



*Iray in
Omniverse*
New Features for
Light Transport
Simulation and
Rendering (S31739)

Carsten Wächter

April, 2021



What is *Iray*?

Production Rendering on *CUDA*

Bring ray & path tracing based production / simulation quality rendering to GPUs

New paradigm: *Push Button* rendering (open up new markets)

Without sacrificing artistic freedom: Full *Material Definition Language (MDL)* support

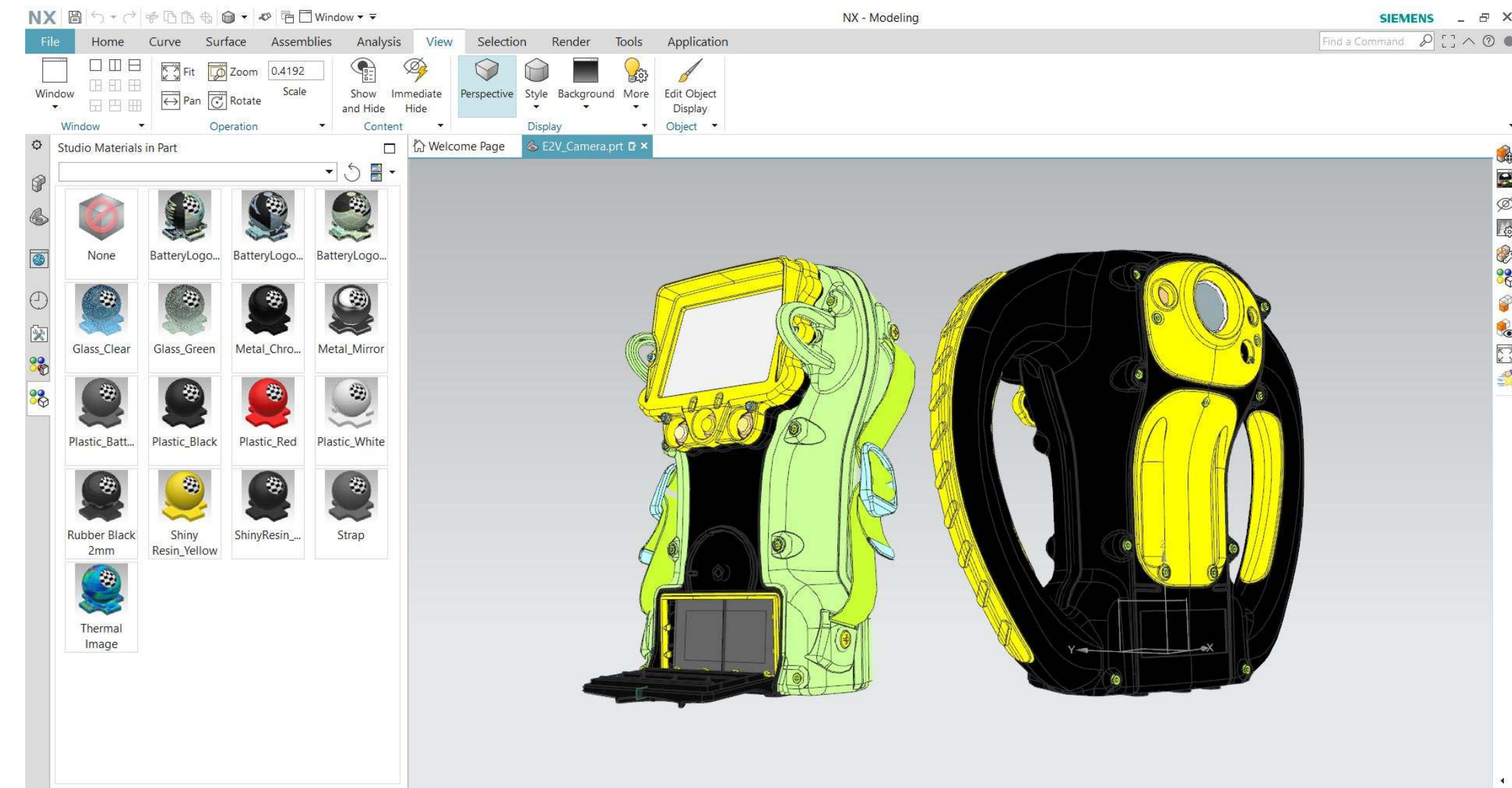
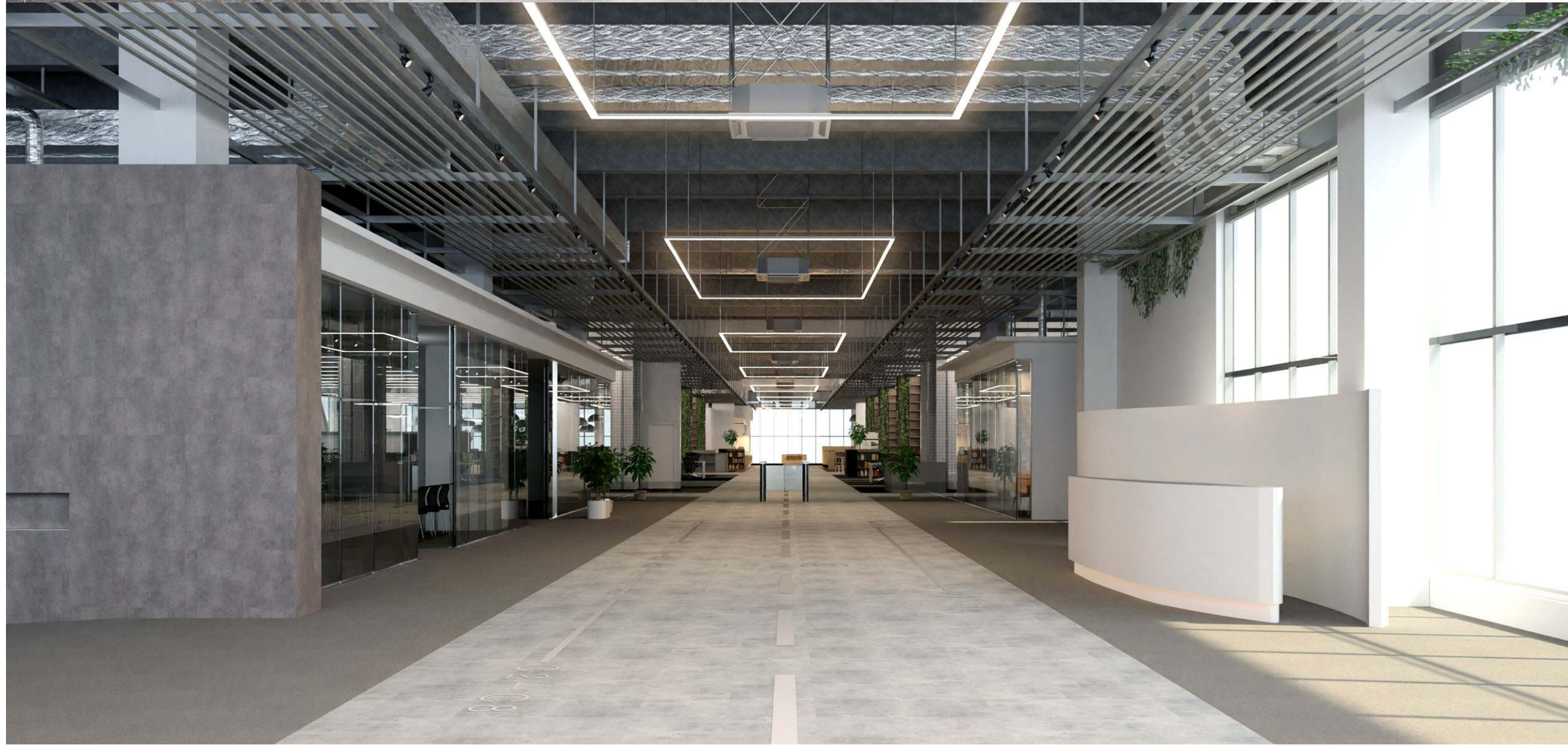
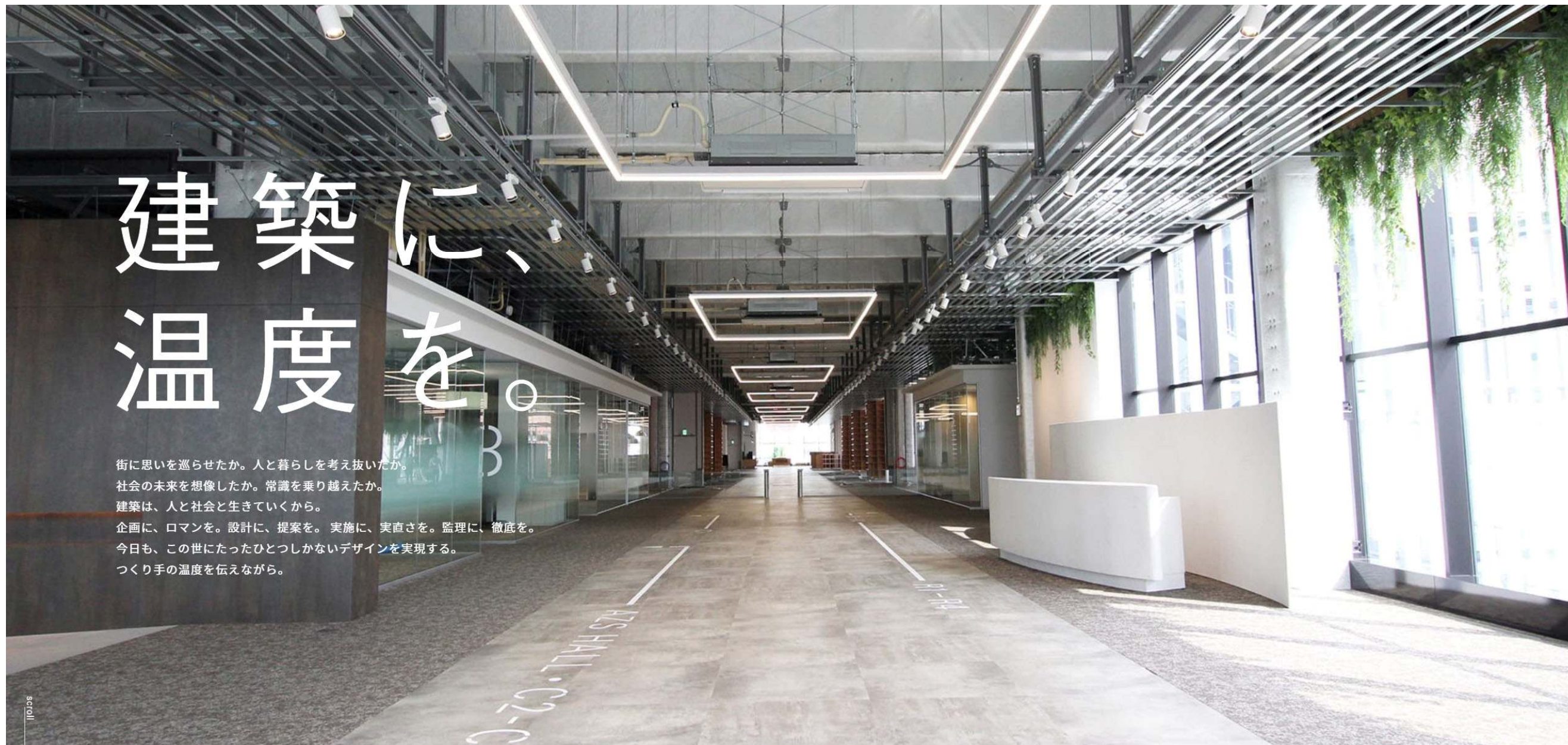
Plugins for



