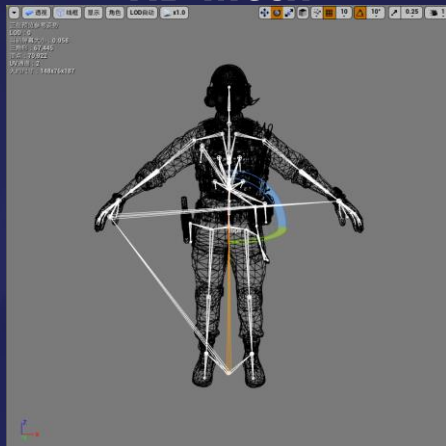
Base Skeleton



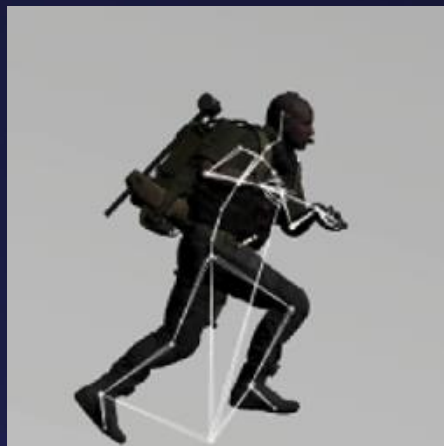
Mobile Mesh



HD Mesh



Unified Animation



PC Details



Mobile Character Animation



PC&Console Character Animation



Flexible Features

- Certain features has backup solution across different platform
- Certain features are extremely limited on mobile

Reload Clip Animation Adaption

High Quality solution

- Animation on Each Clip *
Weapon
- 100+ Animations

Fallback solution

- IK Offset on Clip Animation
- 10+ Animation



Flexible Animation features

- 8-Dir Locomotion
- FootIK / Foot Lock
- Twist IK
- AimOffset
- Reload Adaption
- Grip Adaption
- Death Physical Animation
- Weapon Additive
- Speed Warping
- ...

Feature Planning

- Let high quality feature opening up on important characters, Let mistakes happen on less important characters.

