Course Notes, Siggraph 2010
Los Angeles, CA, USA

Faster Photorealism in Wonderland:

Physically based shading and lighting at
Sony Pictures Imageworks

Presented By:
Adam Martinez, Cg Supervisor
amartinez@imageworks.com

Sony Pictures Imageworks, Culver City, CA

Biographical Information

Adam Martinez is a Computer Graphics supervisor for Sony Pictures Imageworks and a member of the Shading Department, which oversees all aspects of shader writing and production rendering at Imageworks. He is a pipeline developer, look development artist, and technical support liaison for productions at the studio and he is one of the primary architects of Imageworks' rendering strategy behind *2012* and *Alice In Wonderland*. Adam started his career in commercial post houses and animation boutiques in New York City as a freelance computer graphics artist. He began his work in film visual effects on the project "Cremaster 3" by artist-filmmaker Matthew Barney. Since then he has served as both effects and lighting technical director, cg supervisor and pipeline developer for various studios in the San Francisco Bay Area. At ESC Entertainment, Adam led the effects team in the creation of complex insect crowd simulation tools for *Constantine* and destruction effects for *Matrix:Revolutions*. As computer graphics supervisor for The Orphanage on *Superman Returns*, Adam oversaw the creation of a ballistics simulation and rendering system. At Lucas Animation Adam was both a rendering pipeline developer and cg concept artist for television and feature animation. Adam's primary interest is in simulation and the construction of complex, but highly usable, systems for dynamic effects and rendering. Adam has a BA from Rutgers University. This is his first ever Siggraph presentation.

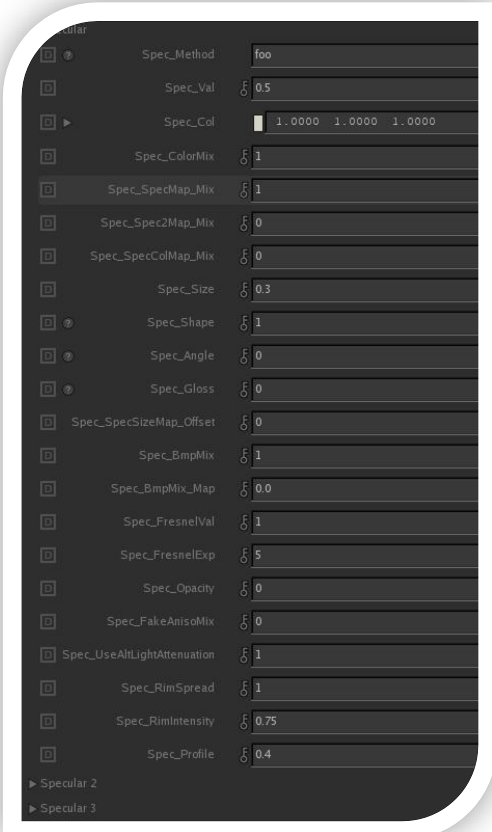
Introduction

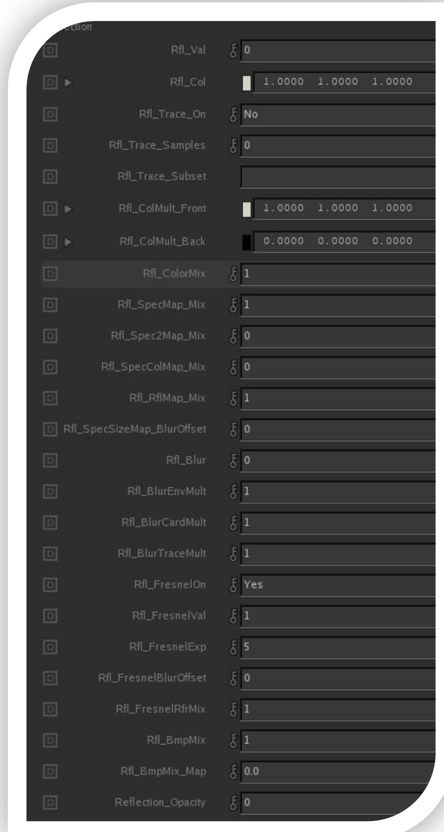
Films such as *Cloudy with a Chance of Meatballs*, *2012* and *Alice in Wonderland* (*Alice*) have represented dramatic changes in the way Sony Pictures Imageworks produces imagery. We have moved away from the traditional biased, multi-pass rendering system to a largely single-pass, global illumination system incorporating modern ray-tracing and physically based shading techniques. We will discuss how these changes applied to our work on *Alice*, and look at specific examples from the film. Topics discussed will include: motivations behind moving to a physically based rendering system and how such motivations would ideally manifest, micro-facet illumination models, global illumination and layered materials. We will talk about the impact of these developments on our creative approach and lighting procedures in production, as well as some of the unintended consequences of these changes.