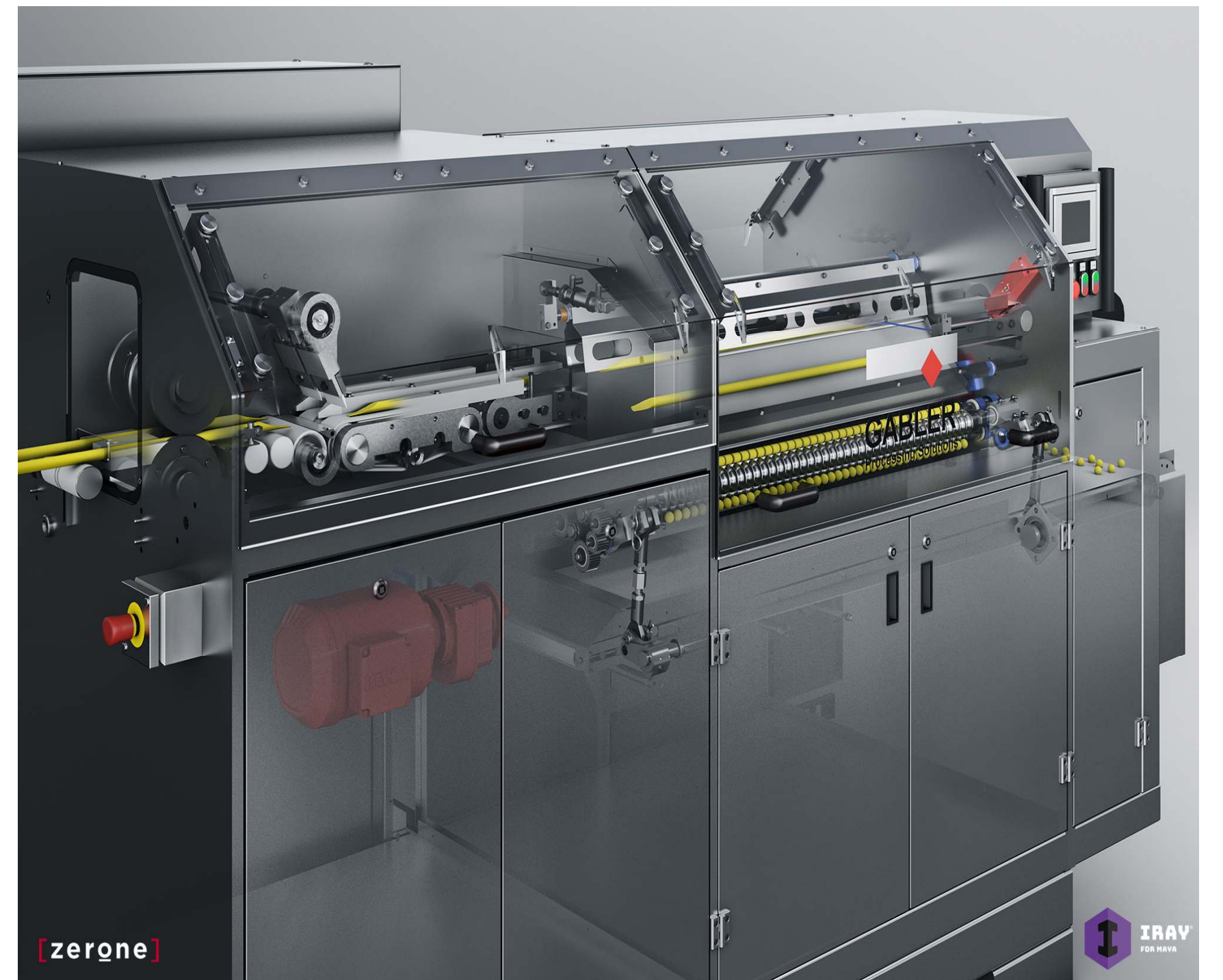
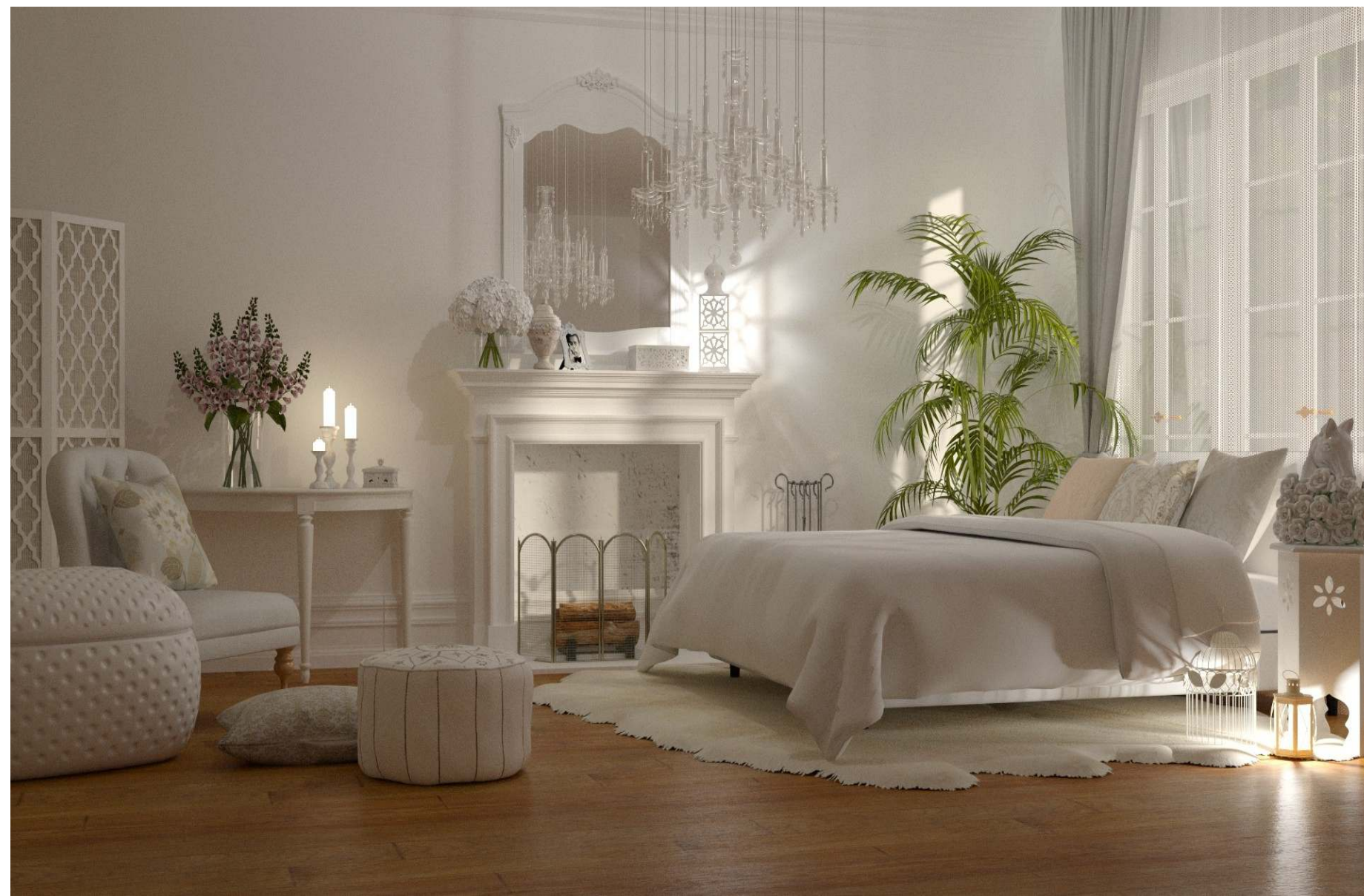
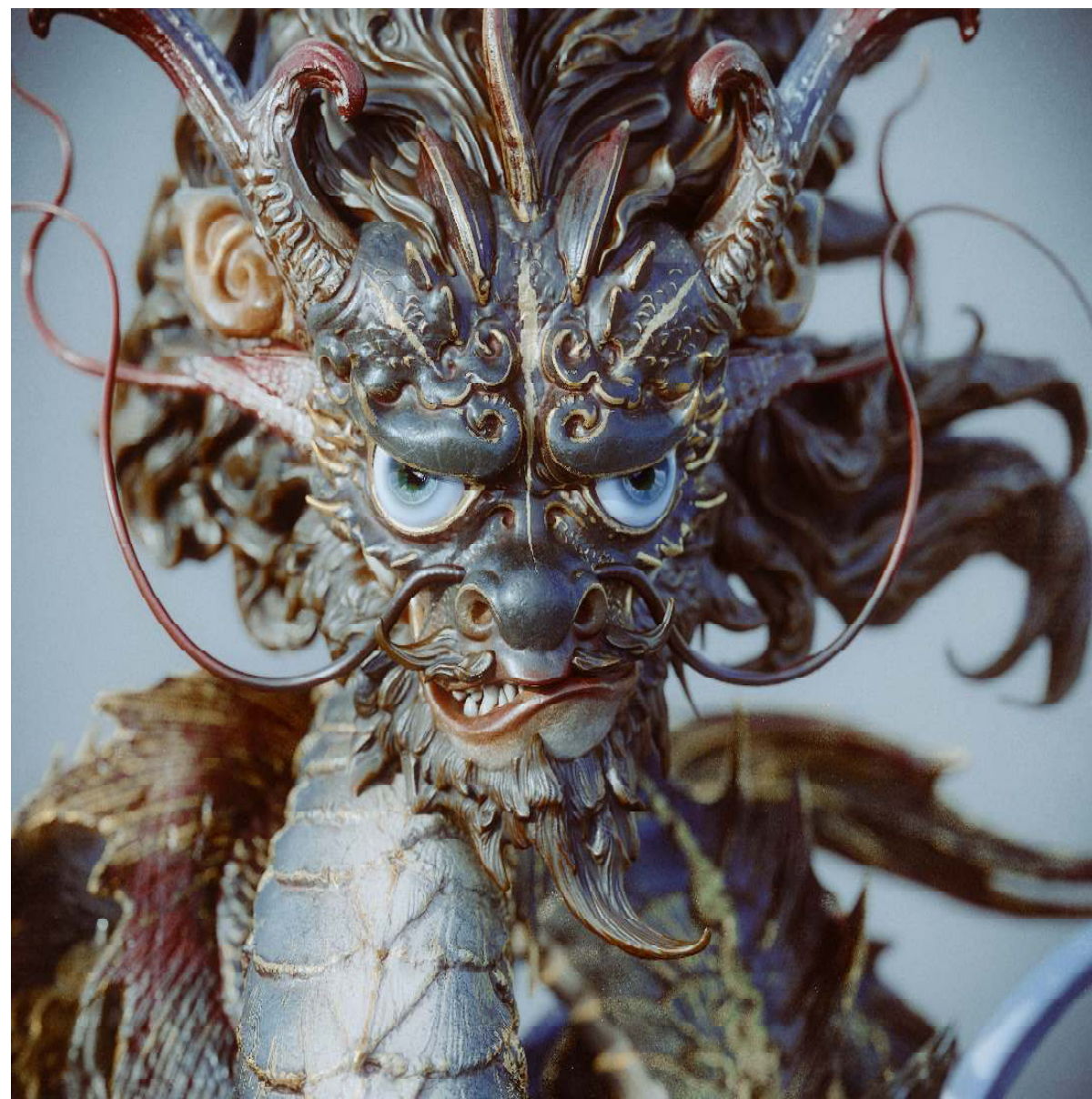
In Production for > 12 Years

> 5M active users install base with even more active users in the cloud





Images courtesy of Luminova Japan / Siemens Digital Industries Software (NX Ray Traced Studio)



Images courtesy of Zhelongxu (zhelongxu.com) / Floorplanner / [0x1] (Iray for Maya) / Siemens Digital Industries Software (NX Ray Traced Studio) / Luminova Japan

