# FROSTBITE™ 3

# Beyond the Battlefield

Adapting a technology stream to ever-evolving console platforms, game designs, and opposing game genres.

Graham Wihlidal, Sr. Rendering Engineer - Frostbite

# Graham Wihlidal

- Senior Rendering Engineer for Frostbite
- Previously at BioWare for almost a decade
- Specialist in:
  - Engine architecture
  - Low level optimizations
  - Graphics and console hardware
  - GPU driver implementation
- Author of Game Engine Toolset Development

# Introduction

- Frostbite 2 was originally built for Battlefield 3 internal at DICE
- Other teams at EA started to express interest in using Frostbite
- Prior to Battlefield 4 launch:
  - Engine was transitioned for next gen consoles
    - Frostbite 3
  - Frostbite became its own team within EA
    - No longer internal to DICE
- Frostbite 3 now used by most teams at EA!
  - Many different game genres…

BATTLEFIELD 4

DRAGON AGE
INQUISITION

Dragon Age
Inquisition

STAR WARS
BATTLEFRONT
EA

STAR WARS
BATTLEFRONT
EA

STAR WARS
BATTLEFRONT
EA

PLANTS vs ZOMBIES
GARDEN WARFARE
TeLo
PoRd
erer
CRETE

PLANTS VS ZOMBIES
GARDEN WARFARE

SOUTH
49
Grand Peak Dr
882
PO
REDVIEW COUNTY
GHOST
OFFICIAL
DIAL
9-1-1
#NEED FOR SPEED

California
GHOST
SPEEDHUNTERS

EA SPORTS
RORY McILROY
PGA TOUR

TPC SAWGRASS
KEEGAN BRADLEY
EA SPORTS
RORY McILROY
PGA TOUR

# Gen4 Challenges: Battlefield 4

- Extensive PS3\360 code to migrate (i.e. SPU jobs)
- Cross generational! (5 platforms)
- Console launch title development was ‘interesting’
  - Early adoption is always a bumpy road
  - Part of the fun!
    - Uncharted territory
    - Creative problem solving

# Gen4 Challenges: Battlefield 4

- Larger asset sizes and in greater quantities
  - i.e. BC7 compression times, photogrammetry, more draw calls, etc.
  - We had some systems buckle a bit under the exponential growth
    - i.e. Content and build pipelines in some areas
    - See: Scaling the Pipeline

# Gen4 Challenges: The Good

- Frostbite was already 64 bit from PC
- Already had an optimized Direct3D 11 renderer
  - Many engines still on Direct3D 9 & 10 at this time
- Architecture already supported multiple platforms
- Most systems already designed with scalability and parallelism
- New consoles are effectively PC platforms
  - x86 64bit
  - Same GPU\CPU vendor for both (AMD)
  - Great support from first party (Microsoft and Sony)

# Gen4 Challenges: The Bad

- Battlefield 3 was 30fps, Battlefield 4 needed to be 60fps
  - A huge undertaking to optimize for
- Gameplay systems were a problem
  - Was single threaded in many places -> had to jobify
  - CPU code is at best 2-3x faster than PS3\360
    - While GPU is about 8x faster
  - Some cases we have measured 2x slower!!
- Console platforms and tool chains were immature
  - Shader compiler bugs
  - Iteration time was very low
  - First party systems available last minute