Recent History

For projects as recent as *Watchmen*, Sony Pictures Imageworks employed a rendering system that does not natively account for global illumination effects. This system was heavily pass-dependant: shadows were depth-mapped, color bleeding came from point clouds, and reflection or environment occlusion was rendered in a separate pass, often in a separate renderer. Biased techniques such as shadow map and point cloud generation meant smaller sub-pipelines had to be developed and maintained. These techniques are largely phenomenological and physically inaccurate. Fidelity of the final result was subject to the quality of the passes that were generated previously. In addition, there was an enormous potential for version mismatches between separate passes. Even so, all of these entities had to be in place before anything approaching a photorealistic render could be accomplished.

These auxiliary pipelines required significant development and maintenance depending on the needs of a particular show or sequence. In addition, the amount of data generated per frame was so large that disk storage and synchronization of elements became critical aspects to the production. If any single shadow or occlusion pass was not in sync with the latest animation or light rig, it meant a lot of work hours tracking down the source of the divergence, and a lot of machine time re-generating the data. In the worst possible cases, a user would just be resigned to regenerating all the data on every update.





All of our shaders were based on the traditional phenomenological "Phong" model, and variations. There was no relationship between the direct illumination specular response and reflection response of the shader. Parameters for color, value and Fresnel response were replicated multiple times, meaning that the same phenomenon was essentially being dialed twice.

This shading system suffered from "feature creep" and near over-development. Parameter names adhered to very CG specific terminology there was very little notion of real world phenomena in the user interface. Terms such as "Specular Size" and "Specular Angle", have no real-world meaning and require a certain amount of education and familiarity to dial successfully.

Massive numbers of parameters and a monolithic shading architecture meant that users had only one recourse for feature implementation or experimentation; the shader developer. This also meant that productions mutated these shaders independently, which led to a certain amount of divergence across shows, and a potential loss of technology when a show wrapped.

Substantial interface real estate and shader development was devoted to simulating area lights in a rendering system that did not natively support the notion. Most of these methods fell apart when it came to complex lighting schemes, and would require significant dialing from both the surface material and the light material to get a passable effect.

Arnold Renderer

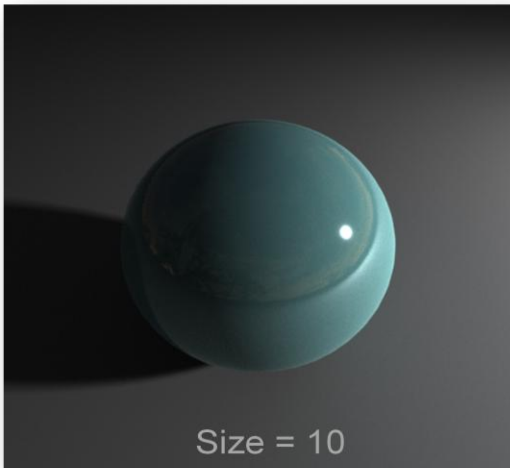
Arnold is Imageworks internal renderer, co-developed with Solid Angle SL. It is a modern global illumination ray tracer with a robust shading API and flexible scene format. For most productions, Arnold was deployed as a utility renderer; generating ray traced occlusion passes for use in compositing. On a few productions it had seen use as the primary rendering engine; Monster House and Cloudy are both full CG animated productions. Cloudy was the driving force behind shader and renderer development for Arnold which made it a much more plausible option for visual effects films. Eagle Eye used the shading system developed for Cloudy, but it was *2012* that really pushed for a new shading system that offered faster photorealism.

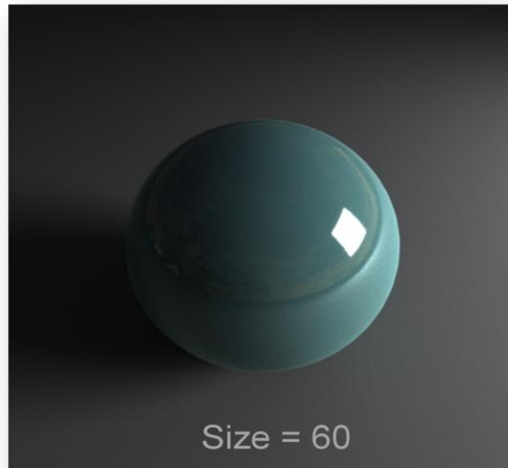
At the same time that *Alice* and *2012* were beginning to evaluate the rendering options, there was a lot of discussion throughout the facility about development philosophy in general. Rob Bredow, Imageworks' CTO, made it clear that the push towards using Arnold, and technologies like it, was to provide more photo-realistic imagery out of the gate, but to also make our human time more productive. This was highly informed by his experience supervising Cloudy, where he witnessed Arnold's capacity for consistency in shot production, provided the requisite time was spent in asset development. This idea coincided with another facility mandate to take interface simplification very seriously. This idea was intended to provide a framework for reducing interface clutter and, ideally, making is so that a multiple doctorate was not required to shade a frog. The task was put before us in the shading group to develop a system that was human readable, intuitive to use, and deployed at the facility level. To top it all off, we had to come up with strategies and programs to educate users on how to use these new tools.

For *Alice* specifically, there was a strong desire to leverage the power of the ray tracer to implement a physically based shading system, but also to expand the capabilities of the shading system during the look development stage. Previously, realism was difficult to control and maintain in the face of creative demands. Supervisors wanted a system that would allow them to start the creative process sooner, and maintain a consistent level of realism.