Knapsack problem

- Limited cpu/gpu resources with optimal expressiveness



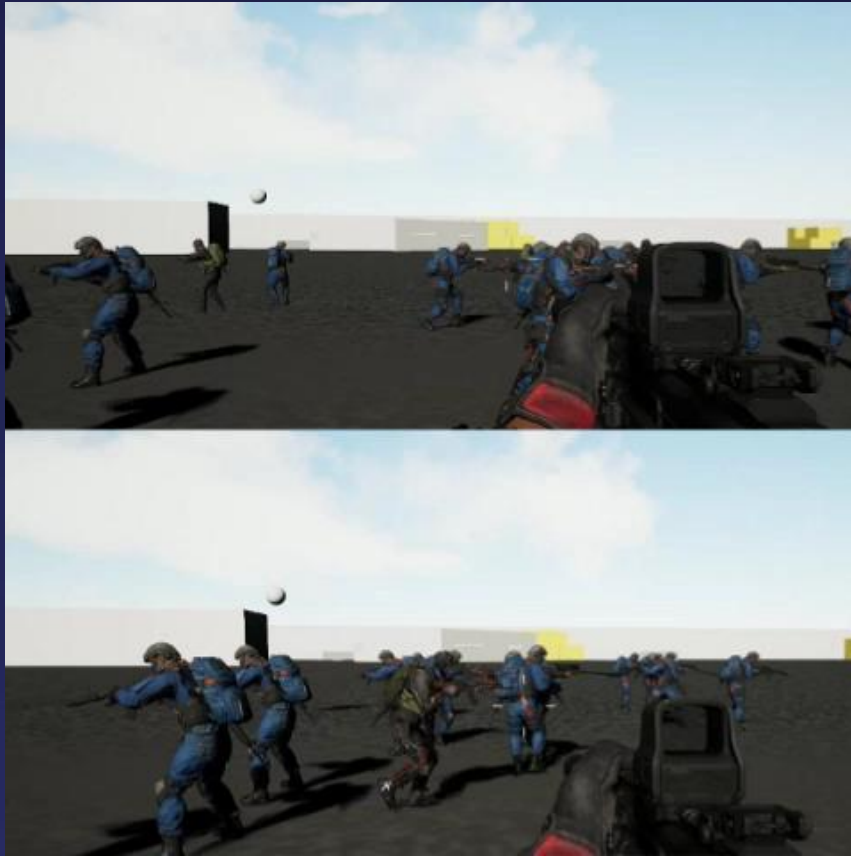
Quality Evaluation

- Single Character Quality $P = \text{Static Quality} * \text{Runtime LOD}$
- All Characters Value = $\text{Min}(\sum_{i=1}^n P_i)$

Constraint

- Performance Budget = (R_{CPU}, R_{GPU})
- Cost Constraint: $\sum_{j=1}^m \text{CharacterCost}_j \leq \text{Budget}$
- Feature Constraint: PCOnlyFeature, InCamera, Radius, etc...

5ms vs 3ms Cost Constraint

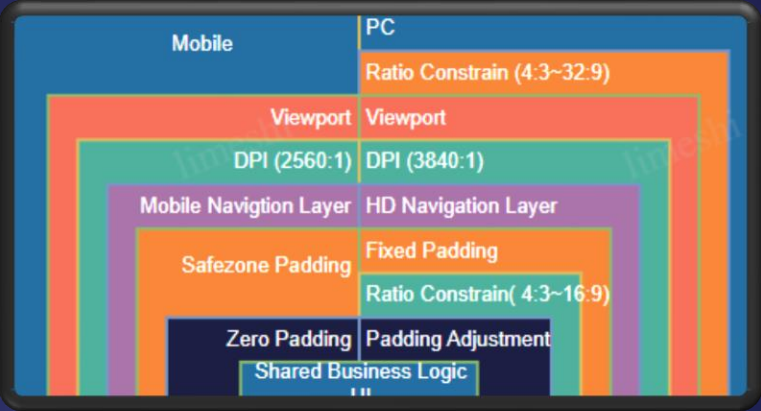


Feature Planning Runtime Result



Shared: most of our business logic UIs
Platform specific: DPI, Ratio, Padding, Navigation

Padding +
Scaling +
Resizing +
Node Culling



UI Resolution

Resolution	Designer Res.	Editor Res.	PC Runtime	Mobile Runtime(Downscaled in cooking process)
Mobile	2560*1440	2560*1440	N/A	1920*1080
Shared	2560*1440	2560*1440	3840*2160 (Downscaled By DPI : 66.67%)	1920*1080
PC Special	3840*2160	3840*2160	3840*2160	N/A

Compression & Rendering

	Format	MipMap	Filter
Mobile	ASTC_5x5	No	Nearest
PC Special	BC7	Yse	Trilinear



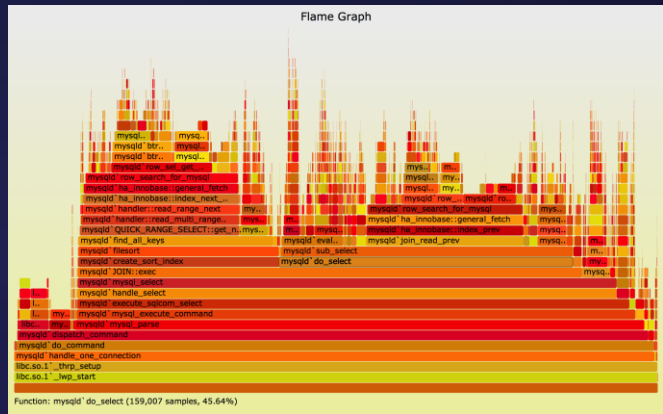
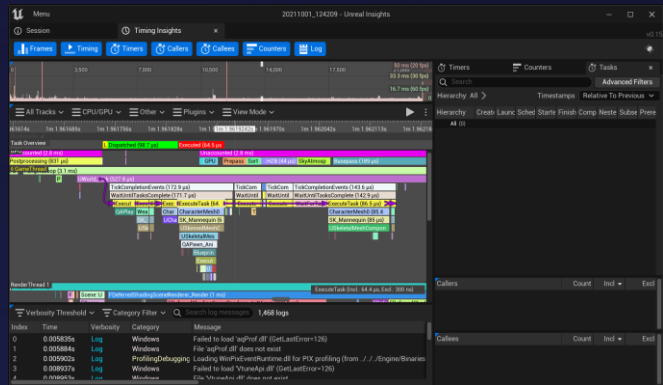
- Rendering
- Gameplay
- Performance