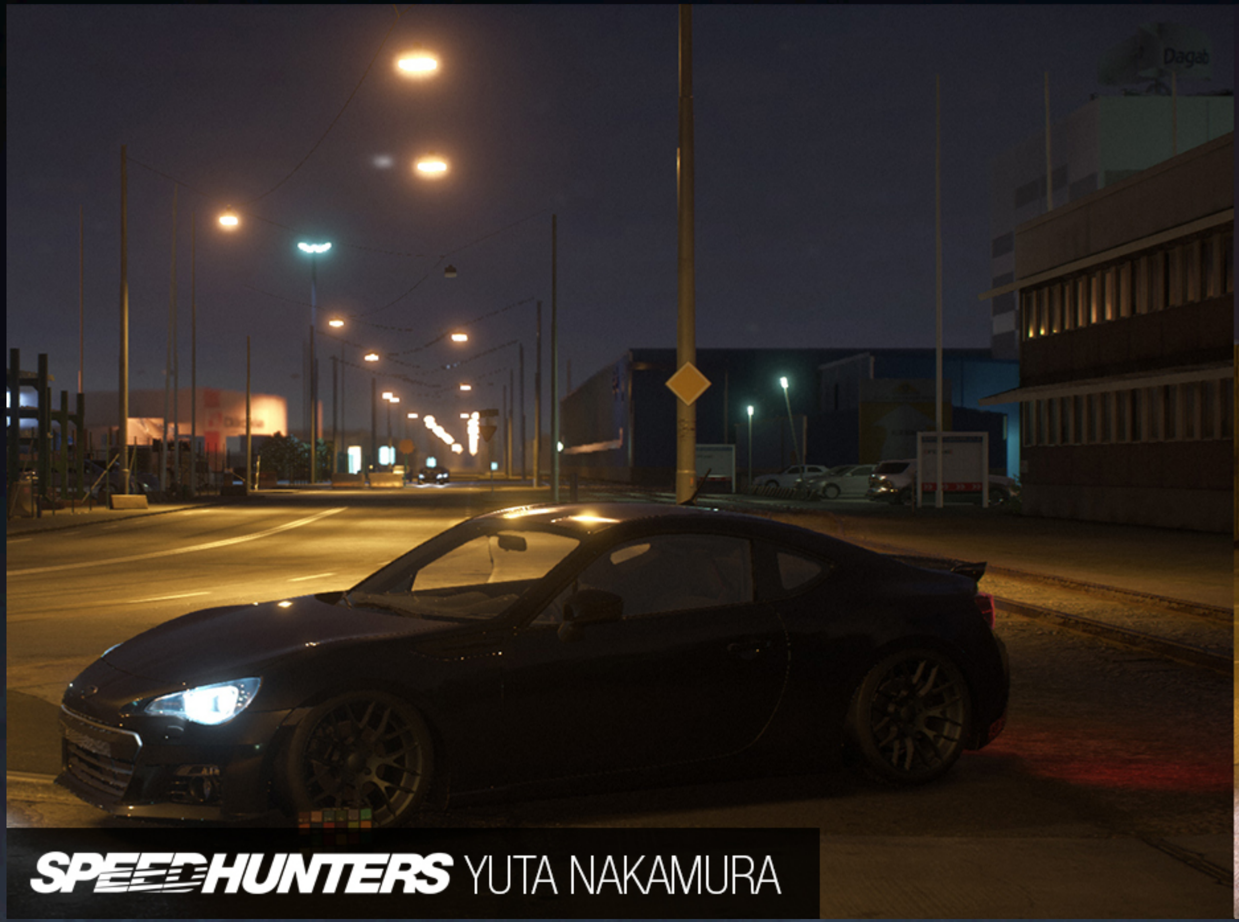
# Gen4 Challenges: The Bad

- Rendering systems also had some challenges
  - Increased asset counts
  - CPU didn't scale up as much as GPU did.
  - Cross generational scaling (gen3 to gen4 to pc)
- Had to implement very complex systems in short time
  - Memory manager
    - Had OS bugs like two virtual addresses pointing to same physical address…
  - Job scheduling
  - New rendering backends (GNM, D3D11.x)