2012 and *Alice* committed to using Arnold, and a yet-to-be developed shading system. This was going to be no small chore. Developing shaders for the renderer was a known quantity, but how to deploy them and what form they would take to the end user was a major question. Other issues we needed to address included the pipeline infrastructure required to handle the predicted volume of data going through the renderer. This also made us sharply aware that our render times were going to be significant, and we knew we were going to have to balance physical accuracy with reasonable times.

Part of the development strategy was informed by the functions of the renderer and how switching to it inherently changed the pipeline and production approach. Lack of maps or need for occlusion passes did away with all of the previous secondary pipelines, and their associated maintenance costs. Light sources in Arnold are implemented as area lights natively, which means no extra development in the shader is required to support them.





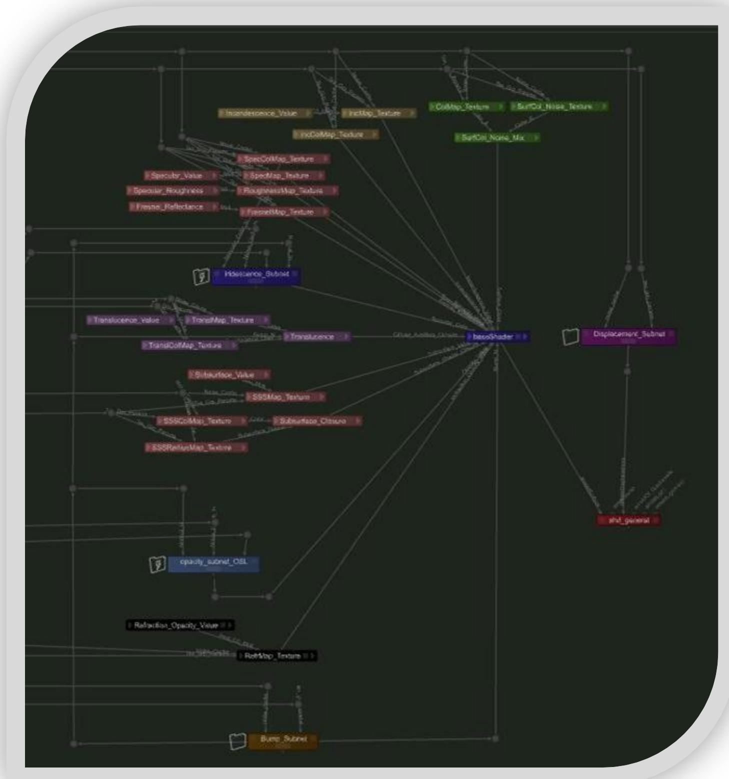
This illustration demonstrates the effect of area light size on a scene. The only variable changes was the size of the light source. On the left, you can see the change in the size of the specular highlight, the softness of the shadow and the diffuse falloff, or "wrap" around the surface normal terminator. The shader has no parameters to control this behavior, it is purely a function of the light source and Arnold's sampling engine. This not only demonstrates a key feature of the lighting pipeline, but is also a great example of interface simplification.

While global illumination is accomplished by indirect sampling means, point clouds weren't entirely extinguished. Arnold has a built-in point based subsurface scattering implementation, but this is entirely generated by the renderer inline to the process and uses the same scene information as the beauty render. In addition, the subsurface effects is informed by and reflected in the global illumination solution by default. As a quick side-note, we did implement our own traced sss used extensively throughout the film, based on Henrik Wann-Jensen's 2001 paper, *A Practical Model for Subsurface Light Transport*.

We also implemented a photon mapped caustics solution which was initially developed for the Watchmen, glass palace sequence. This was based on the methods described in Jensen's 2004 course *A Practical Guide to Global Illumination Using Photon Mapping*. This system followed a similar path in that the photon map was generated at beauty render-time with the actual scene data. Energy is emitted by light sources into the scene and is either reflected, transmitted or absorbed. This energy is stored in a point cloud representation of the photons and then referenced during the illumination stage of the beauty render.

The success of Arnold's global illumination effects depend on the exchange of energy between surfaces. The renderer developers and the shading department both encouraged rendering as much geometry in camera as possible, and only splitting out passes when required by compositing needs. Various memory management strategies built into the renderer helped with this, with minimal user intervention required for special circumstances.

Because Arnold is a stochastic ray-tracer, implementing real-world lighting tools such as bounce cards was a relatively simple matter. We advocated the use of bounce cards and even incandescent emissive geometry in scenes, over the use of special purpose CG light sources. As a result of the Arnold renderer's robust HDRI sampling, most, if not all sequence lighting started out with a sky dome, which was an image mapped sphere encompassing the entire scene. This special purpose entity was the primary source of mood and context for a particular sequence, and usually stayed constant for each shot.



Designing the Shading System

The shading department made some initial design decisions that had far-reaching implications for how we worked as shader developers, and how we maintained the shading system for productions. The decision to use a node based network for shading was motivated primarily by these maintenance concerns. A networked shader can be portioned out to different developers, parts of it can be repurposed, and nodes can be developed once, and re-used in many places.