NVIDIA Ampere

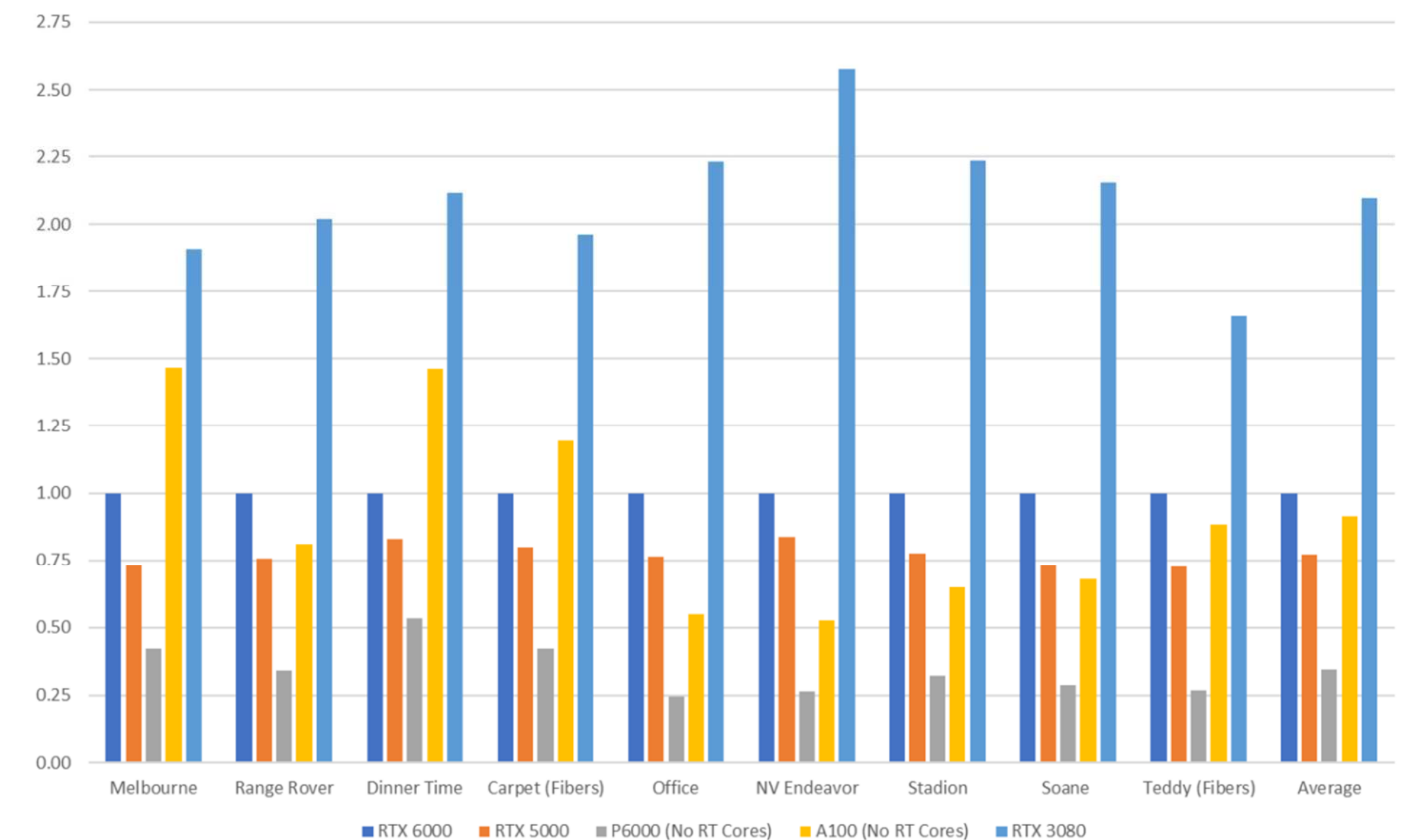
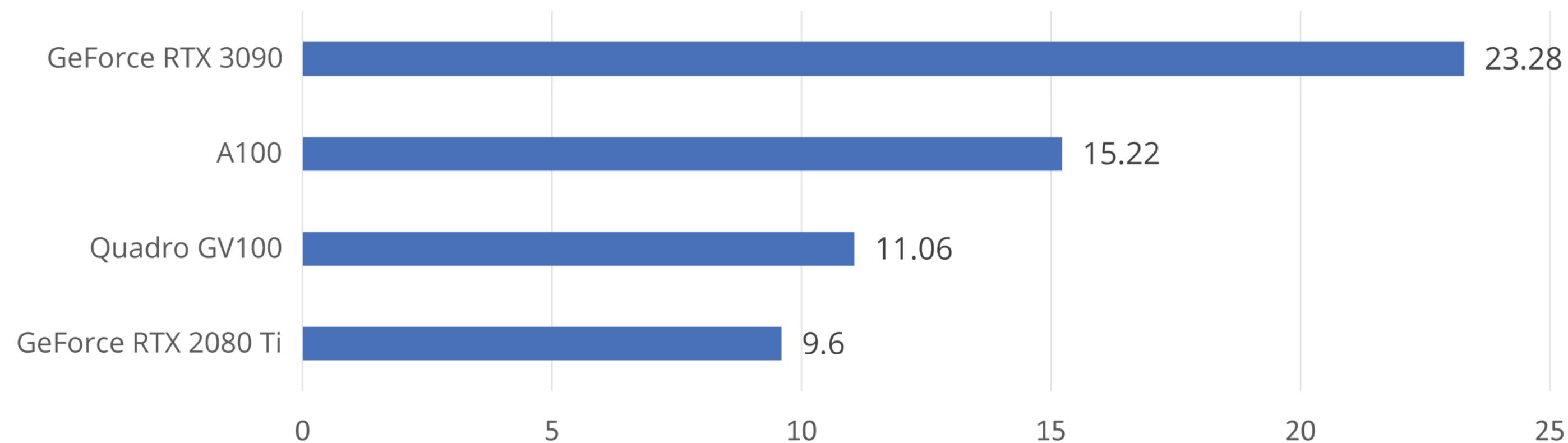
Native support for SM 8.x GPUs

PTC benchmark

“.. Ampere is giving around a 1.4x speedup over Volta when comparing similar cards (the Tesla V100 and the NVIDIA A100) and an even larger speedup of around 2.4x over Turing with a similar generation card (e.g. the GeForce RTX 2080 Ti compared to the GeForce RTX 3090).”

NVIDIA benchmark

Comparing RTX 6000 to the RTX 3080: 1.7x up to 2.6x speedup! (average 2.1x)

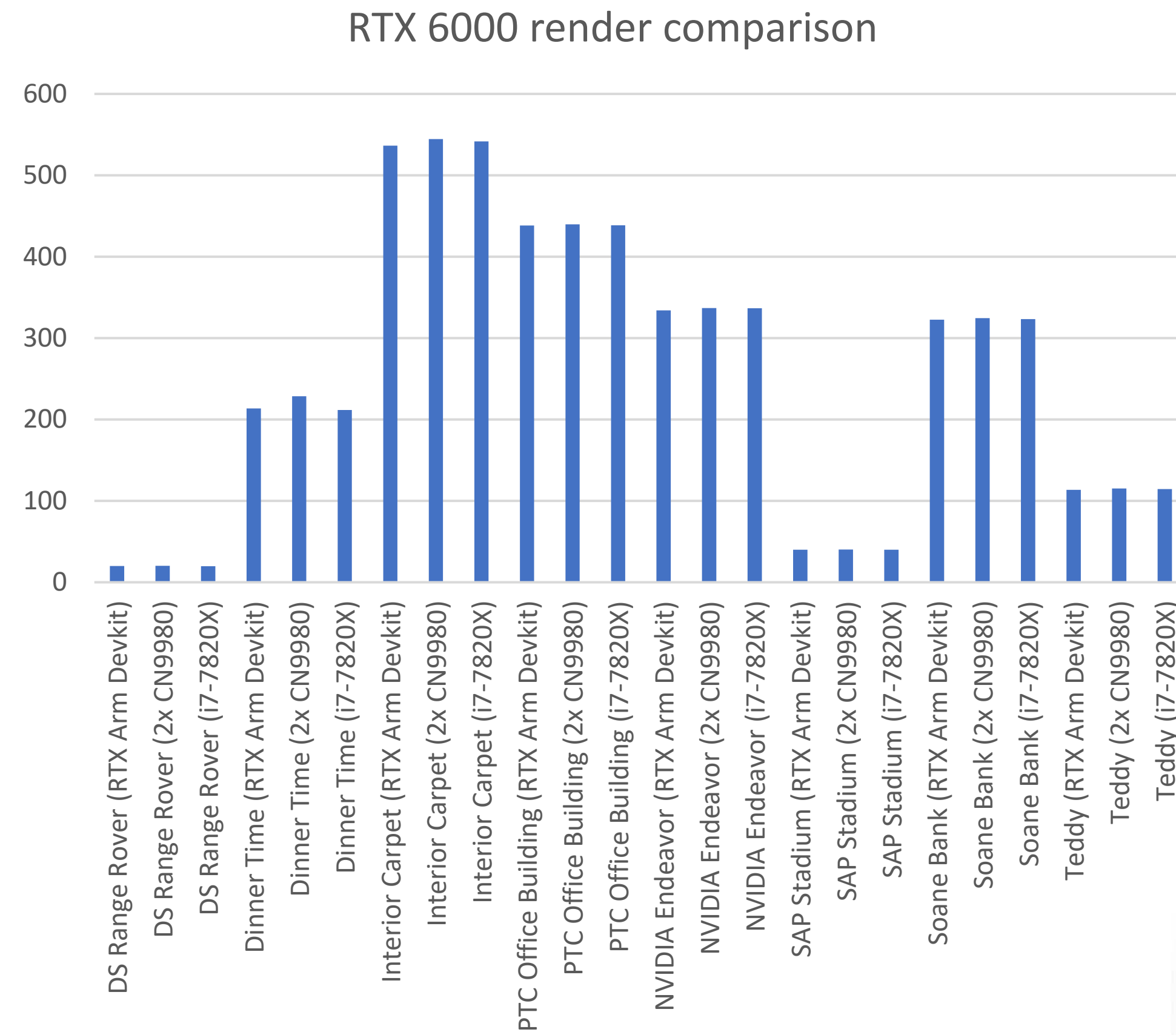


Arm

Native support incl. *NVIDIA RTX Arm PC Developer Kit*

CUDA, OptiX, IndeX just work

Also see <https://developer.nvidia.com/arm>



All tested configs:

RTX Arm PC Developer Kit

2x CN9980 Arm Server

i7-7820X x86 Workstation

Solely performance limited
by the used *RTX 6000* GPU



Images courtesy of Floorplanner

CGI INTERIORS

Render time in seconds (lower is better)

Performance Improvements

RTX accelerated interactive rendering accelerates even more

Improved interactivity at the cost of slight *temporary* bias

Turntable (and similar) scenes: ~5% faster

Wavefront-rendering-architecture-unfriendly scenes: up to ~50% faster

In addition, many many improvements all over the place, e.g.

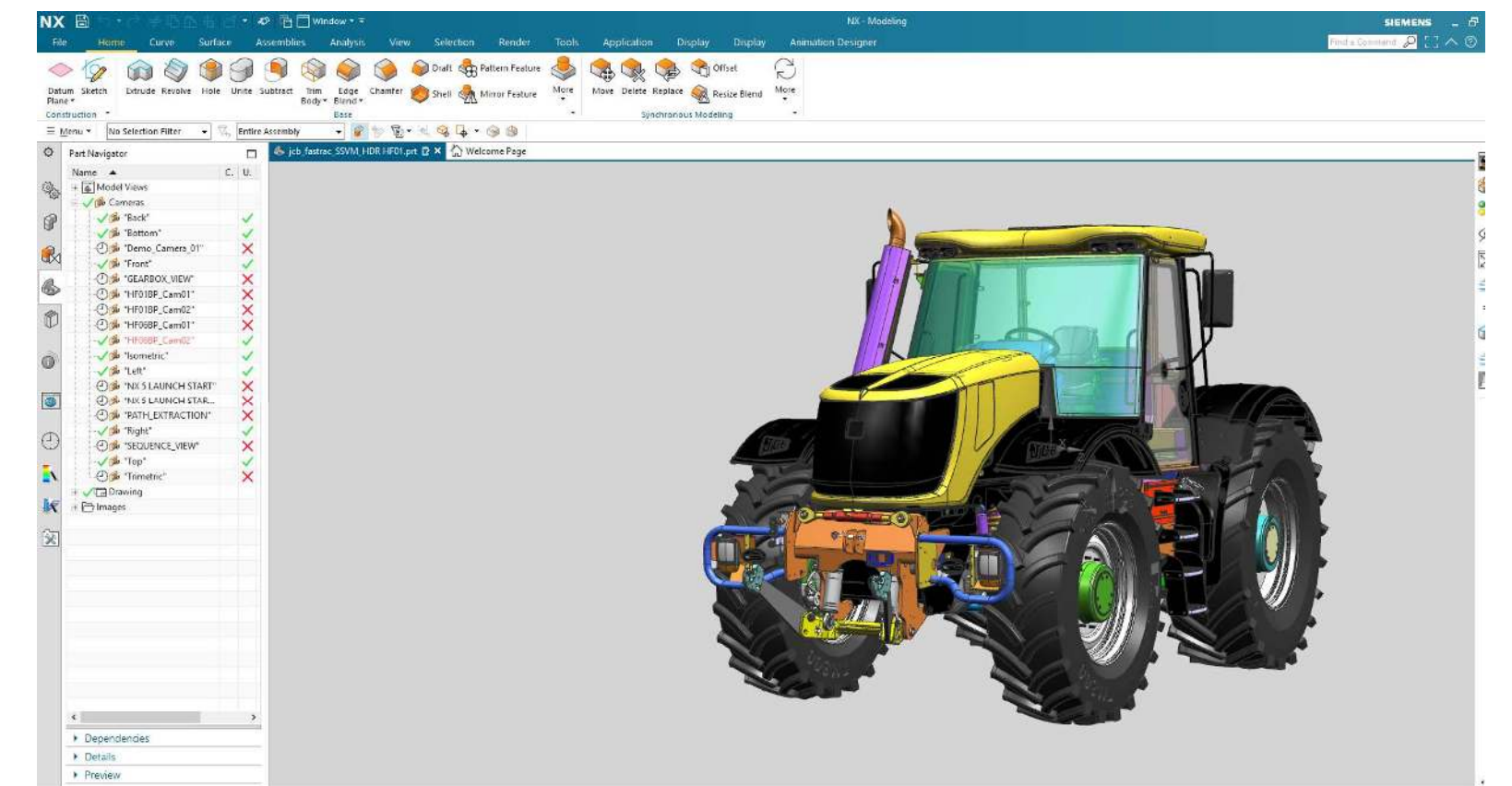
Preprocessing time of scenes with a large amount of instanced geometry or lots of different materials