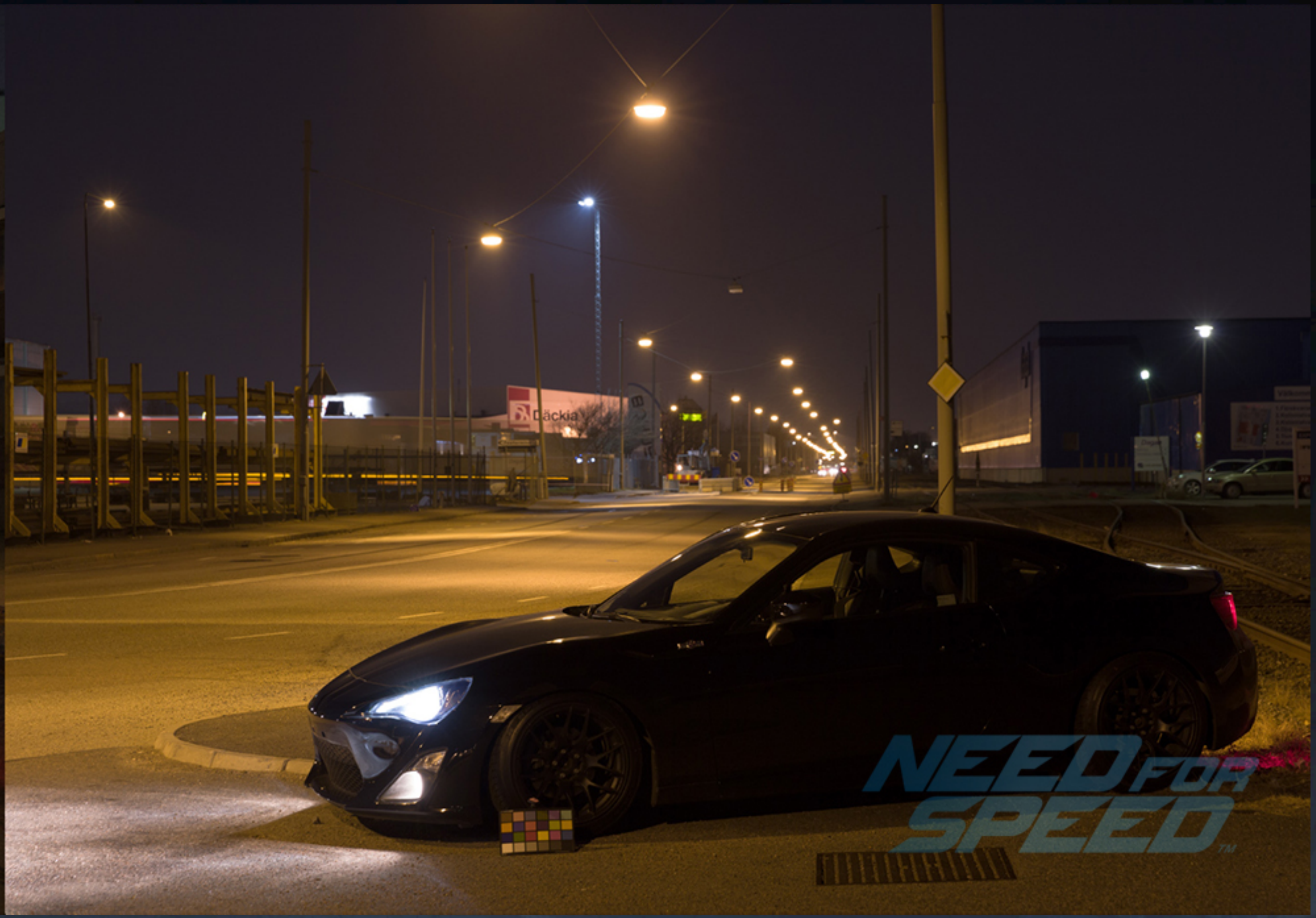
# Current Challenges

- Rendering has progressed into physically based (PBR)
  - Frostbite has done a MAJOR push to revamp our rendering
  - See: SIGGRAPH: Moving Frostbite to PBR
  - See: SIGGRAPH: Unified Volumetrics
  - See: SIGGRAPH: Stochastic Screen-Space Reflections
- Upcoming games want to render much more!
  - 6x more draw calls
  - Frostbite 3 was already efficient at scaling (needed for Battlefield)
  - Linear cost for Pre PBR and PS3\360
    - Well defined problems and approximations
  - Exponential cost for PBR, new rendering systems and PS4\XB1
    - Cutting edge research