In the illustration above you can see a portion of our general shading network. The shading system centered around a root illumination node with encapsulated all of the basic lighting procedures. Procedural and map based texture sub-networks feed this root node with parameter information, such as surface color, specular roughness etc.

When it came to establishing default values, the motivating factor was to give users a more accurate image. This meant enabling reflections, global illumination up front, and for optional shading components supplying sensible and realistic default values without causing a dramatic slowdown in turn around.

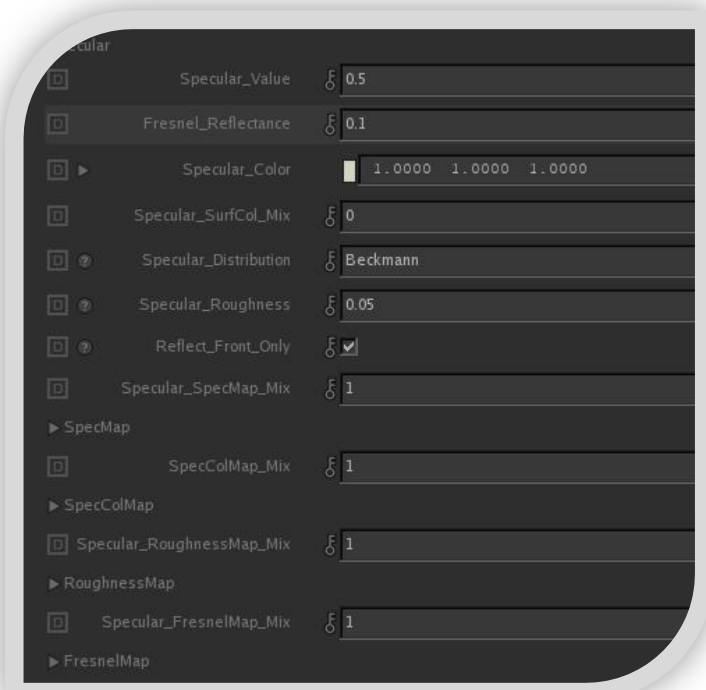
By far the most critical aspect was to involve the users on *2012* and *Alice* from the beginning of the design process. This level of involvement was an opportunity for two-way communication between the developers and the artists. Artists gained early familiarity with the tools as they were being developed and regular feedback and feature requests ensured minimal wasted effort on the development side.

Implementation Details

Much of the information in this section is explained in much greater detail by the source material. Our implementations followed the source material fairly accurately.

For general specular response our shaders employ the Cook-Torrance microfacet functions from Bruce Walter and co. 2007 paper *Microfacet Models for Refraction Through Rough Surfaces*. In this paper the authors describe a flexible bsdf architecture with brdf, sampling and weighting functions for multiple distributions in both reflective and transmissive rendering contexts.

In [Walter07], the reflection term is described by equation 20. It is defined as a Fresnel term, F, a shadow masking term G and a microfacet distribution term D. In our implementation, F is the Schlick approximation. G and D are variable depending on the distribution selected. Equations 25 and 27 describe the Beckmann distribution, which is also our default distribution in the shading system at Imageworks. G is a rational approximation of a gaussian distribution of microfacet normals. Equations 28 and 29 are the sampling equations to generate microsurface normals from random number pairs in a Monte-Carlo sampling scheme.