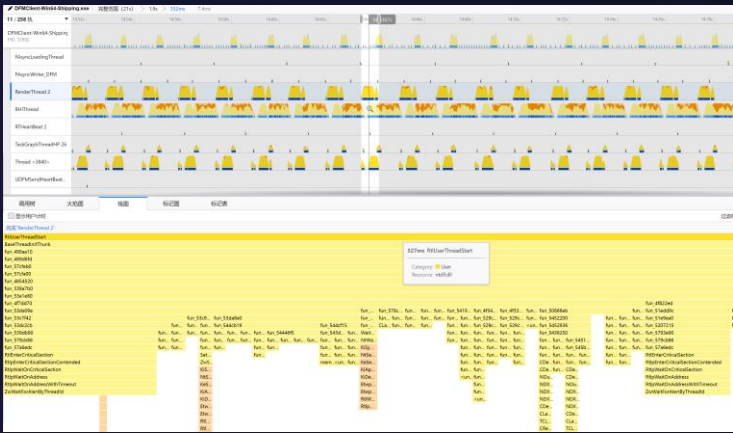
Performance - Metaperf



Profile Tools	Advantages	Disadvantages
UE Frontend / Insights	<ul style="list-style-type: none">- Engine Embedded Tools- Frame-by-Frame Analysis- Custom Game Data	<ul style="list-style-type: none">- Based on code instrumentation- Significant impact on Performance
Superluminal Performance / Pix for Windows	<ul style="list-style-type: none">- Thread Dispatch- Context Switches- Kernel Level Stacks- GPU Capture(PIX)- Minimal impact on performance	<ul style="list-style-type: none">- Generic, not optimized for games- Difficult to extend and customize- Difficult to collect over a long period- Unable to analyze by frame
Android/Linux (simpleperf / perf)	<ul style="list-style-type: none">- Minimal impact on performance- No instrumentation Requires	<ul style="list-style-type: none">- LIKE ABOVE ~
Xcode Instruments (Time Profile)	<ul style="list-style-type: none">- LIKE ABOVE ~	<ul style="list-style-type: none">- LIKE ABOVE ~ &&- Difficult to extend and customize