# Physically Based Rendering

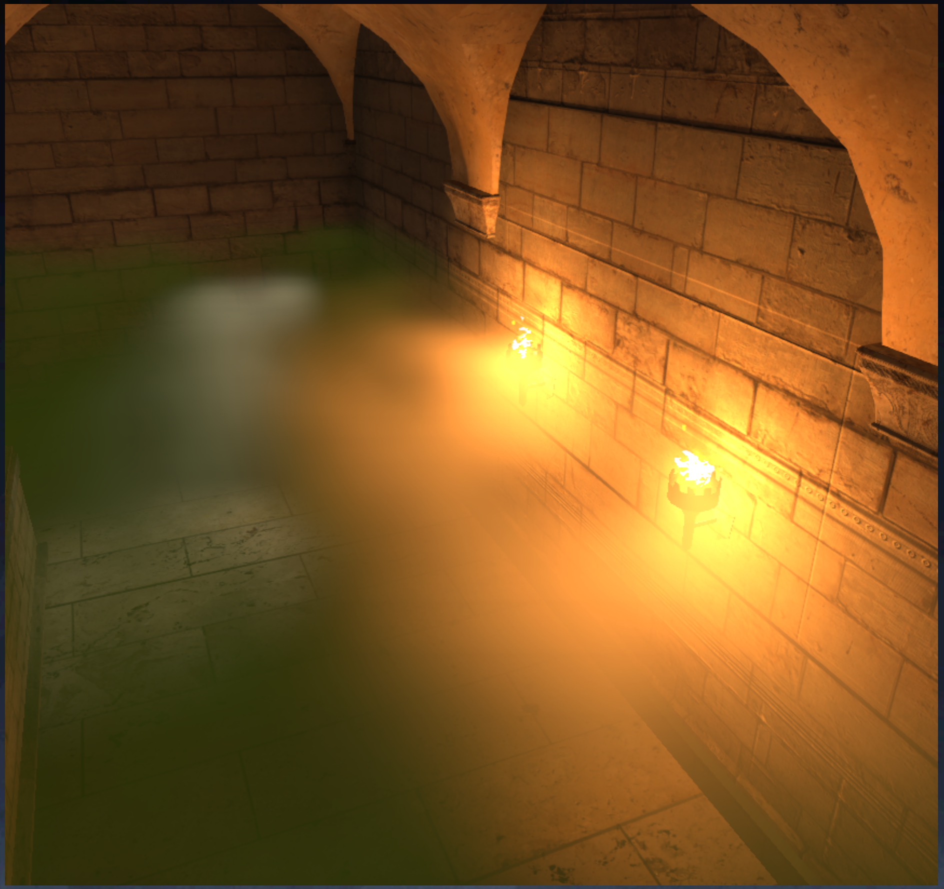


SPEEDHUNTERS YUTA NAKAMURA

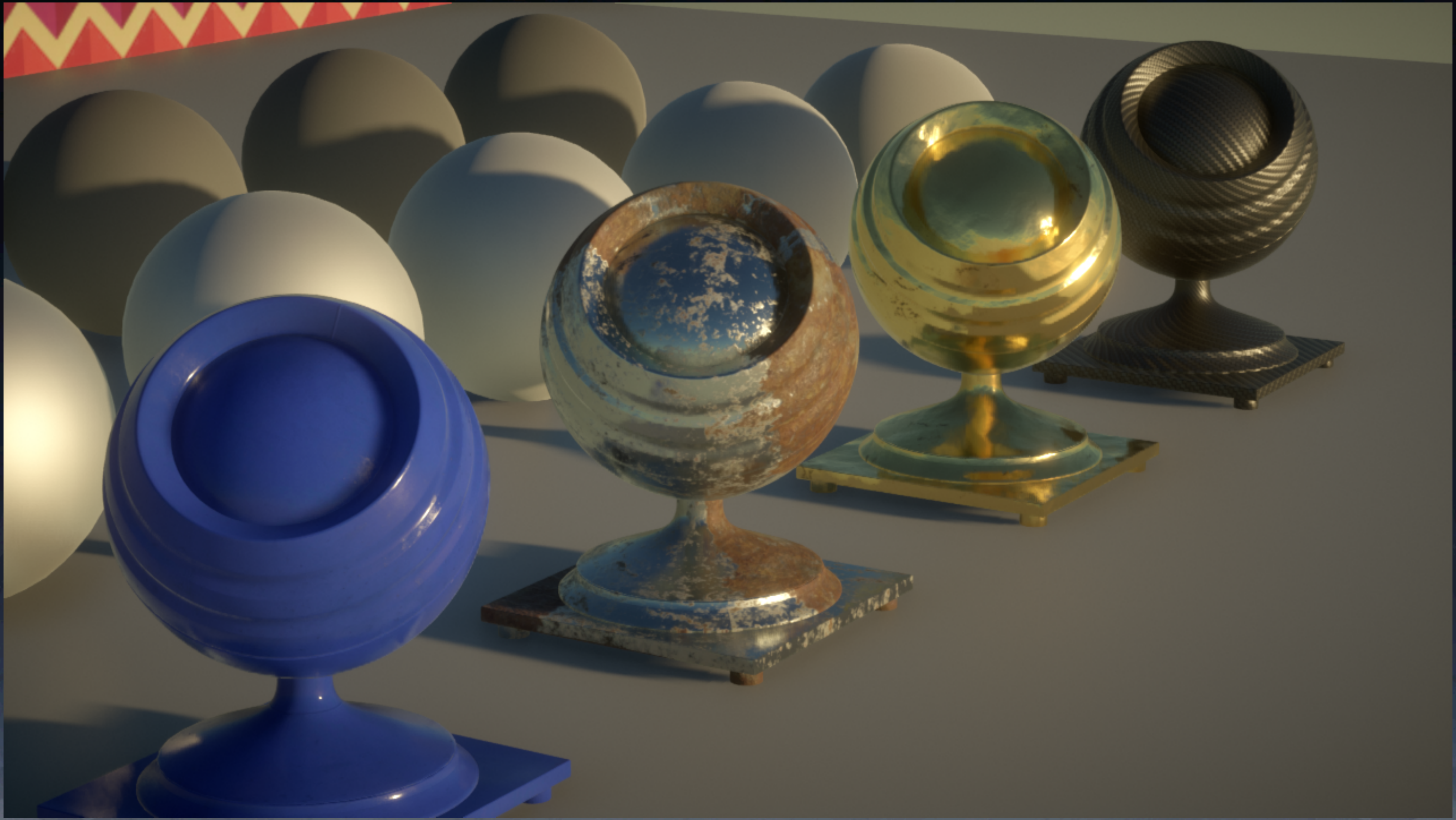


# Displacement Mapping

# Unified Volumetrics



# Screen Space Reflections



# Current Challenges

- Result: Beautiful looking games! ☺
- Result: Unhappy performance! ☹

- The key factor is scalability
  - Increased asset counts and cost
  - [CPU] rendering systems
  - [CPU] submission to GPU
  - [GPU] rendering passes

# Improving Scalability

- Major optimization efforts ongoing at engine and game level
  - Shaders (Algorithms and Shader Compilers)
  - Passes (Reduction and Simplification)
  - Improvements to Culling, Occlusion, and Shadow Map Generation
    - Sensitive to asset counts
    - Reducing overhead (Proxy meshes, improved cache coherency, etc.)
    - Further optimized our already efficient systems
      - See: Culling the Battlefield
  - GPU driven rendering pipelines

# Improving Scalability

- Have already made significant optimizations and improvements
  - Still need to squeeze a bit more performance, though…

# Improving Scalability

- Have to look at full rendering stack to get the big picture
  - This includes the graphics APIs and drivers!
- PS4 is very low level
  - Improvements here are on the developers, instead of Sony
  - Our backend is already very efficient, but can always be more optimal
  - We have high level information and knowledge
    - Easier to write a “driver” that favors patterns in our engine and games
    - Unlike PC drivers (which includes XB1 DX11)

# Graphics APIs

- Direct3D 11
  - High amount of CPU overhead due to a number of reasons
    - Lack of high level engine knowledge\context
    - Extensive validation of inputs to protect developers
    - Robustness over performance
    - Many unnecessary flushes and cache invalidation
    - Very inefficient parallelism