Instance transformation changes especially in combination with many materials

Rendering performance of JIT compiled *MDL* materials

Full *CUDA* and *OpenGL* canvas support, incl. accelerated color conversions

Interacting with IBL/environment maps and procedurals





Images courtesy of PTC



Improved Fiber support

New *OptiX 7.3 RTX* accelerated curve intersector

Improved Performance, comparing overall Rendertime

Experimental, i.e. not productized yet performance in ()
(hopefully gained via upcoming driver updates)

Also see *OptiX Advanced Topics [S31752]* & *What's New in OptiX [S31736]*

Also upcoming: Native support in Omniverse

+8% (+85%)



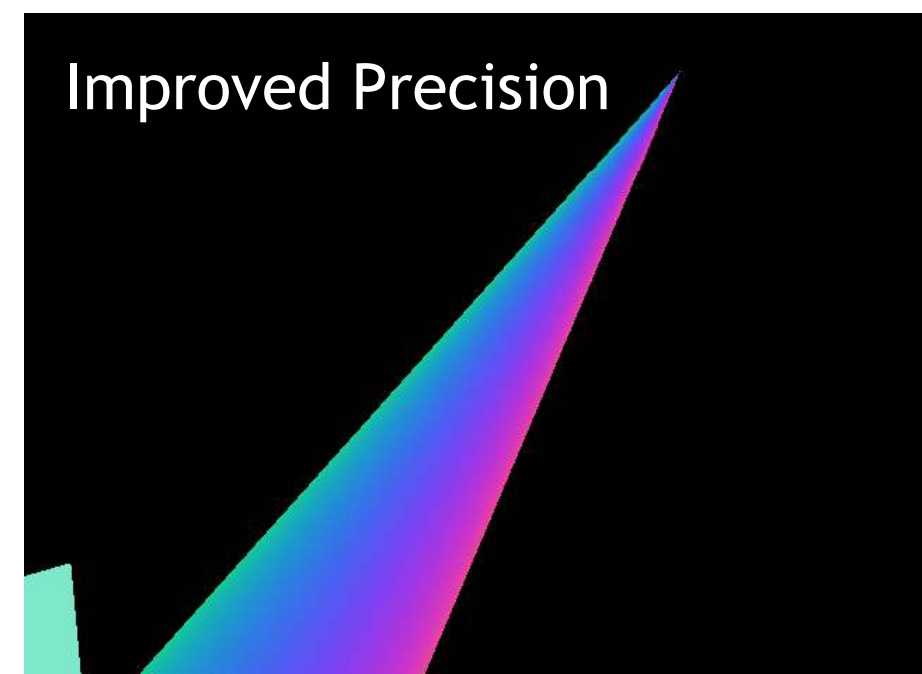
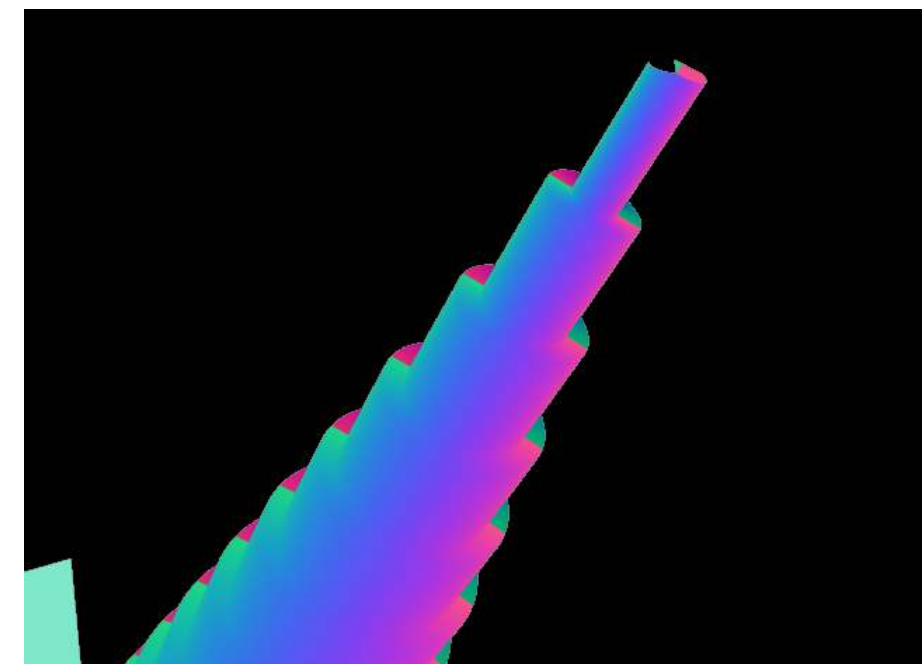
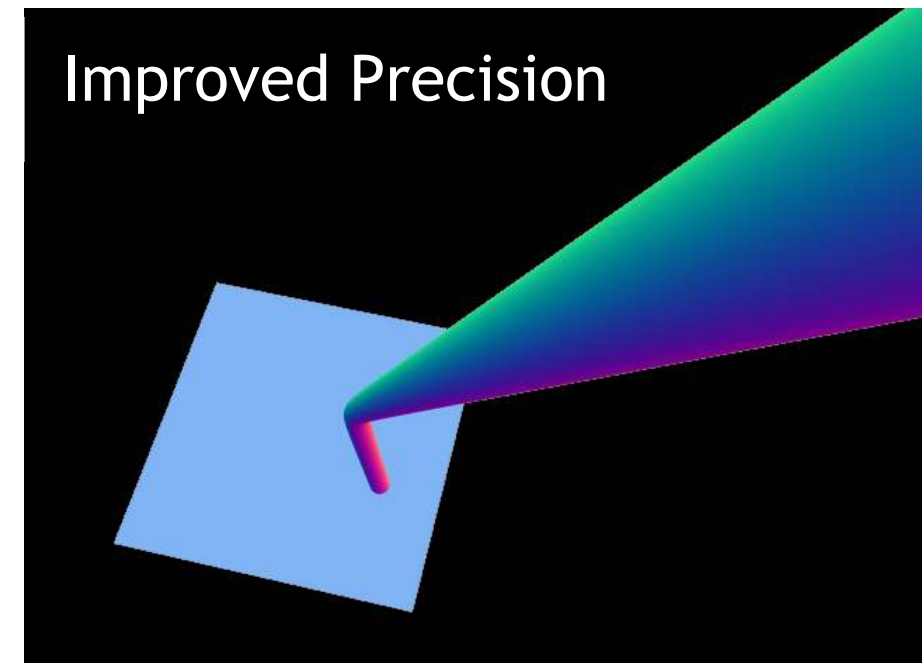
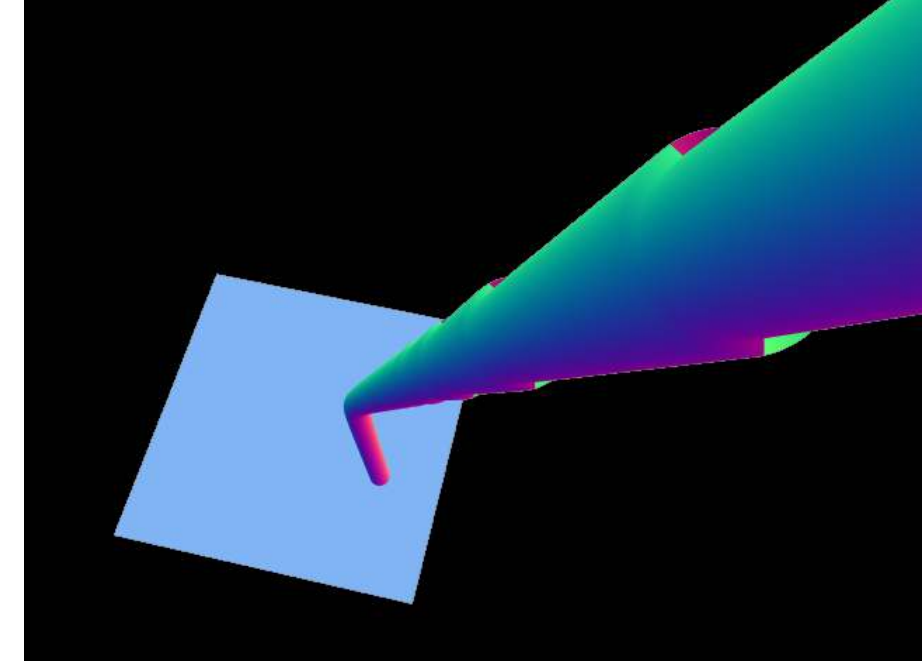
+20% (+45%)



+2% (+3%)



Image courtesy of [0x1] (Iray for Maya)



AI Denoiser

Improved performance and quality

AI Denoiser library is now employed via the *OptiX* driver
i.e. reduced size of the *Iray SDK* and always state of the art quality

Due to improvements now able to use the Normal auxiliary/AOV buffer

Work ongoing to include new *OptiX 7.3* temporal mode
better quality while interacting and when rendering animations

Also see *OptiX Advanced Topics [S31752]* & *What's New in OptiX [S31736]*

No Normals

+Normals

Reference













Optional Volume Priorities

On top of existing automatic Stack handling

Optional `volume_priority` can be set to define which object's volume properties take precedence in case of an overlap

Helps solving the “icecube-problem” and volume hulls with fine scale details

With Priorities



Without Priorities





Heterogeneous Volume Support

via *IndeX Direct*

Includes *OpenVDB* & *NanoVDB* support

Support for *Flow* (& other volume/simulation data) within *OV* (WIP)

Strong focus on simulation quality

Features

Heterogeneous (Oil/Water) &
Inhomogeneous (Density variation, e.g. Clouds) volumes supported

Nested volumes

Additive Mix of volumes

Mesh Clipping Support (both ways)

Limitations

No support for giant datasets (yet)

No emission (yet)



Heterogeneous Volume Support

via *IndeX Direct*

Includes *OpenVDB* & *NanoVDB* support

Support for *Flow* (& other volume/simulation data) within *OV* (WIP)

Strong focus on simulation quality

Features

Heterogeneous (Oil/Water) &
Inhomogeneous (Density variation, e.g. Clouds) volumes supported