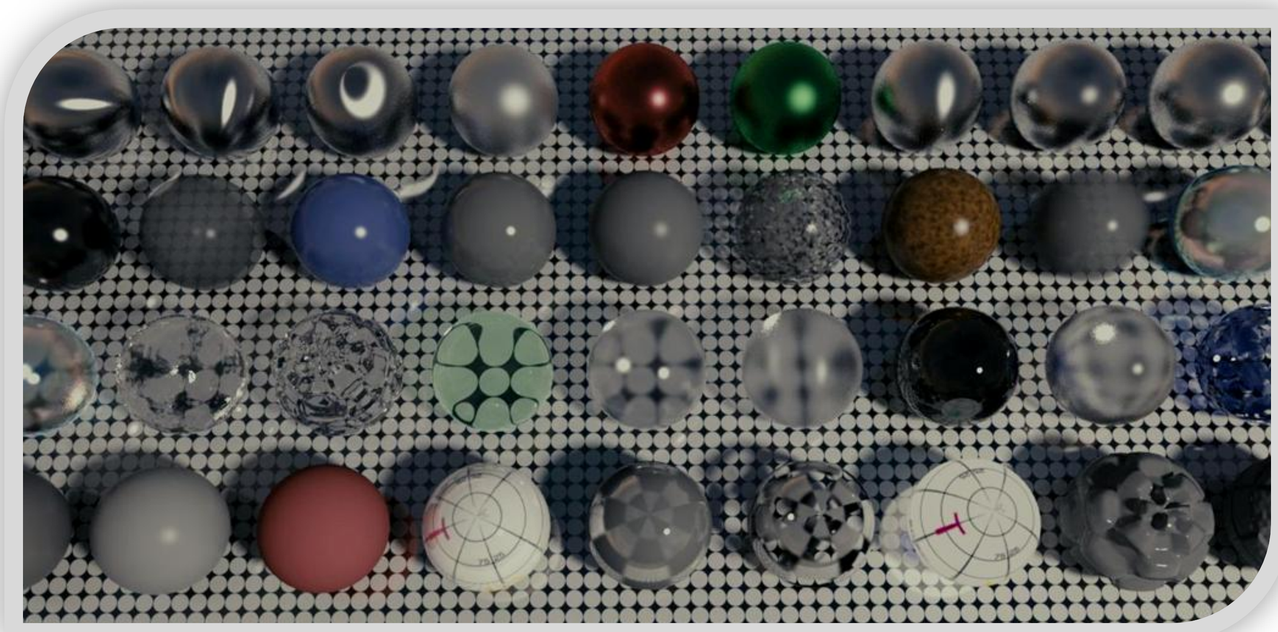


The parameterization of these functions is deceptively simple: roughness and index of refraction. And yet the model controls direct specular reflections from cg sources, indirect sampled reflections and refractions. What we get from this is a common appearance of surface roughness between all three of these contexts. This contributes greatly to an overall physically plausible look, and we realize a significantly reduced parameter set over our old shaders. In addition the term “roughness” is a commonly used term to describe real world surfaces, and it translates quite well into our system. Frensel Reflectance is a bit more obscure: we use it to describe what is actually the index of refraction, or eta in the Schlick approximation, (the Fresnel term), of the previous equations.

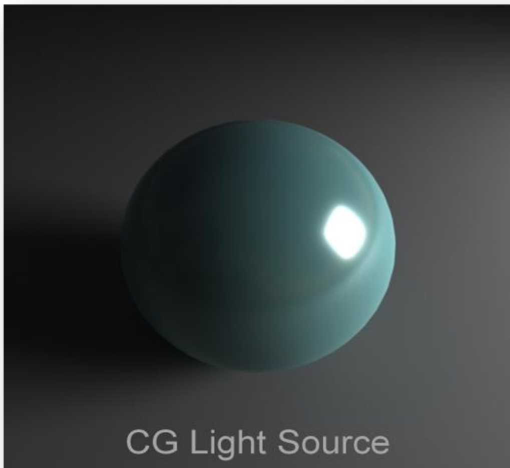
Roughness is a normalized parameter from 0-1 and controls the spread, or “blurriness” of the direct and indirect reflections. At high values it approaches a perfect diffuser, but resolving this requires a significant number of samples.

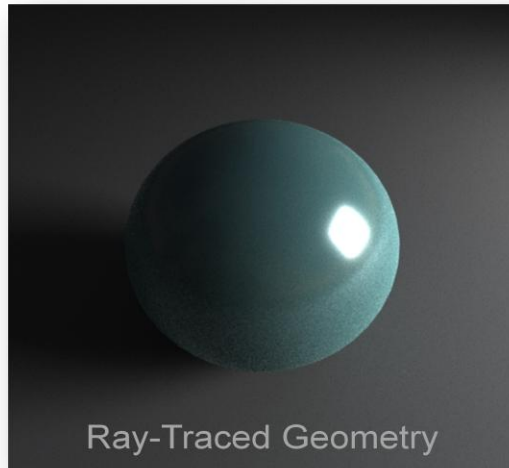
The fresnel response is also a normalized parameter from 0-1. At low values, around 0.02, the material behaves as a perfect dielectric, like plastic. At 1, the material is a near-perfect conductor, or metallic surface. In our implementation we used the Schlick approximation for normal-incidence reflectance.

These are three examples of the most commonly used specular distributions in the shading system. Beckmann is the default and is widely used for most materials. GGX is a distribution introduced in the Walter paper and we found that is samples slightly better at very low roughness settings, which makes it a good candidate for corneas and other wet surfaces. The anisotropic distribution is based off of the work of Ashikhminh-Shirley and is the primary specular model used for metal surfaces on *Alice*.



This illustration shows our standard shader network with a variety of parameter settings. This image is used to test shader behavior as new versions are released. You can see a variety of distributions and settings here including anisotropic reflections, with various patterns, as well as anisotropic refractions. In our implementation we consciously decided to decouple the index of refraction of the reflection component from that of the refraction component for creative reasons. Absolem, the caterpillar is an example of this use. The refractions of his eye are raytraced, but the effect of the eyeball filling the lens had to be dialed separately from the glassy reflections at the surface interface.

