Design and Implementation of Modern Production Renderers

Matt Pharr
Google, Inc.
Mountain View, California, USA
matt@pharr.org

Brent Burley
Walt Disney Animation Studios
Burbank, California, USA
brent.burley@disneyanimation.com

Per Christensen
Pixar Animation Studios
Seattle, Washington, USA
per@pixar.com

Marcos Fajardo
Solid Angle / Autodesk
Madrid, Spain
marcos@solidangle.com

Luca Fascione
Weta Digital
Wellington, New Zealand
lukes@wetafx.co.nz

Christopher Kulla
Sony Pictures Imageworks
Culver City, California, USA
ckulla@imageworks.com

ABSTRACT

Over the past decade, production rendering has moved from primarily using Reyes-based algorithms to being based on path tracing and physically based approaches. The developers of five of the most significant production renderers have each recently written comprehensive systems papers on their renderers, describing the challenges of modern production rendering and the systems they have respectively built to solve them. In May 2018, these papers were published in a special issue of *ACM Transactions on Graphics*.

In this panel, the developers of these renderers will go into depth on how the challenges of production rendering have influenced the systems they've built and how they have developed new techniques to make path tracing viable in practice. These systems are remarkably varied in some of their core design decisions; the panelists will also compare and contrast their own design decisions with respect to topics like precomputation versus runtime computation, RGB versus spectral rendering, and out-of-core rendering versus requiring scene geometry to fit into memory.

CCS CONCEPTS

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

KEYWORDS

Path tracing, production rendering, graphics software systems

ACM Reference Format:
Matt Pharr, Brent Burley, Per Christensen, Marcos Fajardo, Luca Fascione, and Christopher Kulla. 2018. Design and Implementation of Modern Production Renderers. In *Proceedings of SIGGRAPH '18 Panels.* ACM, New York, NY, USA, Article 4, 2 pages. https://doi.org/10.1145/3209621.3214901

1 PANEL OVERVIEW

Over the past decade, production rendering for Computer Generated Animation and Visual Effects has seen a major transition, moving to physically-based approaches with path tracing [Kajiya 1986] at their foundation. While before many film studios used Pixar's *Photorealistic RenderMan* renderer, today there are a wide variety of production path tracers in use, some developed in-house and others commercial products.

The Reyes algorithm that RenderMan was based on was described in a classic 1987 SIGGRAPH paper by Cook et al. [1987] that laid out the challenges of production rendering then and described how Reyes's micropolygon rasterization approach was a good solution for them. Until recently, however, there haven't been equivalents for modern production renderers: comprehensive systems papers that explained their foundations, goals, and design trade-offs.

This has recently changed with a special issue of *ACM Transactions on Graphics*: the developers of five of the most widely-used production renderers have each written comprehensive systems papers, describing their respective rendering systems, the problems they were designed to address, how they addressed them, and the new techniques they have invented to make path tracing effective for modern production rendering. The renderers described are:

- Arnold (Solid Angle), described by Georgiev et al. [2018]: a commercial renderer used on hundreds of feature films, and numerous commercials and TV series since 2001.
- Arnold (Sony Pictures Imageworks), described by Kulla et al. [2018]: Sony's proprietary version of Arnold, used on all of Sony's movies since 2008 after early success on the film *Monster House* (2006).
- Hyperion (Walt Disney Animation Studios), described by Burley et al. [2018]: used on all of the studio's animated movies starting with *Big Hero 6* (2014).
- Manuka (Weta Digital), described by Fascione et al. [2018]: used on all of Weta's films starting with *Dawn of the Planet of the Apes* (2014).
- RenderMan[1] (Pixar Animation Studios), described by Christensen et al. [Christensen et al. 2018]: both a commercial product, used on over 45 feature films, and the renderer used by Pixar for all of its movies since *Finding Dory* (2016).

These renderers are built on a few common foundations: all use path tracing and employ bounding volume hierarchies for ray intersection acceleration. All support a range of geometric primitives, including subdivision surfaces, hair, and scattering volumes, and all run on CPUs and are multi-threaded, with varying levels of adoption of SIMD. The challenges of GPU rendering (primarily,

[1]An all new path-tracer that has kept Pixar's Reyes renderer's name.

SIGGRAPH '18 Panels, August 12-16, 2018, Vancouver, BC, Canada

ACM ISBN 978-1-4503-5812-5/18/08.
https://doi.org/10.1145/3209621.3214901

limited local memory on GPUs) have so far precluded any of these systems also using GPUs, though work in this area continues, given the computational capabilities they offer.

Beyond that core, there's a remarkable amount of variety among them; different system designers have made different implementation decisions in a number of fundamental areas. Examples include:

- The variety of light transport algorithms supported, ranging from just path tracing (the Solid Angle version of Arnold), to path tracing with selective photon mapping (Hyperion), to a wide variety of global illumination algorithms (Manuka, RenderMan, Sony's Arnold).
- When texture mapping and shading occurs: in Manuka, all of it is done before rendering begins, allowing efficient BRDF generation at intersection points during rendering, with no need for a texture cache at that point; other systems do this the conventional way—at ray intersection time.
- Whether the renderer is sold commercially (Solid Angle Arnold and RenderMan): commercial products must serve a wider variety of workflows than in-house renderers and must include features that let the end-user customize their behavior. For an in-house renderer, therefore simpler software architectures can be possible.
- Specialization to studio workflows and needs: Hyperion's design was strongly influenced by the need for artistic control, while Manuka places particular emphasis on grounding rendering in measured data for predictably matching on-set measurements.
- Whether "out of core" rendering is supported: Hyperion batches and sorts rays in order to access geometry and texture more coherently, while other systems do not, accepting incoherent memory access from path tracing.