Nested volumes

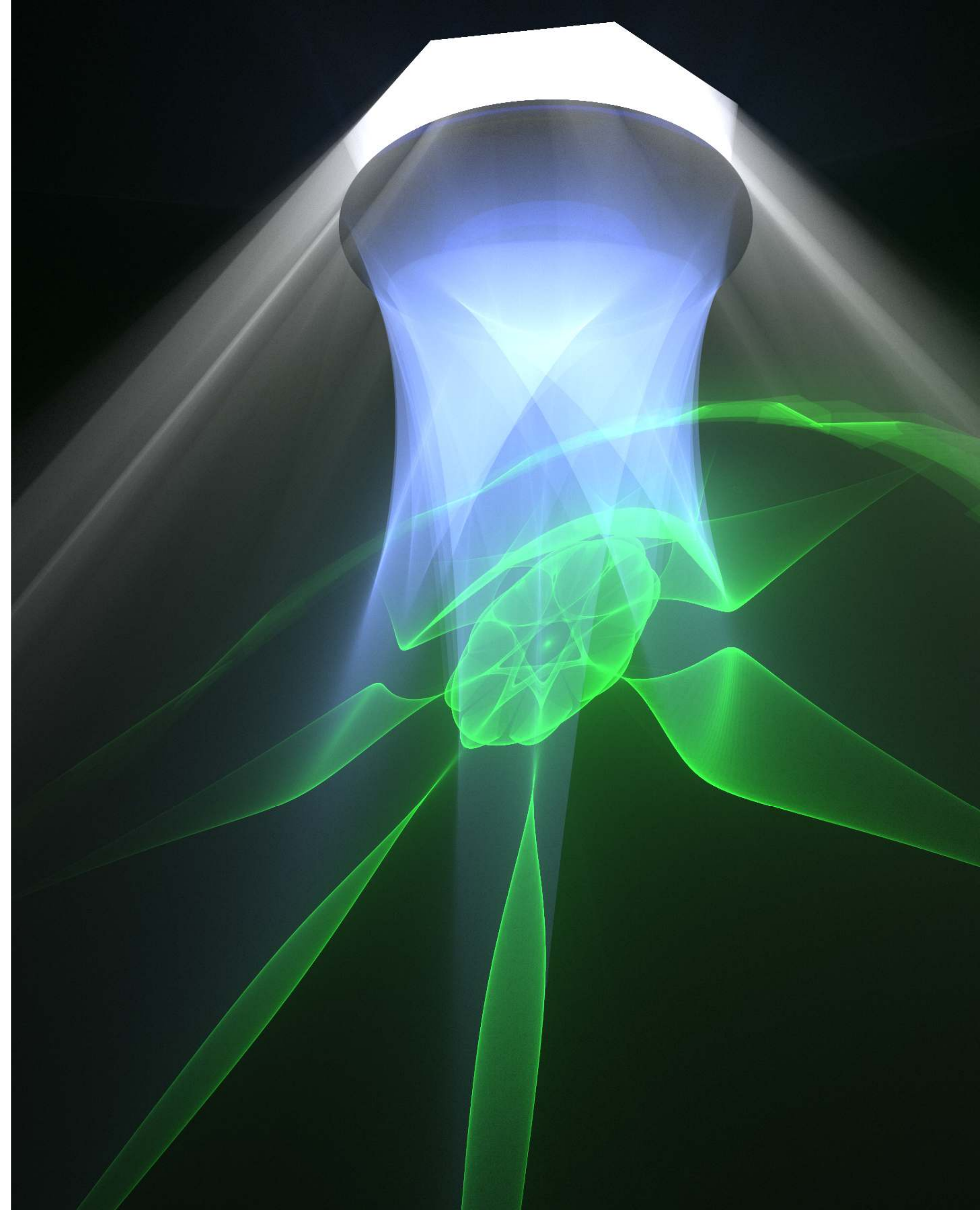
Additive Mix of volumes

Mesh Clipping Support (both ways)

Limitations

No support for giant datasets (yet)

No emission (yet)



Prominent *MDL 1.7* Features

Improved volume support

- *OpenVDB* support as 3D texture
- Add a tint modifier for volumes (effectively changing scattering coefficient)
- Volume mixing



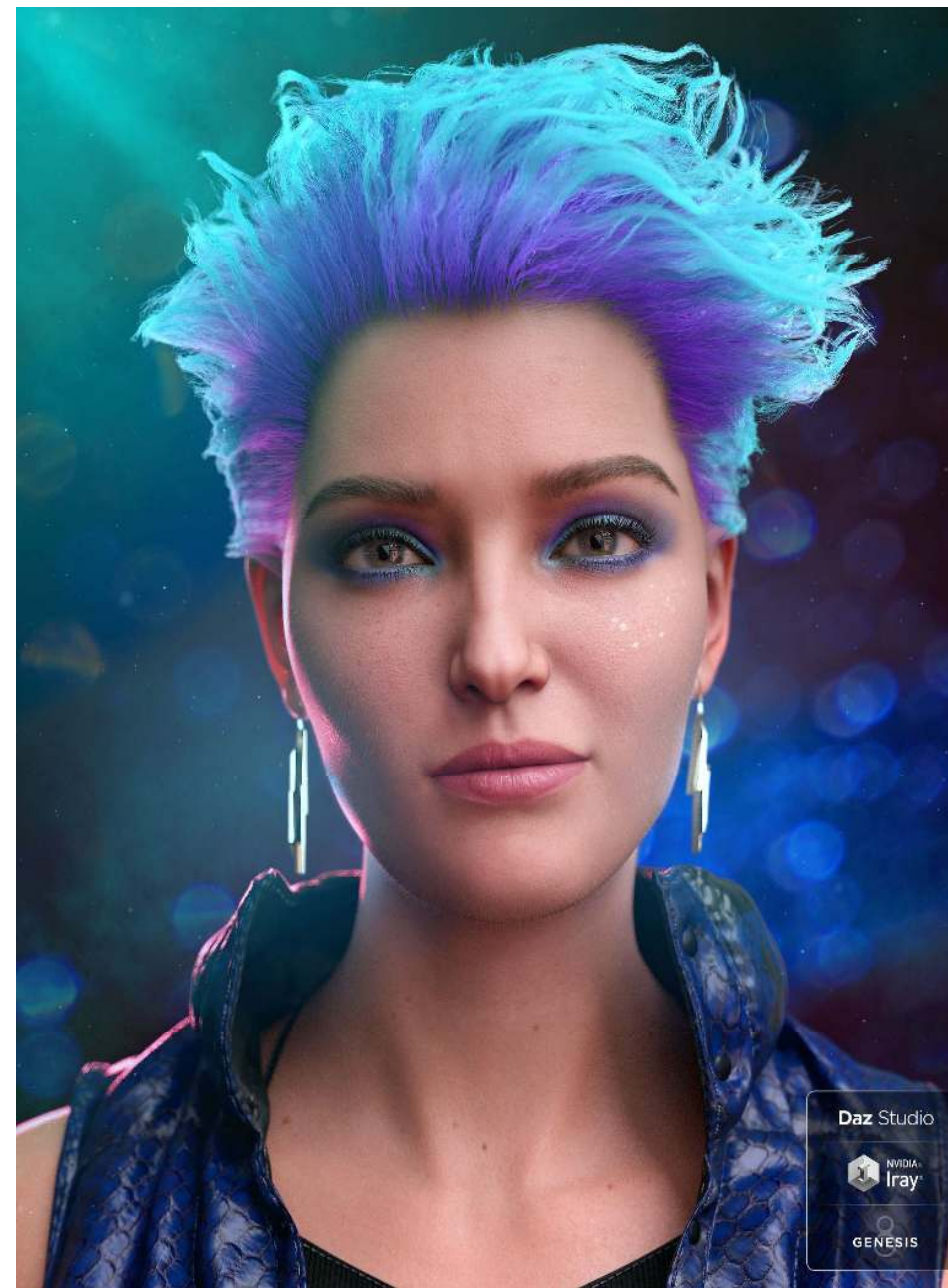
Prominent *MDL 1.7* Features

Backscatter BSDF (Sheen) revised as modifier

Can have a multi-scatter component now for "sheen" coatings on arbitrary base materials

"Revisiting Physically Based Shading at Imageworks" (Christopher Kulla and Alejandro Conty)

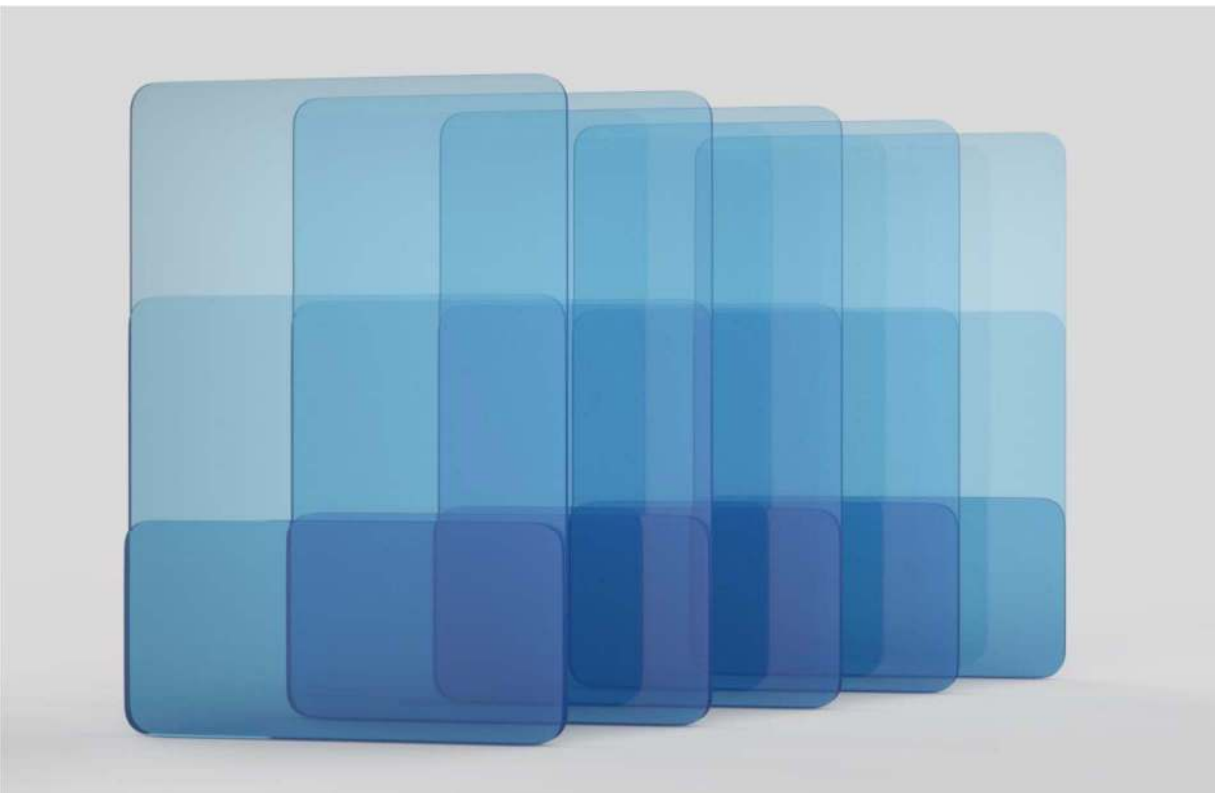




Measured Materials / Spectral Textures

Digital Twin examples scanned with the *X-Rite TAC7*

Blue Semi-Trans. Plastic Step-Chip



Blue Alcantara



Red Fine Grain Leather



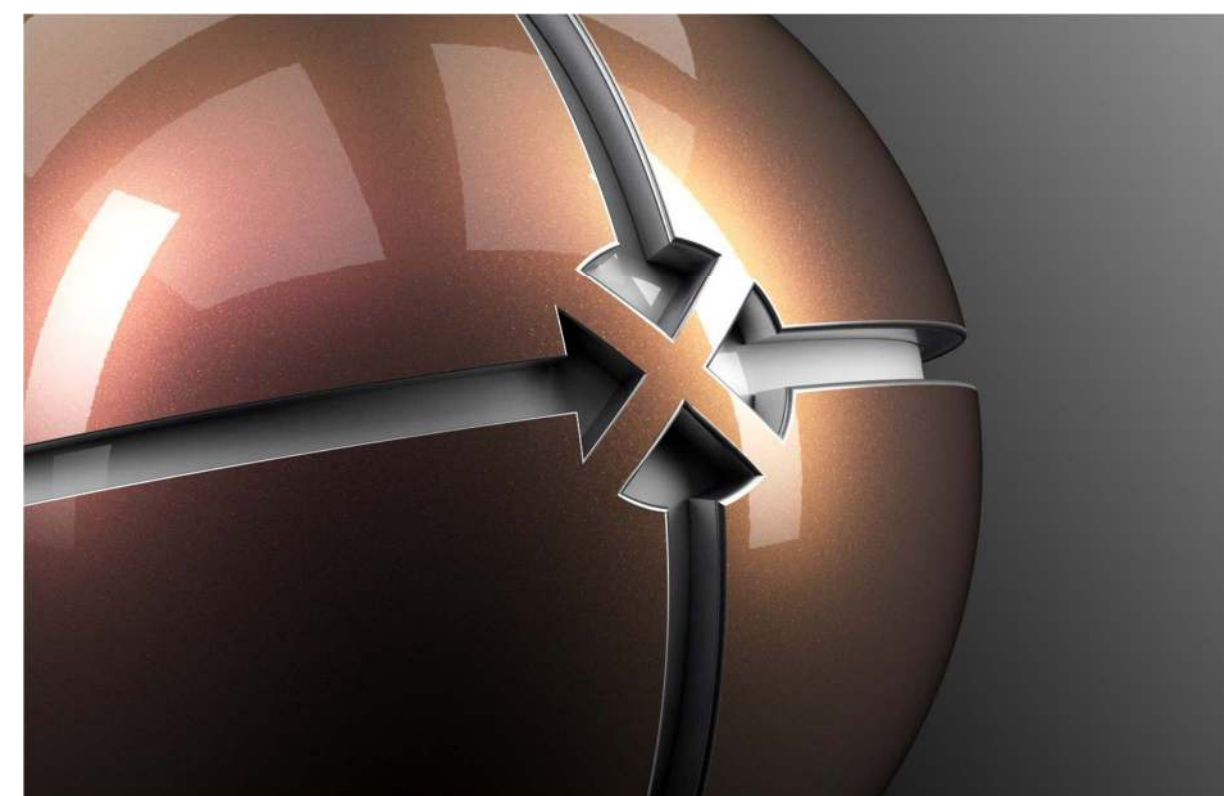
Charcoal Coarse Grain Leather



Dark Green Car Effect Paint



Blue Car Effect Paint



Color Flop Car Effect Paint



Hounds Tooth Textile

Measured Materials / Spectral Textures

Digital Twins

Iray supports both

X-Rite AxF 1.8.1

caddon AIX 1.6.2 spectral textures

New *AxF 1.7* and *1.8.1* feature support

EPSVBRDF representation (Energy Preserving Spatially Varying BRDF)