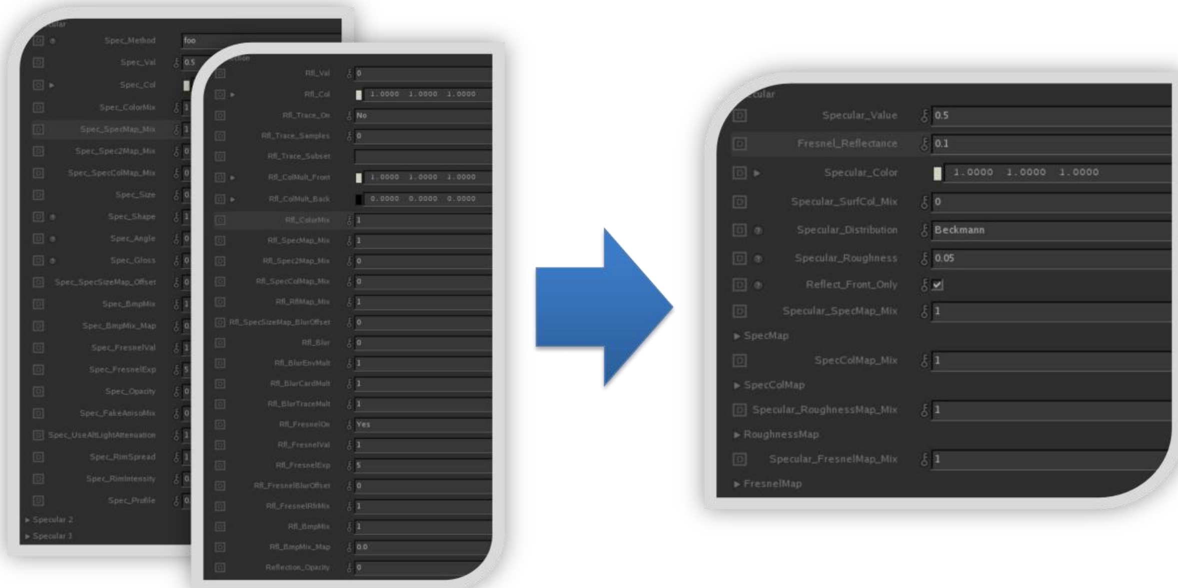




This illustration demonstrates the coherency of our implementation across direct and indirect illumination sampling methods. On the right is the sphere rendered with a cg light source reflected using the just the shading model. On the left is a card with an emissive material, it is being traced in the reflections using the sample distribution.   This behavior makes the architecture very flexible in production by allowing artists to substitute lights for geometry and vice versa as the need may arise, and get a predictable result.

Development Successes

In summary, the implementation of a microfacet BSDF architecture provided numerous benefits to the shading system: We realize visual coherence between the direct and indirect reflection methods as well as consistent behavior from refractions, and our shaders provide realistic response that is consistent across all lighting conditions. Since we combine the specular illumination and reflection terms into a single interface, and by virtue of the renderer's native behavior, we also accomplished substantial interface simplification and a streamlined the user experience.



Dozens of parameters were reduced to a few parameters that appropriately affected multiple components. The user should not have to adjust any other settings to get the same response from direct light sources and traced reflections. Dialing the simpler shader is significantly easier, but didn't we lose some flexibility? How did this change affect other departments in the color and lighting pipeline? There was certainly concerns regarding the combination of illumination and traced reflections, and the impact across many of the departments was significant.

With the simplified ui came a much smaller learning curve on how to use the shaders, but more development was needed to accomplish complex hybrid looks that *Alice* would require. It required a shift in the thinking of our look development artists from composing a highlight on a surface, to critical thinking of the nature of the surface itself. Early on in the process I would advise artists to ask themselves: How would this surface feel to the touch? Smooth or rough? Is this material more like plastic, or glass? Stone or metal? Answering these questions, one could arrive at initial values for roughness and Fresnel reflectance and then go from there. Alternatively, we had contextual help dialogs that listed example settings.

In the technology arena, we had to design tools that would allow the artists to create complex looking materials without having to develop a new shader every time a feature needed to be added. Our node based network shading system allowed us develop a straight forward material layering scheme. Material layering has been present in commercial software for many years. However Imageworks traditionally avoided such schemes due to consistency and efficiency concerns, not for lack of trying. Our approach leveraged a simple interface that stacked shaders in an intuitive way.

Material Layering

Material layering allowed for much greater flexibility in the shading system and room for experimentation with hybrid materials. Red knight for example is made up of dirt over black paint, over red paint, over metal. Each layer has very particular settings based on look desires. The presence of each layer is determined by an opacity control map. Most layers can make use of the same maps. The opacity mask for the red paint in this example, is also one of the bump maps for that layer. Variations on this character were as simple as swapping out the opacity mask for the black paint.

The production also realized less dependence on shader writers; look dev artists could easily add their own secondary specular response, or other shading effects and composite a material without too many restrictions. Material Layering became a critical aspect to the definition of looks on *Alice*: it was used to add slime on frogs, puddles of water on cobblestone, moss on stone.

Look Development

The lighting environments that were set up for *Alice* were approximated from the physical studio lighting used on the actors during reference photography and costume acquisitions. Many of these early light rigs became standard light setups for the look development turntables.

In the past, this was common practice to use a different light rig for characters than for environments. In a global illumination context it doesn't make sense to use two different light rigs; indirect illumination depends on a homogenous environment. Therefore sequence and shot lighting was always designed with the characters in context.

The initial expectation was that combined specular and reflection parameters would result in reduced ability to art-direct and dial the looks. In practice however, the imagery looked more correct out of the box, we almost never received direction to dial specular illumination and reflection independently.