This panel will bring together the authors of those papers for a wide-ranging discussion of the design and implementation of these systems, with a particular focus on their points of difference.

2 SPEAKER BIOS

Matt Pharr was the guest editor for this special issue of *TOG*. He is a software engineer in the Google Brain team, where he works on applications of rendering to Machine Learning and applications of Machine Learning to rendering. He is one of the authors of the book *Physically Based Rendering*, for which he was awarded a Scientific and Technical Academy Award for the book's influence on rendering in film. He has a Ph.D. from the Stanford Computer Graphics lab and previously worked at Intel, Neoptica, NVIDIA, Exluna, and Pixar.

Brent Burley is a Principal Software Engineer at Walt Disney Animation Studios leading the Hyperion development team. Previously he led the development of the physically-based shading model used in all WDAS productions since *Wreck-It Ralph*, and created Ptex, an open-source texture mapping system for subdivision surfaces used on all WDAS productions since Bolt. Prior to joining Disney in 1996, he worked at Philips Media developing a cross-platform game engine, and also worked on aircraft training simulators at Hughes Training Inc.

Per Christensen is a principal software developer in Pixar's RenderMan group in Seattle. He received an M.Sc. degree in electrical engineering from the Technical University of Denmark and a Ph.D. in computer science from the University of Washington in Seattle. Before joining Pixar, he worked at Mental Images in Berlin and at Square USA in Honolulu. His movie credits include every Pixar movie since *Finding Nemo*, and he has received an Academy Award for his contributions to the development of point-based global illumination and ambient occlusion.

Marcos Fajardo is the founder of Solid Angle, where he leads the research and development team working on the Arnold renderer. Previously he was a visiting software architect at Sony Pictures Imageworks, a visiting researcher at USC Institute for Creative Technologies, and a consultant at various CG studios around the world. He studied Computer Science at University of Malaga, Spain. He is a frequent speaker at SIGGRAPH, FMX and EGSR and recently received an Academy Award for the design and implementation of the Arnold renderer. His favorite sushi is engawa.

Luca Fascione is Senior Head of Technology & Research at Weta Digital, where he oversees Weta's core R&D efforts including Simulation and Rendering Research, Software Engineering and Production Engineering. He is the lead architect of Weta Digital's next-generation renderer, Manuka, which is the culmination of a three-year research endeavour involving over 40 researchers. Luca joined Weta Digital in 2004 and has also worked for Pixar Animation Studios. Luca was recently recognized with a Scientific and Engineering award from the Academy of Motion Pictures for his work on FACETS, Weta's facial motion capture system.

Christopher Kulla is a principal software engineer at Sony Pictures Imageworks where he has worked on the in-house branch of the Arnold renderer since 2007. He focuses on ray tracing kernels, sampling techniques and volume rendering. In 2017 he was recognized with a Scientific and Engineering Award from the Academy of Motion Picture Arts and Sciences for his work on Arnold.

REFERENCES

Brent Burley, David Adler, Matt Jen-Yuan Chiang, Hank Driskill, Ralf Habel, Patrick Kelly, Peter Kutz, Yining Karl Li, and Daniel Teece. 2018. The Design and Evolution of Disney's Hyperion Renderer. *ACM Transactions on Graphics* 37, 3 (May 2018).

Per Christensen, Julian Fong, Jonathan Shade, Wayne Wooten, Brenden Schubert, Andrew Kensler, Stephen Friedman, Charlie Kilpatrick, Cliff Ramshaw, Marc Bannister, Brenton Rayner, Jonathan Brouillat, and Max Liani. 2018. RenderMan: An Advanced Path Tracing Architecture for Movie Rendering. *ACM Transactions on Graphics* 37, 3 (May 2018).

Robert L. Cook, Loren Carpenter, and Edwin Catmull. 1987. The Reyes image rendering architecture. *Computer Graphics (Proceedings of SIGGRAPH)* (Aug. 1987), 95–102. https://doi.org/10.1145/37402.37414

Luca Fascione, Johannes Hanika, Mark Leone, Marc Droske, Jorge Schwarzhaupt, Tomas Davidovic, Andrea Weidlich, and Johannes Meng. 2018. Manuka: A batch-shading architecture for spectral path tracing in movie production. *ACM Transactions on Graphics* 37, 3 (May 2018).

Iliyan Georgiev, Thiago Ize, Mike Farnsworth, Ramón Montoya-Vozmediano, Alan King, Brecht Van Lommel, Angel Jimenez, Oscar Anson, Shinji Ogaki, Eric Johnston, Adrien Herubel, Declan Russell, Frederic Servant, and Marcos Fajardo. 2018. Arnold: A Brute-force Production Path Tracer. *ACM Transactions on Graphics* 37, 3 (May 2018).

James T. Kajiya. 1986. The rendering equation. *Computer Graphics (Proceedings of SIGGRAPH)* (1986), 143–150. https://doi.org/10.1145/15922.15902

Christopher Kulla, Alejandro Conty, Clifford Stein, and Larry Gritz. 2018. Sony Pictures Imageworks Arnold. *ACM Transactions on Graphics* 37, 3 (May 2018).