& refracting clearcoats (via SDK provided conversion to SVBRDF)

SVBRDF representation with color transmission

Spectral color and texture data in SVBRDF, carpaint, and volumetric representations

Measured, but still able to create variants of *AxFs*

Or to combine only certain aspects of different scans

e.g. Surface structure, color, coating layer

On top: Full flexibility of *MDL*



Measured Materials / Spectral Textures

Digital Twins

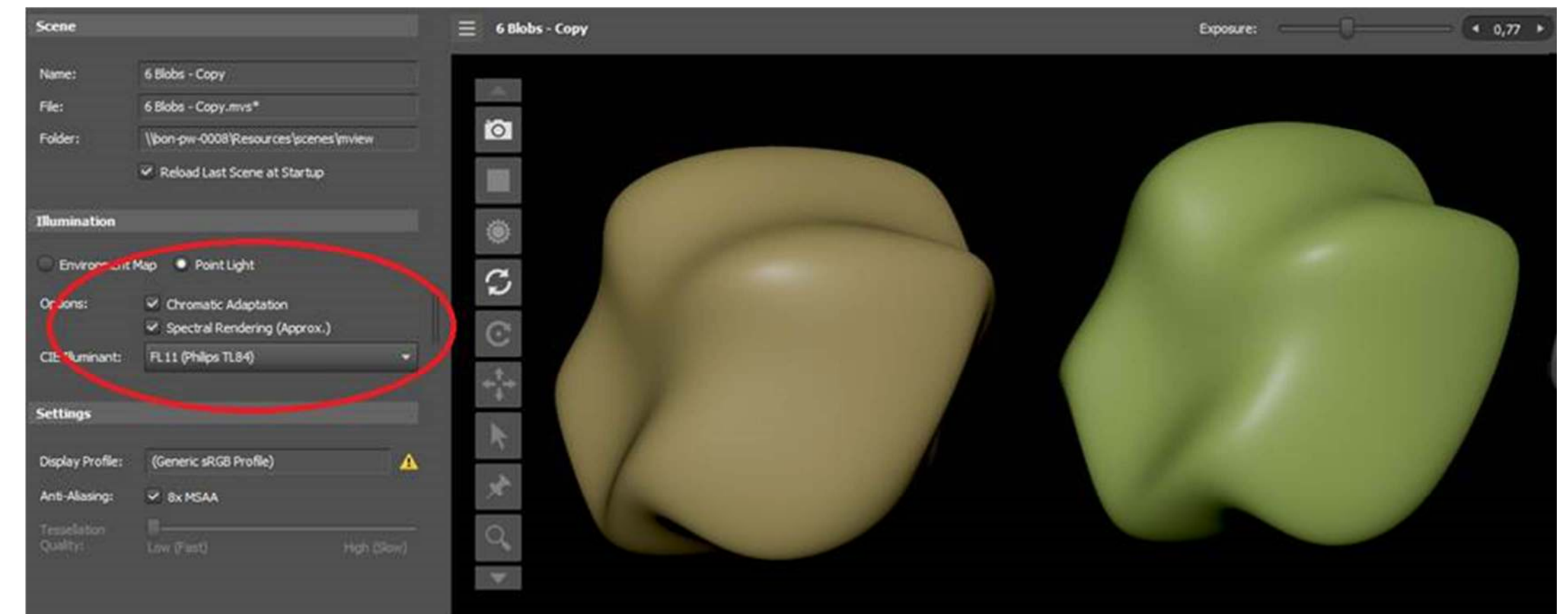
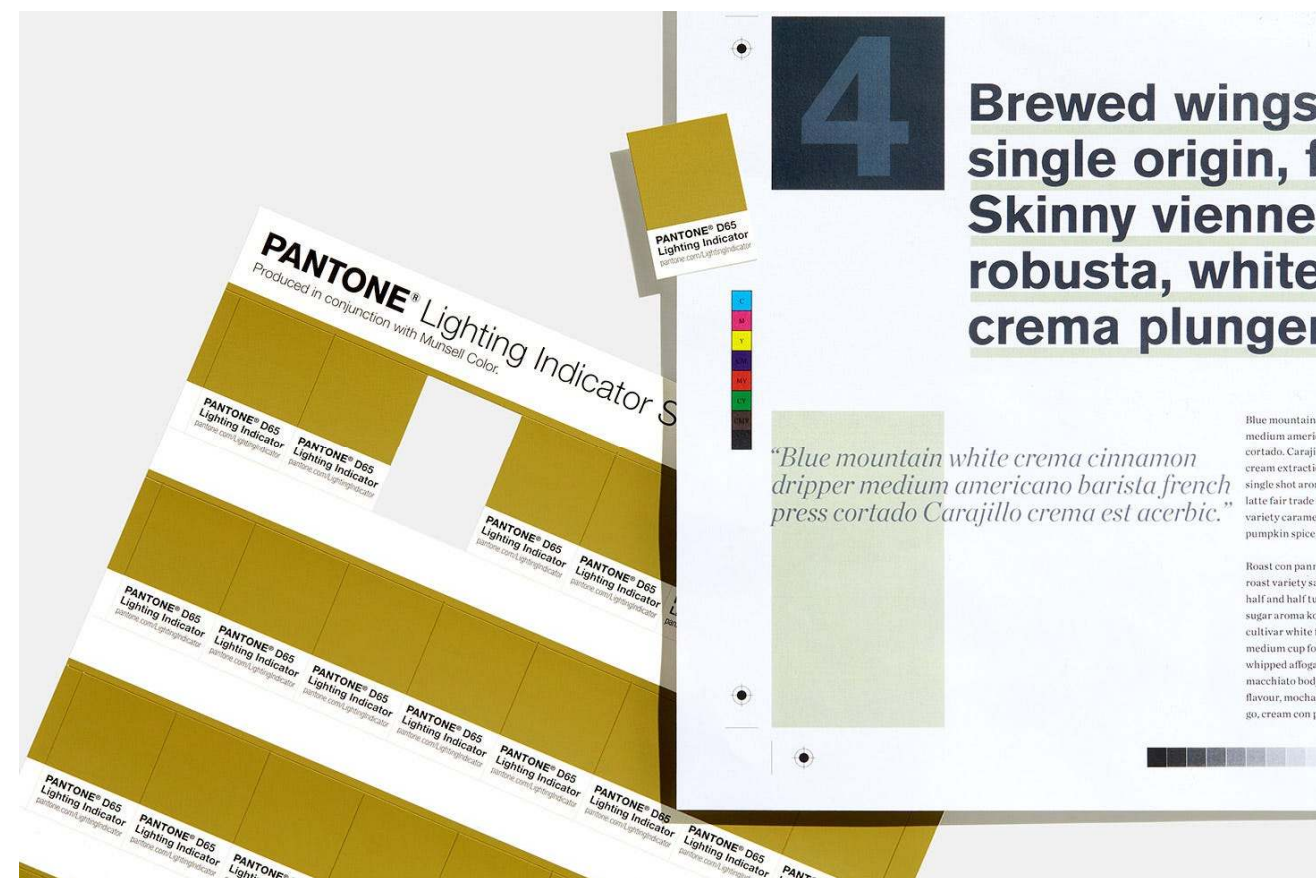
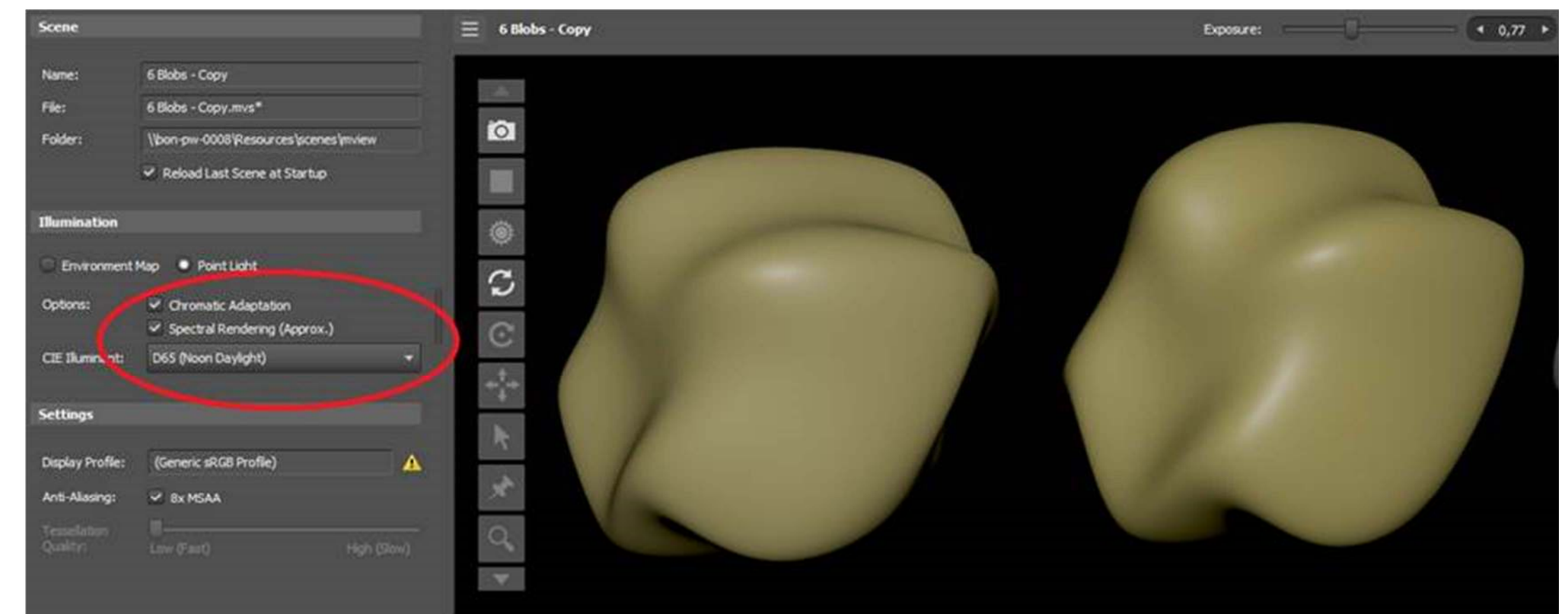
Spectral Rendering extremely important to reliably judge and select both lighting conditions and materials

Example: *Pantone D65 Lighting Indicator Sticker*

2 regions with different colors

Using D65 (6500K daylight) both regions “match”