It required some education on the part of the texture painters to distinguish specular intensity from "blurriness." Roughness was a parameter that had never been present in prior shading systems at Imageworks. However, with microfacet models it is a phenomenon that is critical to the realism of any material.

Texture Painting

All of these developments had a significant impact on the texture painting departments as well. Fewer parameters in the system and more combined controls meant an overall reduction in the number of maps that were needed for an asset. In addition, the modular nature of our new shading system allowed look developers to experiment with texture maps for unintended purposes. Control maps are for the most part floating point data, intended to be used for attenuation, bump or utility purposes in the shader. These were always painted in a 0-1 normalized range and dialed by the look development artist using a set of built-in correction controls available for each map. Simple masks were by far the most common maps used on creatures, since they defined regions where a material layer would be active, but they could also be used to mask other procedural effects etc. Variations among multiple characters can be as simple as a single shader setting on a shader or material layer, or as complex as an entirely different set of color and control maps. The red knights are an example of map-based and procedural variations in practice. Each particular red knights numerical designation is a simple mask on the black paint layer. Dirt and grime are modified by procedurally modifying parameters on the materials at render time.

Hair Shading

The grooming, animation and rendering of hair on *Alice* is worthy of a course all its own. The geometric complexity of hair makes it very challenging to sample efficiently in a ray-traced global illumination context. We had to establish new techniques and procedures for dealing with the amount of hair on the show.

Shadows on and from hair were always ray traced in the same manner as other geometry in the scene. This was a major source of fidelity and clarity in the renders of hairy creatures, but also of massive render times if not managed carefully. All of our hair received indirect light from the skin and surrounding environment by default. However, the hair did not occlude itself in the indirect diffuse solution. While completely possible, the sampling requirements and render time overhead of tracing the hair-to-hair occlusion was too prohibitive but in most cases this was a acceptable compromise. For these same reasons, the hair did not trace glossy reflections.

Traditionally look development artists leveraged the inaccuracies of deep shadow maps and point clouds to create a sense of softness and hair-to-hair energy transmission. Because none of these techniques were available to accomplish softness in the hair shading we relied heavily on an accurate model of the hair volume, and careful procedural texturing of the hair opacity. Setting up a successful hair look was an exercise in balancing hair thickness with transparency. While the up front cost of setting this up per character was significant, rendering hair was much less problematic in shots than in the past.

The door mouse presented a particularly interesting challenge in the hair-to-subsurface scattering interaction.

The Catch

None of this came without a cost.

Our render times for the average frame were long. This was not unexpected: we are asking the renderer to do a lot more at one time where previously we were executing a lot of incremental steps. This cost was easily offset by the amount of time saved by not having to render a massive number of passes through the system. In addition the predictability of the resulting images meant less tweaking for consistency with surrounding shots or sequences. Overall, the net rendering costs of switching to Arnold, did not go up nearly as much as anyone thought it would.

One of the most notable problems of ray-traced imagery in production is sampling noise, and because we were essentially sampling more than we ever had before, noise was an issue. One of the major culprits of sampling noise is contrast between consecutive samples, and this could usually be tracked down to light sources with very non-physical properties. Lights with zero area, an intensity of one and no decay over distance have no real world counterpart, and tend to introduce large amounts of energy in to the global illumination solution.

Case Study: Red Queen Throne Room

Let's take a look at an example of one of the more challenging look development and lighting cases. The red queen throne room is an ornate, richly colored set with no shortage of reflective surfaces, high-contrast lighting and fine geometric detail. The mood of the sequences in this location is sinister and threatening, but still well lit.

Interiors were generally difficult scenes to tackle. In this environment the key lighting is a bright exterior source, and it is considerably brighter, and cooler than any interior sources. This situation exacerbates sample contrast and contributes to noise in both the indirect diffuse and specular solutions.

There are numerous small accent light sources, from the candles, that need to contribute to the scene as well. We found that we could either cast specular illumination from these source OR reflect the flame geometry, but rarely if ever would both be appropriate. These accent sources are very localized, but will be computed for samples that are very far away, which is a waste of resources.

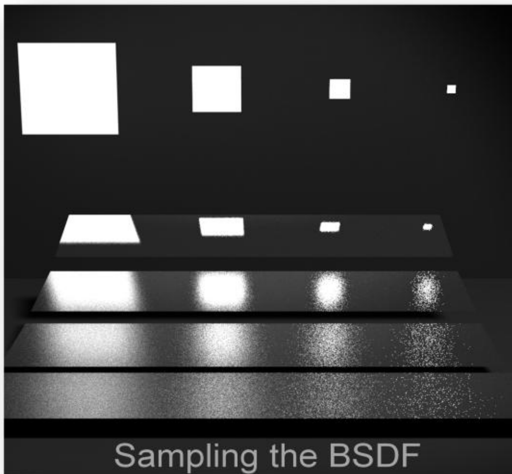
Fine geometric detail combined with highly reflective surfaces also proved to be an sampling challenge. The column bases for example are almost entirely defined by their specular response. This required multiple glossy specular bounces in order to resolve column self-reflections appropriately. In addition, the columns and the characters have conflicting sampling requirements. Where the characters would benefit mostly from cg light sources, the columns do better reflecting geometric representations of the light sources.