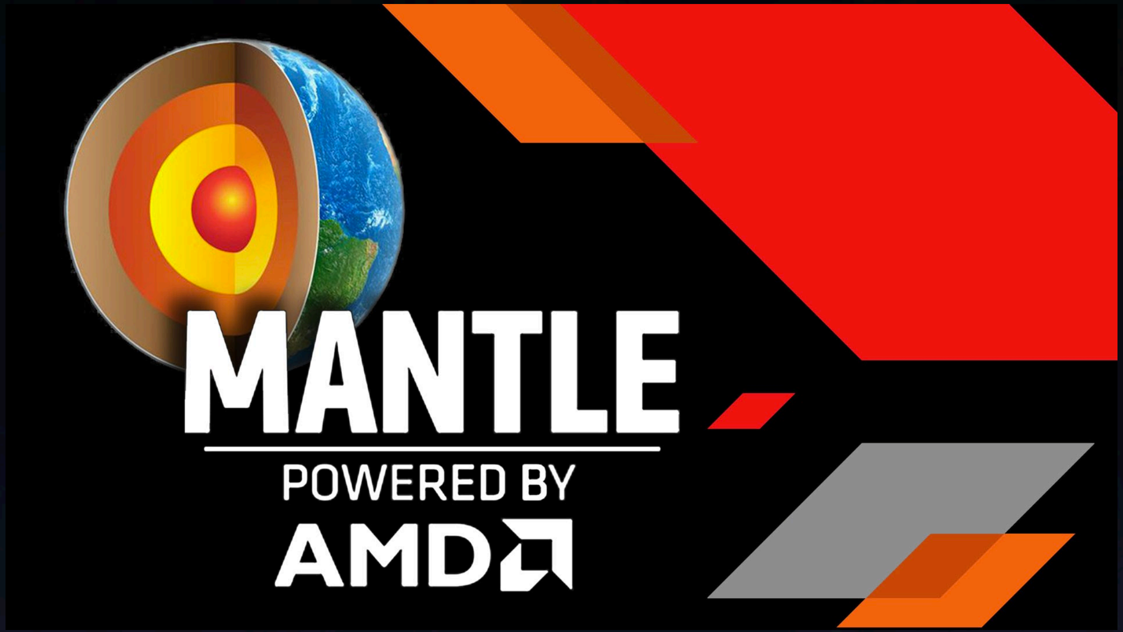
# Graphics APIs

- Direct3D 11
  - Architecture does not easily allow for GPU extensions
    - Limit engines from taking advantage of new hardware features
- Windows OpenGL – Similar Story (though a bit better)
- Apple OpenGL – Don’t even go there…

# Graphics APIs

- Frostbite has been pushing IHVs, Microsoft, and Khronos for years to give AAA developers something better!
  - They didn’t think we could handle a low level API
    - GPUs are complex and with many nuances
    - PS4 gave us the opportunity to truly prove our competence
- In collaboration with AMD, Frostbite helped pioneer Mantle
  - Extremely successful because the results kicked the industry into gear