Other lighting setups will result in visible differences



Iray in NVIDIA Omniverse

Light transport simulation quality in *Create & View*

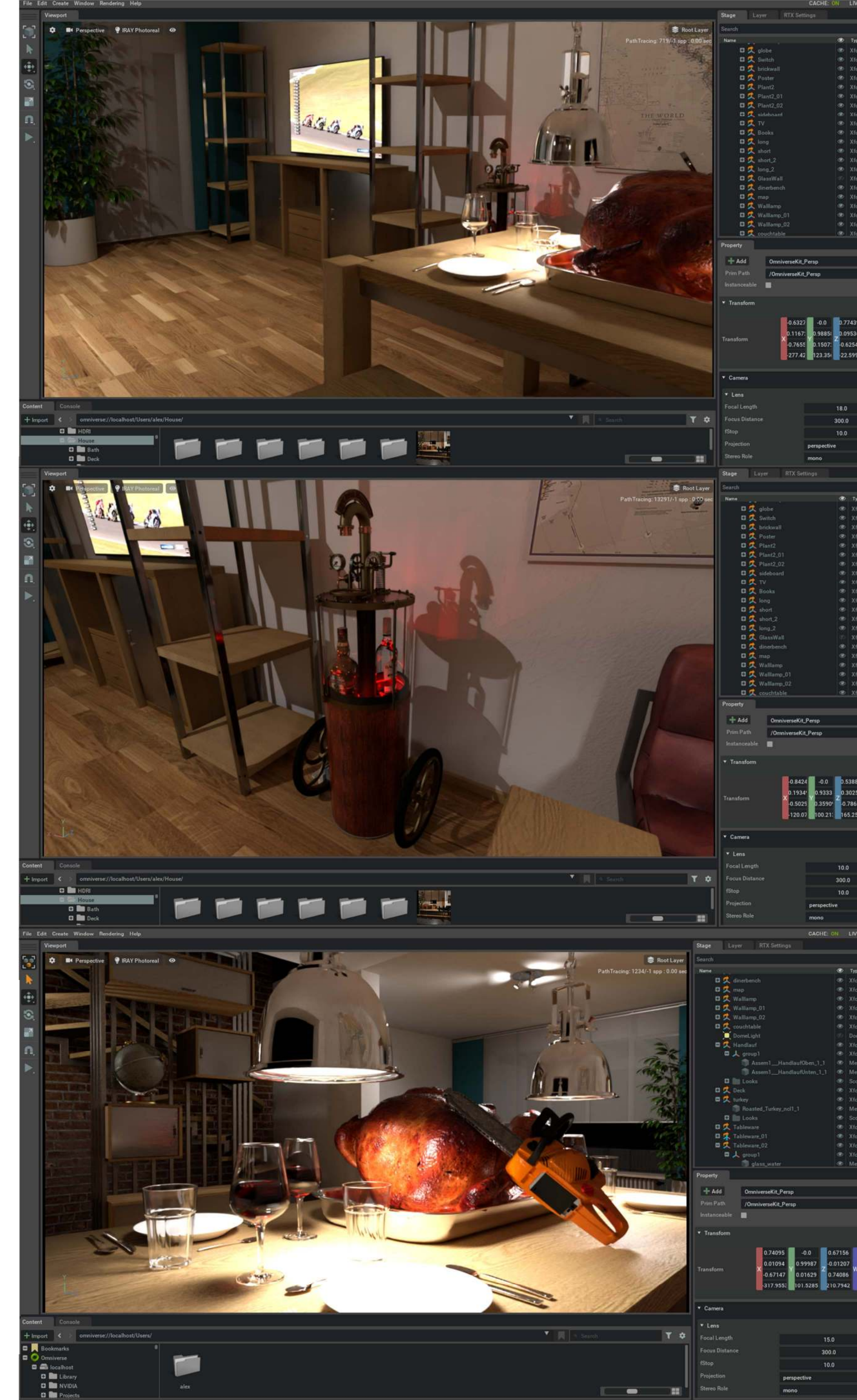
Current *OV Create* release still needs *Iray* to be enabled externally
next major releases of *OV Create* and *View*: on by default
along with vastly improved integration into *OV* (features & performance)

RTX Realtime & RTX Path tracing rendering modes
sacrifice reliability and quality/precision for progressive performance
more (non-physical) knobs to tweak

Iray Photoreal rendering simulation mode
brings trusted & *CIE 171:2006*-verified light transport simulation to *OV*
at the *Push of a Button*



Awards received by  **BLOOM UNIT.**



Iray in NVIDIA Omniverse

Light transport simulation quality in *Create* & *View*

Current *OV Create* release still needs *Iray* to be enabled externally
next major releases of *OV Create* and *View*: on by default
along with vastly improved integration into *OV* (features & performance)

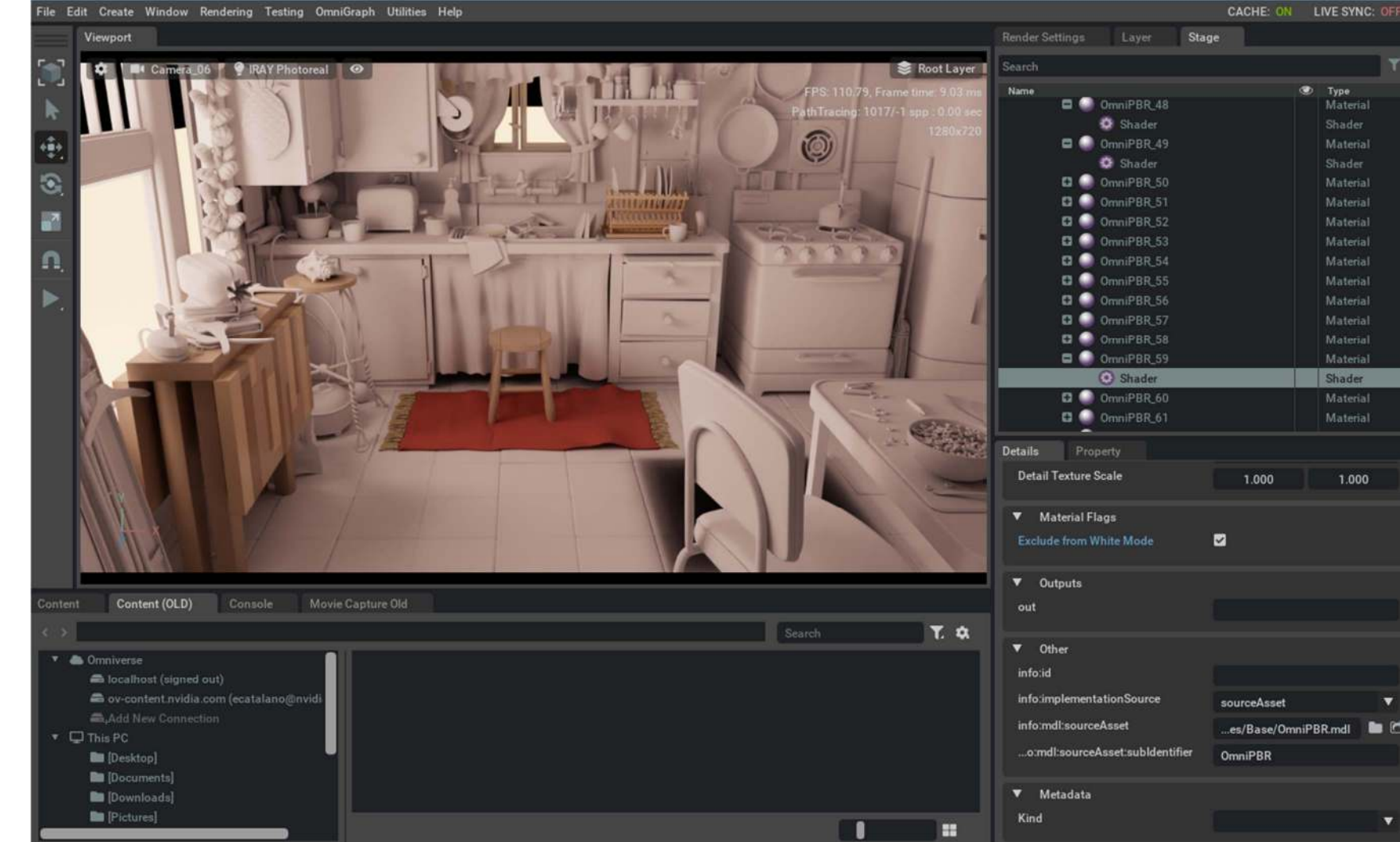
RTX Realtime & *RTX Path tracing* rendering modes
sacrifice reliability and quality/precision for progressive performance
more (non-physical) knobs to tweak

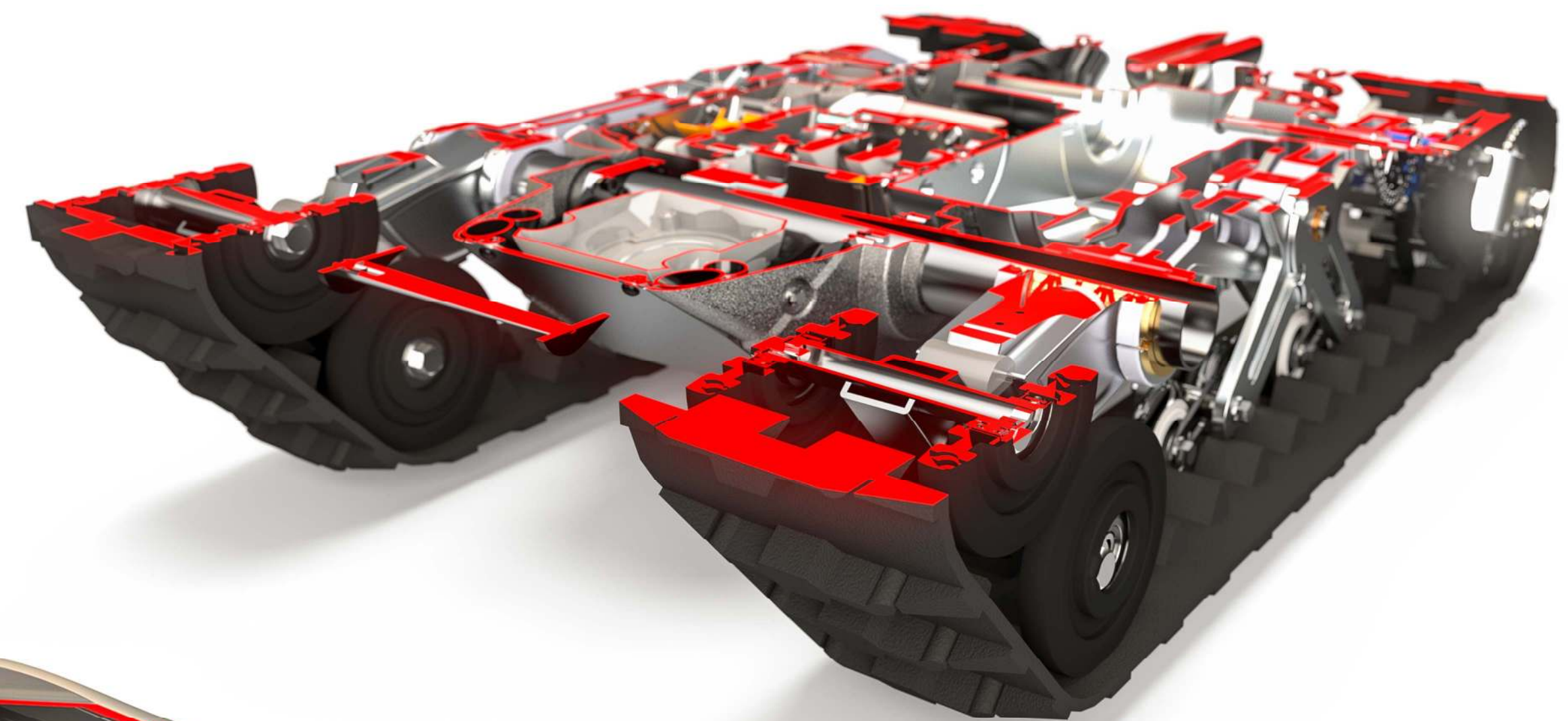
Iray Photoreal rendering simulation mode
brings trusted & CIE 171:2006-verified light transport simulation to *OV*
at the *Push of a Button*

But, as with all other *Iray* integrations, not limited
to only light transport simulation features



Awards received by  **BLOOM UNIT.**





..and even more new stuff

Iray 2020.1.1++ and 2021.0

AI quality prediction / automatic stopping criterion

Primitive Variables support (Vertex Colors, etc.)

Motion Vectors auxiliary buffer / AOV

Improved normal/bump mapping support
incl. optional high quality B-spline interpolation

CUDA render targets (and improved support for *OpenGL* targets)
incl. *CUDA* color conversions

Improved *DDS* compressed image support

Extended message tagging functionality

Via *MDL 1.7*, Unbounded mix:
Convenient for EDFs & VDFs, user needs to ensure energy conservation (BSDFs)

Via *MDL 1.7*, Textured EDF mixer weights, more freedom for textured lights:
More than just a single EDF is supported, any EDF hierarchy will work
Spectral values everywhere (not just the intensity slot)
Texturing of EDF parameters will work

Via *MDL 1.7*, Directional factor for EDF: allows for “under-coat emission”

... ..



