

Gears of War 4: Creating a Layered Material System for 60fps

Colin Penty

The Coalition, Microsoft Studios
858 Beatty St
Vancouver, BC, Canada V6B 1C1
colinpen@microsoft.com

Ian Wong

The Coalition, Microsoft Studios
858 Beatty St
Vancouver, BC, Canada V6B 1C1
ianwong@microsoft.com



Figure 1: The iconic Gears of War lancer using The Coalition Material Layering System. ©Microsoft

ABSTRACT

We have created a new material system for Gears of War 4 inside Unreal Engine that allows artists to layer dozens of materials with complete material tuning control and flexibility - then cook out the results in-engine for efficient run-time performance.

CCS CONCEPTS

• **Computing methodologies** → **Computer graphics**; *Rendering*;

KEYWORDS

materials, realtime rendering, tools, performance optimization

ACM Reference format:

Colin Penty and Ian Wong. 2017. Gears of War 4: Creating a Layered Material System for 60fps. In *Proceedings of SIGGRAPH '17 Talks, Los Angeles, CA, USA, July 30 - August 03, 2017*, 3 pages.

DOI: 10.1145/3084363.3085026

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGGRAPH '17 Talks, Los Angeles, CA, USA

© 2017 Copyright held by the owner/author(s). 978-1-4503-5008-2/17/07...\$15.00
DOI: 10.1145/3084363.3085026

1 OVERVIEW

For Gears of War 4 we had the challenge of shipping our multiplayer mode at 1080p / 60fps on Xbox One while maintaining a high visual bar. To achieve this high visual bar, artists needed to be empowered with complete material tuning and layering control. Unreal Engine 4 has an incredibly powerful material system, but exposing material controls to the artists can come with a high run-time performance cost that isn't acceptable at 60fps. Finally, creating these materials is very time consuming and costly for AAA - so we needed a way to curate a shared material library allowing maximum reuse without the run-time performance costs of a traditional layered material. We present a solution to the material problem many game studios are faced with.

2 IMPLEMENTATION

Inspired by Ready at Dawn's: *Crafting a Next-Gen Material Pipeline for The Order: 1886* [Neubelt and Pettineo 2013], we solved all the issues outlined above by creating a custom Material Layering System in Unreal Engine 4 that sits on top of the existing UE4 material system. The system allows the artist to add materials from a shared library to a material stack, and composite the materials with various blend modes and masks very similar to Substance Painter

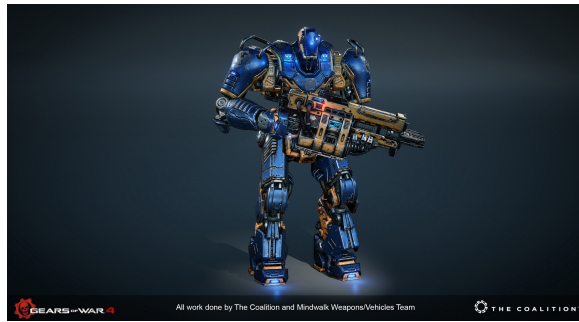


Figure 2: Mega-Mech Created with The Coalition Material Layer System

or Photoshop. The result is then cooked down for optimal run-time performance. Having a shared material library inside Unreal allows us to ensure everything is PBR compliant and dramatically decreases iteration time as the artist does not have to jump out to a 3rd party application to tune the look or react to Art Director feedback.

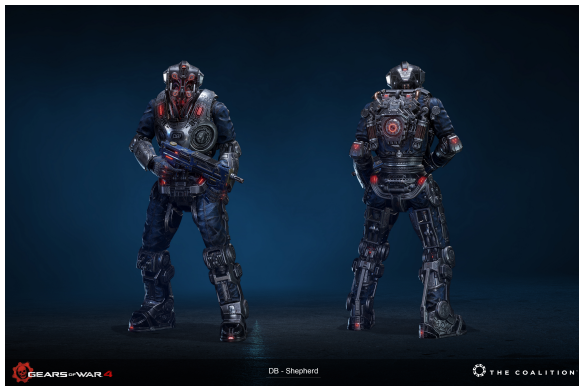


Figure 3: All weapons, characters, and environments, in Gears of War 4 were created using the Material Layer System

3 WORKFLOW

There were numerous technical and workflow constraints to overcome allowing the tool to scale across multiple artist domains and numerous external partners across the world. The Coalition Material Layering System allows hybrid materials with static and dynamic parameters, as well as hybrid dynamic and static material layers. This allows artists to make complex tiling ground materials or visual effects using the system and still leverage the material cooking step to get additional performance back. The material cooker will identify live-blended layers and cook them individually before compositing them with the rest of the stack to ensure maximum performance if necessary. The system supports material nesting, so a material stack can be created and shared (ex: cobblestone, dirt, paint) and, if desired, nested directly into another material stack which can then further layer grass and pebbles on

top. We built upon the foundations of the Ready at Dawn paper by focusing on flexibility (arbitrary material nesting), consistency (all artist domains use the same foundational system), and artist control (we encourage complex tuning and blend parameters since they are cooked out anyways). We have integrated the Material Layering System into Unreal Engine 4 as a 1st class system but the fundamentals of the system could be used in any engine.

Full Screen Quad Test on Xbox One	Milliseconds
Coalition Material Layer System - 1 layer, 100 controls	1.37
Vanilla Base Material - 1 layer, 5 controls	1.5
Coalition Material Layer System - 4 live layers	1.9
Vanilla Base Material - 4 live layers	2.9

Figure 4: We have optimized the material cook step to output the most optimal data set possible for console hardware.

4 TECHNOLOGY

There are a few systems that form the foundation of the material layering system, including the material cooker (which we call the Smelter), the new Material Layer UI, which allows re-ordering of the material stack and supports both nested materials and individual materials, and finally dramatic accelerations in material compile times to allow artists to re-structure materials on the fly using the Material Layer UI and incur almost no wait for shader compilation. The tech artists define material functions for the artist to use in their layer stacks such as: Base Layer, Normal Overlay, Normal from Mask, etc. The artists can then build a layer stack using whichever functions they choose, optionally inputting masks they've built offline in Photoshop or Substance. As the layers are added, under the hood the system is building a new Unreal material which is an aggregation of all the layers used. Artists can specify layer blend modes (ex: multiply, overlay), and specify if a layer is live or to be cooked. An artist may want, for example, a detail normal map layer to stay live so the texel density is not lost on cook. Once saved, the system will cook the material down to its simplest form. The cooker will take the artists inputs as recommendations but ultimately do what makes the most sense. The cooker is able to reduce instruction count helping primarily the Xbox Vector ALU, as well as reduce texture count helping Texture efficiency. In essence, it's a system that sits comfortably between the complexity of the material graph and the ease of use of Unreal's Material Instance Constants allowing complete control with optimal results.

REFERENCES

David Neubelt and Matt Pettineo. 2013. Crafting a Next-Gen Material Pipeline for The Order: 1886. ACM SIGGRAPH 2013 Course Notes. (July 2013). DOI: <https://doi.org/10.1145/2776880.2787670>