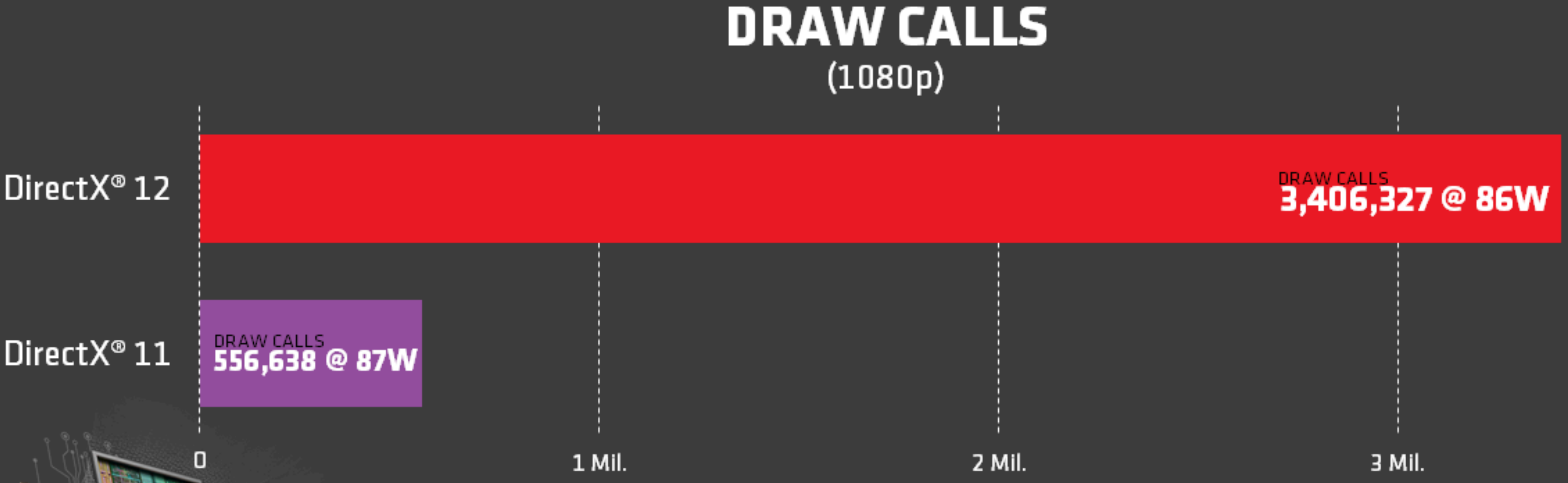
# Graphics APIs



# DIRECTX® 12 SOC PERFORMANCE/WATT

## 3DMARK® API OVERHEAD FEATURE TEST

AMD

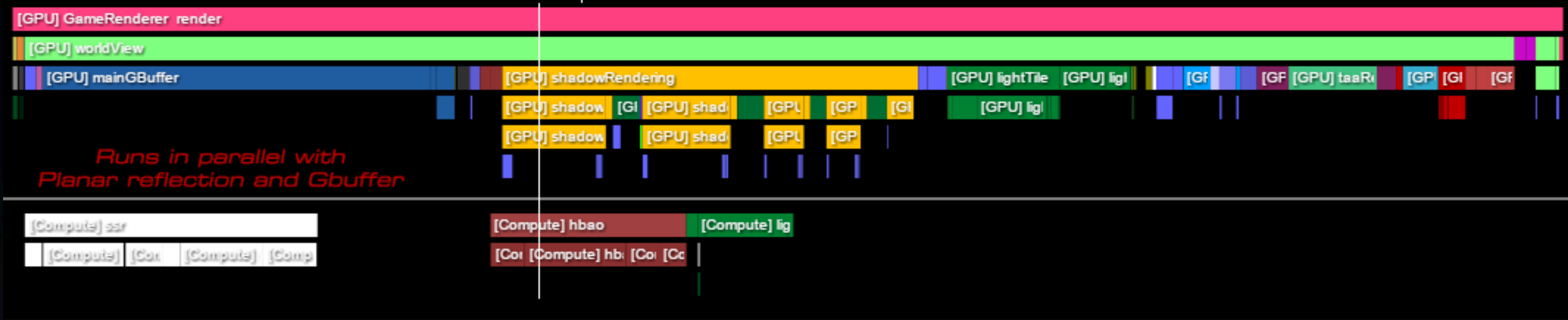


**AMD A10-7850K**

+511% PERF/WATT
WITH DIRECTX® 12*

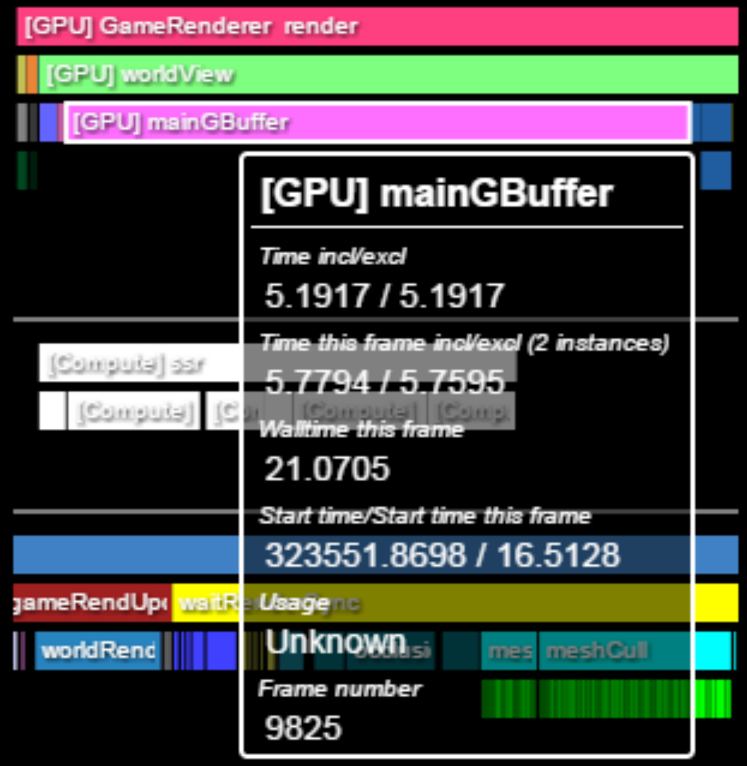
# Graphics APIs

- CPU parallelism is not the only goal!
- Current GPUs support parallelism
  - Known as “Asynchronous Compute”
  - Allows for new performance tricks
  - Legacy APIs serialized potential graphics parallelism
    - i.e. Graphics -> Compute -> Graphics
  - Supported on PS4, Vulkan, DX12, Mantle
    - We now overlap compute with graphics when possible
- We have moved a lot of our GPU work to async compute
  - We want to use it wherever there is a performance win



Runs in parallel with
Planar reflection and Gbuffer

ASYNC SSR with GBUFFER TIMER



[GPU] mainGBuffer

*Time incl/excl*
5.1917 / 5.1917

*Time this frame incl/excl (2 instances)*