..upcoming attractions

Iray 2021.1 ++

Improved sampling for many of the used algorithms
resulting in faster convergence (at least) when batch rendering
and less visible noise in interactive use

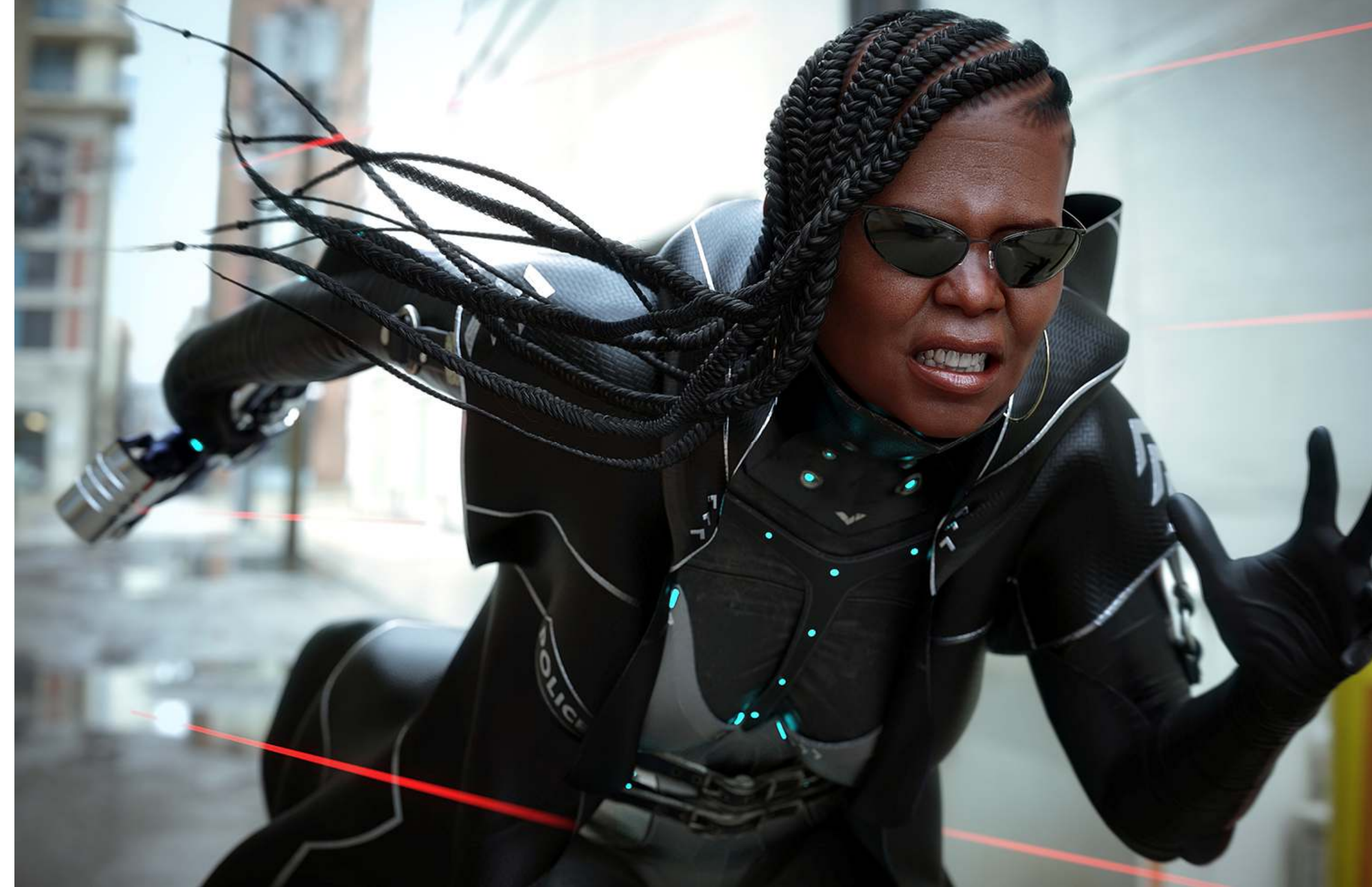
Improved interactivity / optimized preprocessing

Even better Omniverse integration

Native use of GPU texture compression (hopefully)

More optimal Arm CPU usage

Quite some more secret things ;)



Images courtesy of Daz 3D



Questions?

Acknowledgments

The Iray Team / NVIDIA ARC Berlin

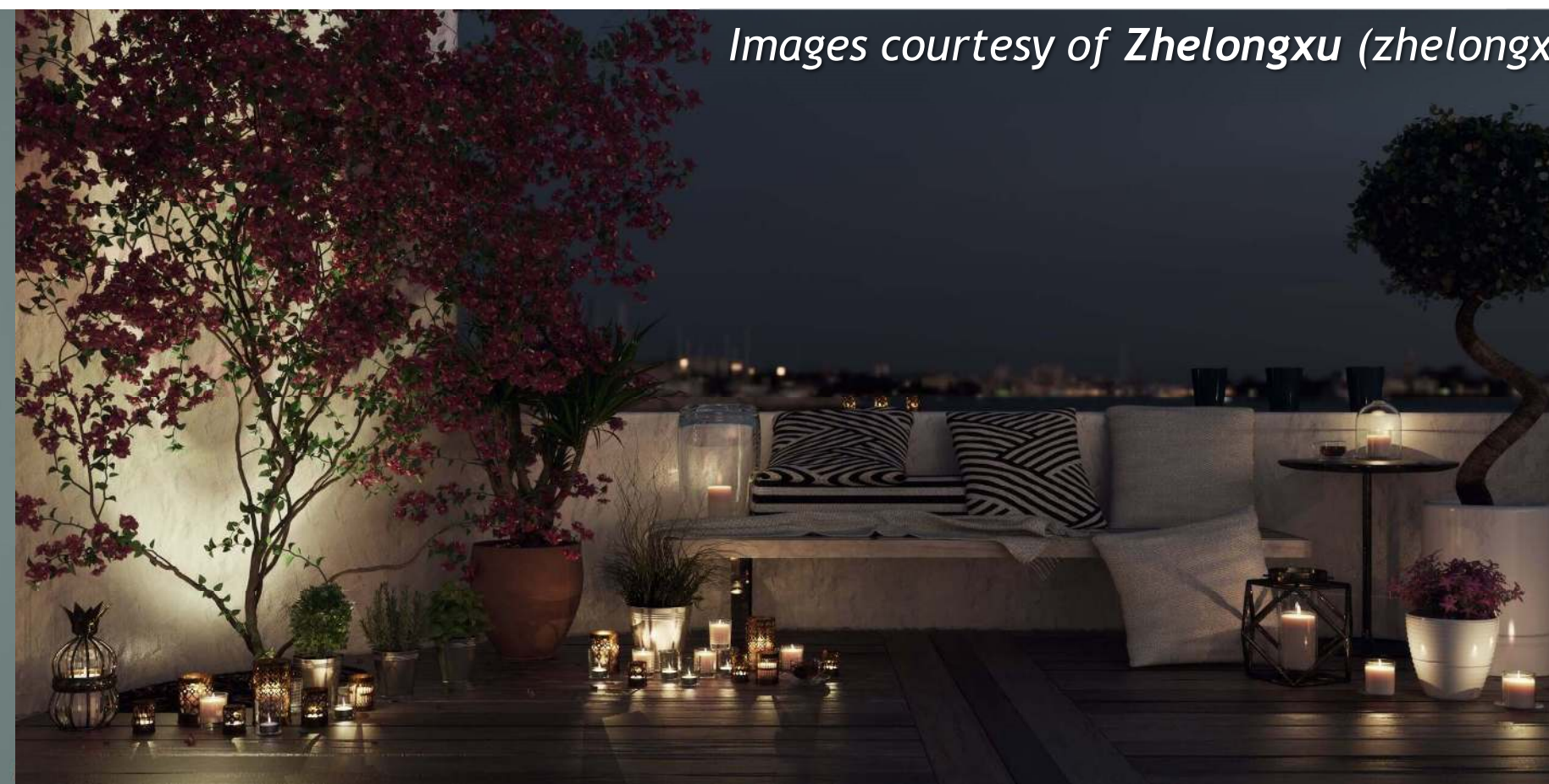
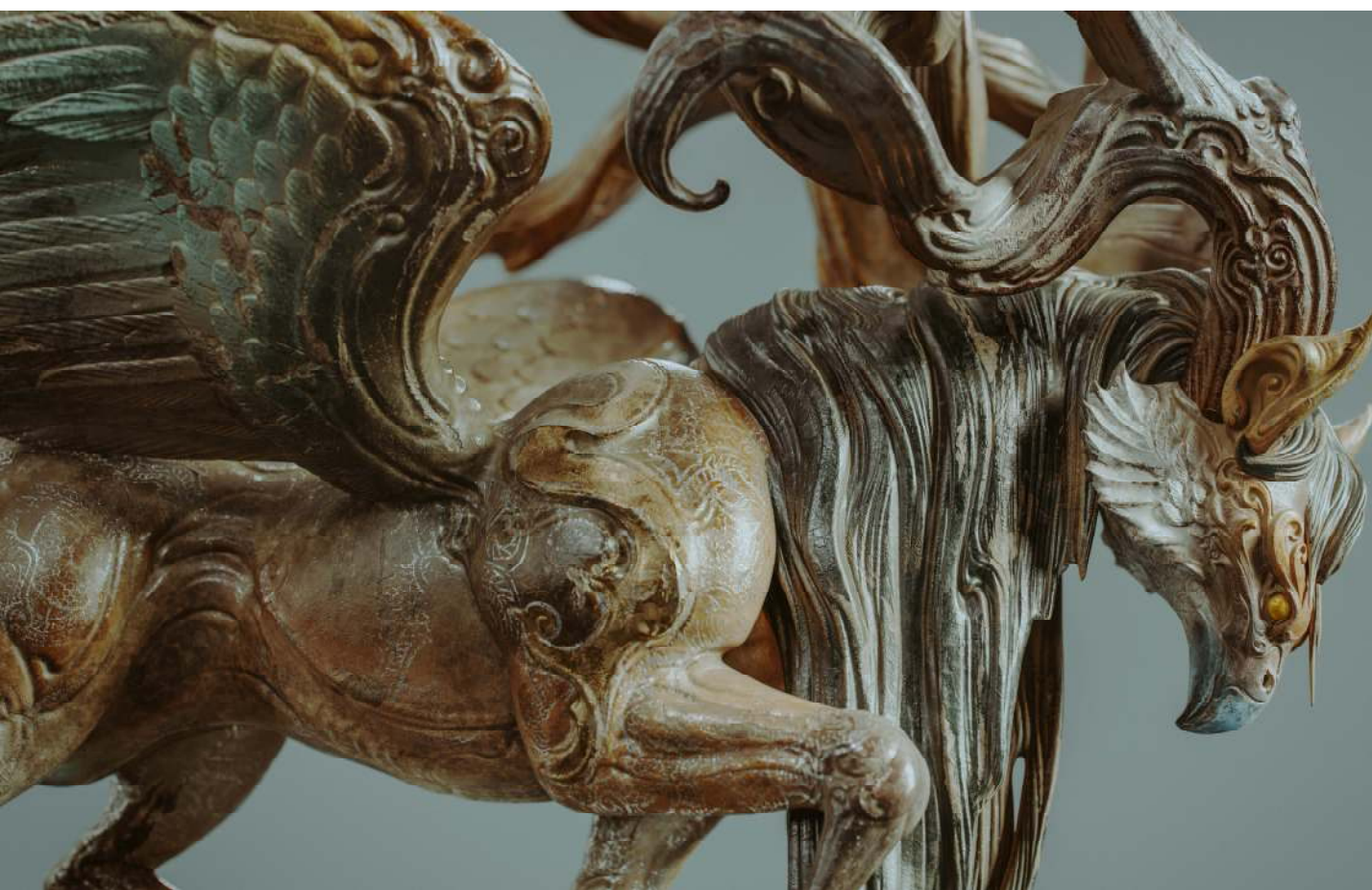
More Information

Techreport: The Iray Light Transport Simulation and Rendering System

<https://arxiv.org/pdf/1705.01263.pdf>

<https://raytracing-docs.nvidia.com/iray/index.html>

<https://www.nvidia.com/en-us/design-visualization/iray>



Images courtesy of Zhelongxu (zhelongxu.com) / Siemens Digital Industries Software (NX Ray Traced Studio)



 **NVIDIA**
GTC 21



Other sessions featuring *Iray* tech

SOLIDWORKS Visualize: Recent Developments [S31988]
Mike Sande, Sr. Tech. Sol. Consul. Dassault Systèmes

Learn how *NVIDIA Iray* Physically-Based Rendering
and *RTX* is Driving Enhancements to Consumer
Products & Retail [S31751]
David Hutchinson, Product Manager Siemens NX Vis.

Rendering Realistic Figures: A Detailed Look at
Skin Shaders Using *Iray* and *MDL* [S31893]
Christopher Jones, Daz 3D

Sharing Physically Based Materials
Between Renderers with *MDL* [S31207]
Lutz Kettner & Jan Jordan, NVIDIA

Integrating the *NVIDIA Material Definition*
Language in Your Application [S31241]
Sandra Pappenguth & Joachim Reichel &
Moritz Haenke de Cansino, NVIDIA

Real-Time, High-Fidelity Visual
Experience of Large-Scale Scientific
Simulations in *Omniverse* using
NVIDIA IndeX [S32064]
Marc Nienhaus, NVIDIA