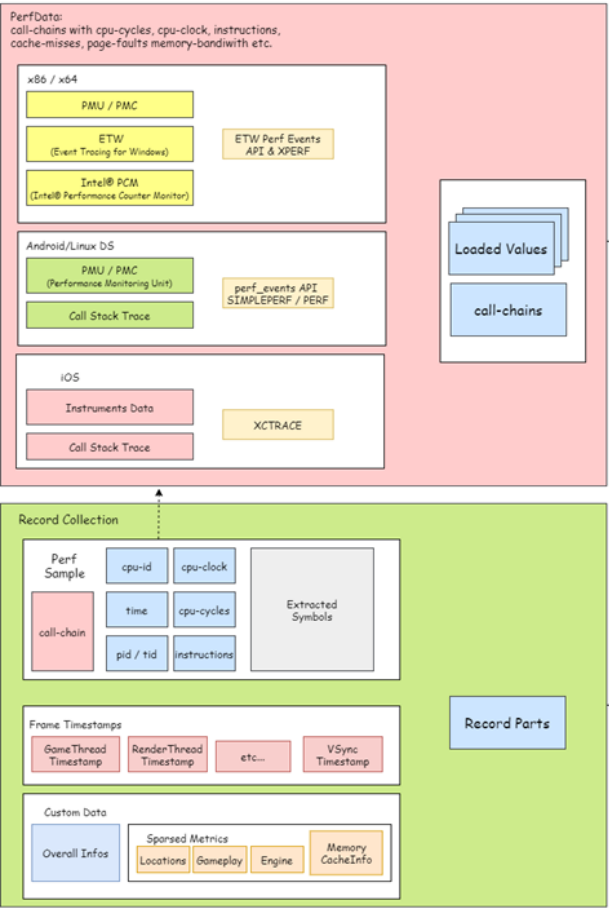


Our Goal

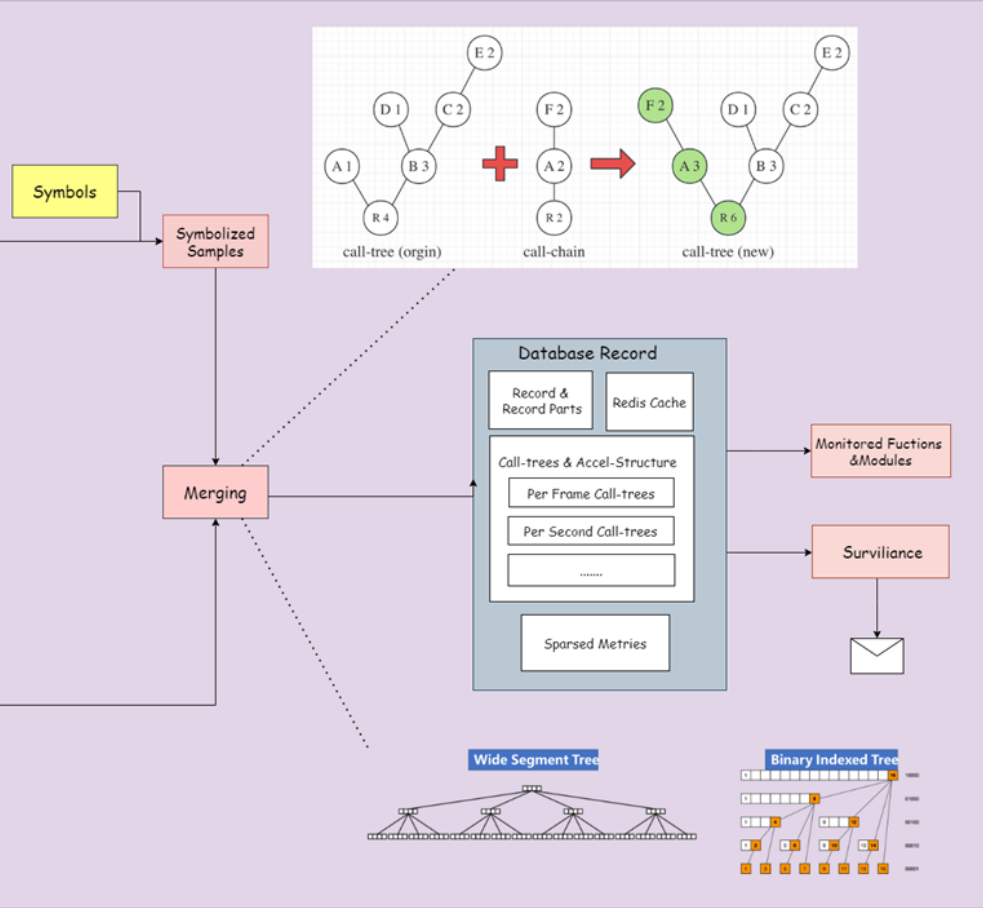


Metaperf Profiler

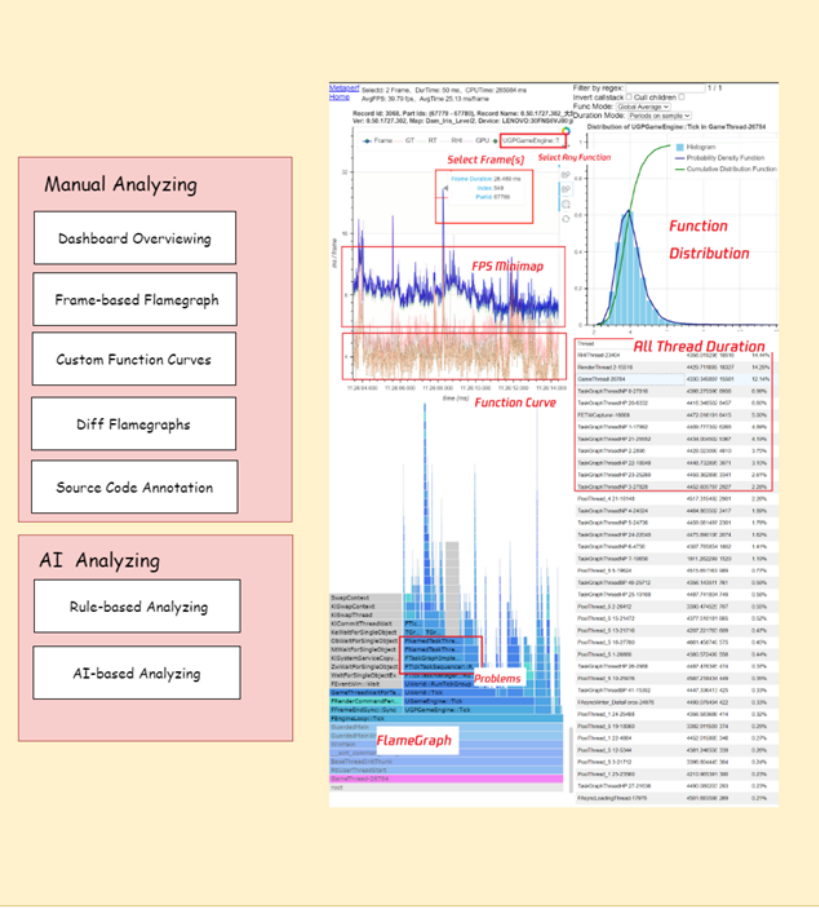
Record



Process



Analyze



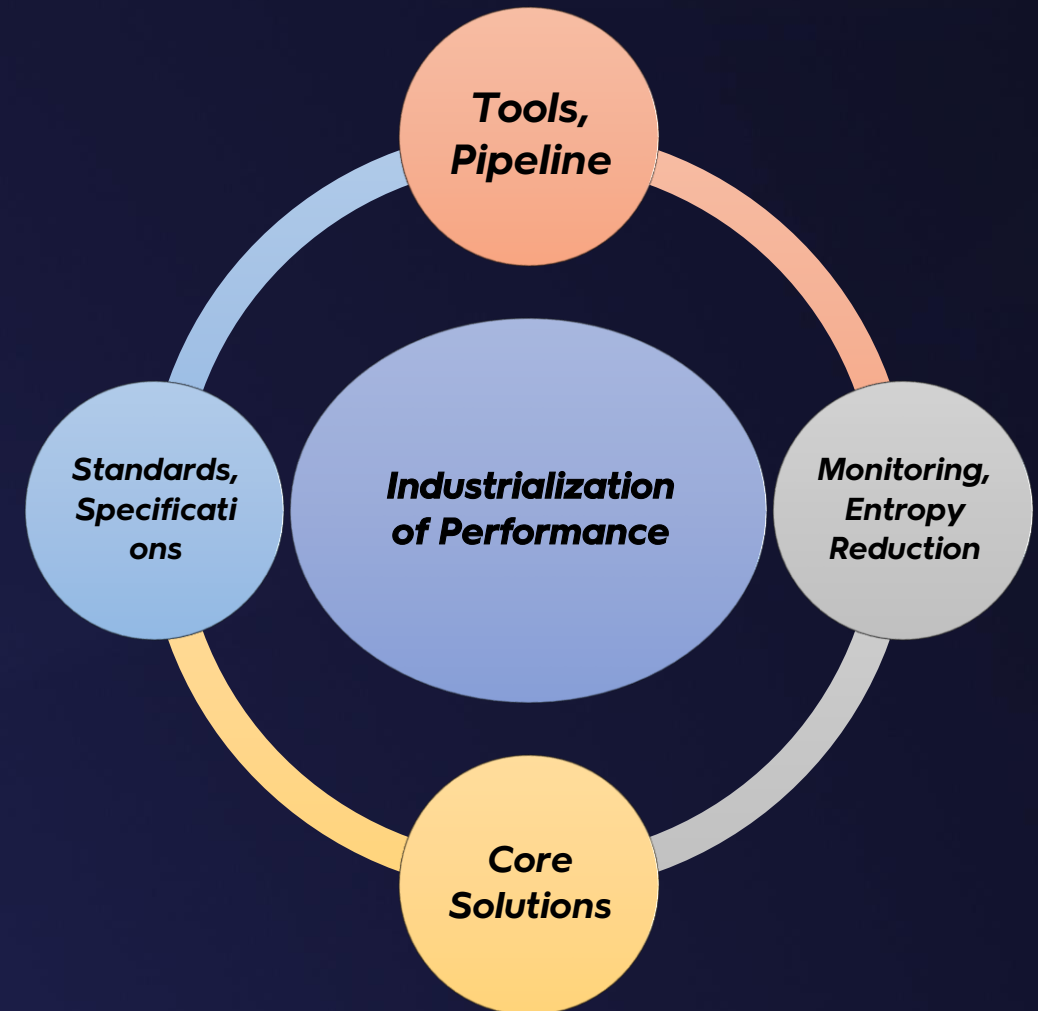
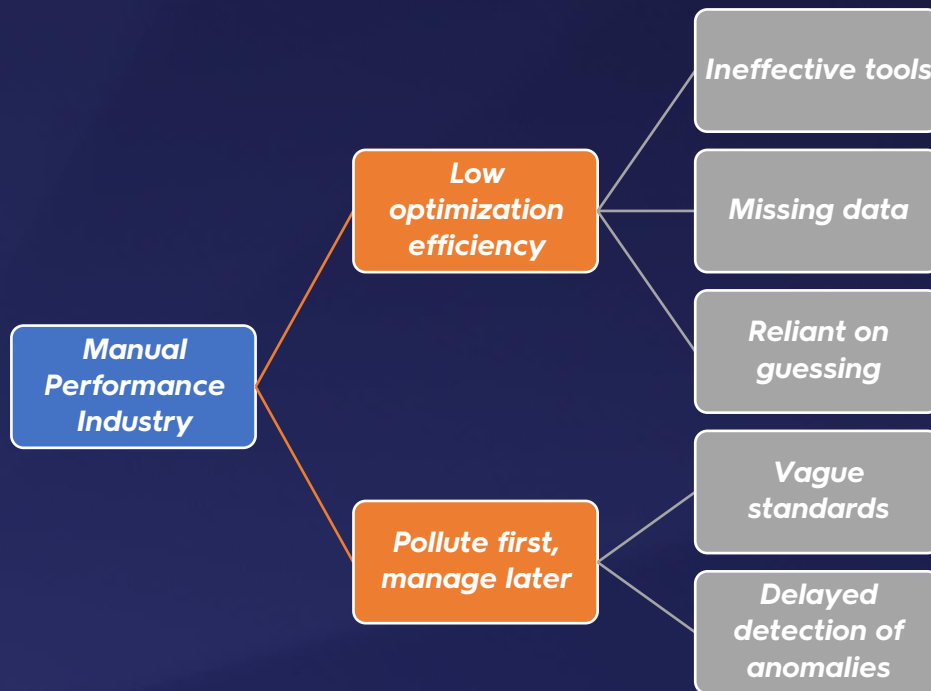
Pearson Correlation Coefficient

Tool Automatically Identifies Key Functions Causing Stuttering"

$$r = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}}$$

Evaluation and Effectiveness

- Approach widely adopted in Delta Force
- Established full performance analysis system
- Over 10000 records resolved 1000+ issues



- *There is no right solutions but only suitable ones*
- *This presentation only gives the possible way of cross platform development*
- *Find out the solution based on the product and team situations*
- *Tools are way more important*
- *Hope this can give some help*

Special Thanks



DELTA FORCE



We are hiring! cherylqiu@tencent.com

*Thank
You*



GDC

TEAM JADE



DELTA FORCE
HAWK OPS

FOLLOW US! [PLAYDELTAFORCE.COM](https://playdeltaforce.com)

FOR GLOBAL PARTNERSHIP INQUIRIES,
PLEASE REACH OUT TO US AT DELTAFORCE@PROXIMABETA.COM

📍 BOOTH S1127