5.7794 / 5.7595

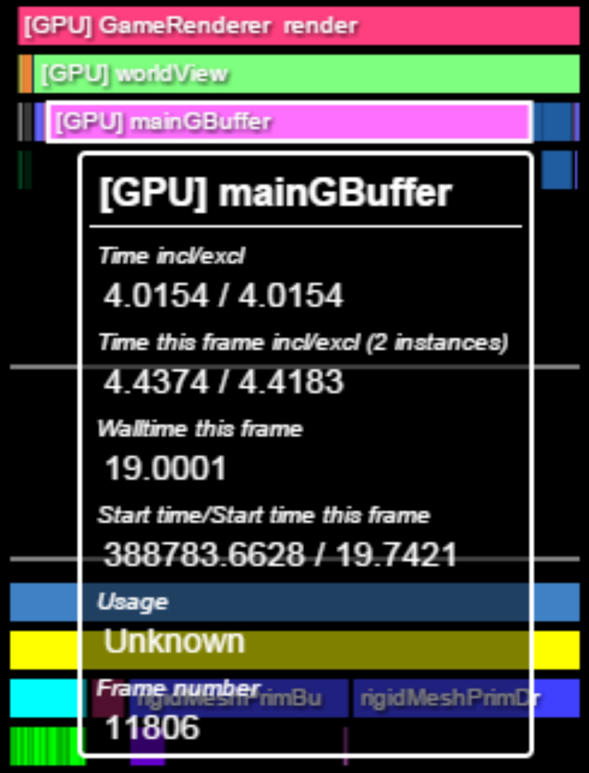
*Walltime this frame*
21.0705

*Start time/Start time this frame*
323551.8698 / 16.5128

*Usage*
Unknown

*Frame number*
9825

REGULAR GBUFFER TIMER



[GPU] mainGBuffer

*Time incl/excl*
4.0154 / 4.0154

*Time this frame incl/excl (2 instances)*
4.4374 / 4.4183

*Walltime this frame*
19.0001

*Start time/Start time this frame*
388783.6628 / 19.7421

*Usage*
Unknown

*Frame number*
11806

Frame is TWO
milliseconds
shorter
however!

# Summary

- Frostbite has evolved significantly over the past decade
- Game team adoption has grown exponentially
- No longer developing tech for a single genre
- Architectures need to be able to scale linearly
- New consoles allow us to raise the bar on fidelity
- Important to develop flexible software
- Aim for full parallelism on CPU, but also GPU!
  - Upcoming graphics APIs let us optimize the whole stack

# Thank You!

# Questions?

graham@frostbite.com