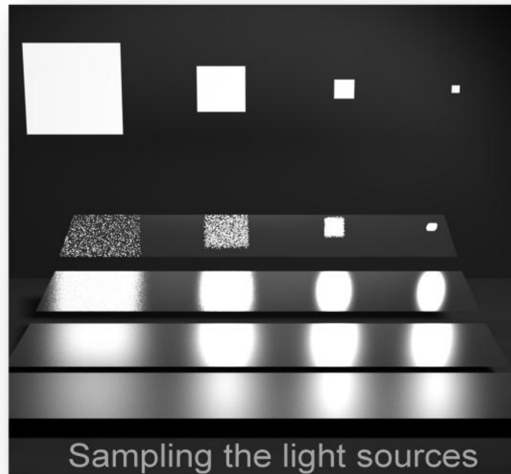
Sequences set in the red queen throne room were some of the first in the pipeline. These shots became a test bed for significant experimentation with the renderer and the shading system. One of the things we learned from this environment was the expense of indirect versus direct light sources. In most cases the indirect illumination was a fixed cost while the direct illumination cost increased as light sources were added to the scene. We found that substituting cg lights with geometry representations solved some sampling problems very efficiently.

We also implemented hooks into the shading system to cull out shader operations in certain contexts. By default, the specular component of the shader is not evaluated during the indirect diffuse solution, such a computation would be a caustic effect that would be extremely expensive to sample appropriately (but not impossible). Additionally we can cut the cost of our secondary illumination solution by removing light sources that do not significantly impact the overall result. Candle lights for example will illuminate surfaces during primary, or camera-ray shading, but will not be calculated during the indirect diffuse solution.

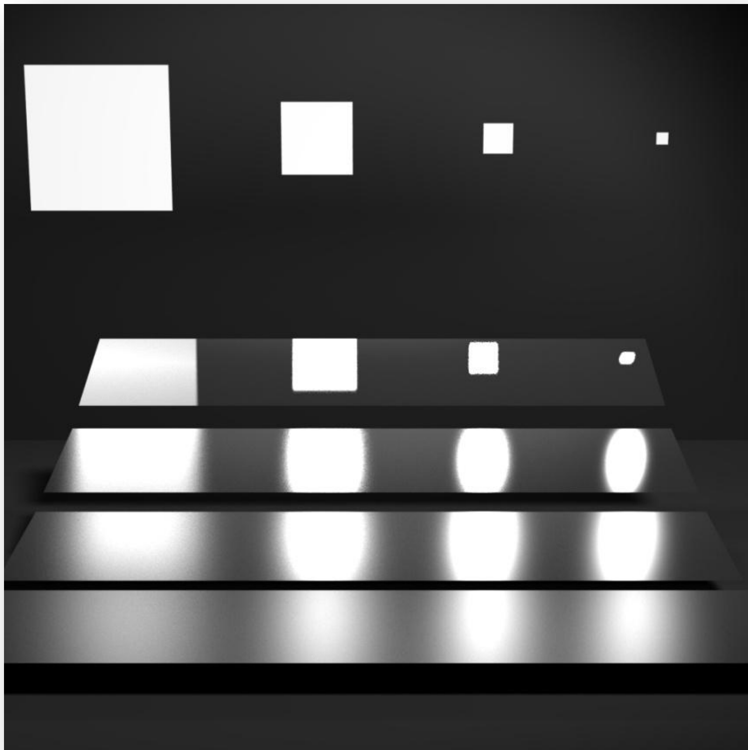
Other efficiencies were gained by altering the behavior of material layering in secondary shading. User can selectively disable the layering features in favor of a less complicated single-shader representation of the surface. By far the most effective solution to sampling problems, and render times, was Multiple Importance Sampling.

Multiple Importance Sampling: The Glossy Reflection Problem





On the right is an image of four area light sources emitting onto four planes of increasing roughness. This is sampling the light sources and evaluating the BRDF at each sample.
On the left is an image of a series of emissive cards being reflected in the same planes. This is sampling the BSDF of the shader. You can see that when sampling the light sources, the sampling scheme falls apart when large area light sources are combined with very low roughness values. On the other hand when sampling the BSDF, that case is handled quite well. However, for small area sources and high roughness values many of the reflection samples miss and we get significant noise in the results.



As presented by Eric Veach in his 1997 thesis *Monte Carlo Methods for Light Transport Simulation*, multiple importance sampling (MIS) is a modification of importance sampling designed to alleviate sampling artifacts in the glossy reflection problem.
Importance sampling depends on a probability density function (PDF) to ensure that stochastic samples are oriented in meaningful directions, and the sampling distribution does this for us. MIS introduces a balance heuristic to weight samples from both sampling methods in order to reduce variance.

MIS significantly increases the efficiency of our sampling and results in an overall reduction of noise for almost every situation. Having a meaningful PDF that corresponds accurately to the BRDF and sampling distributions is critical to the overall effectiveness of a BSDF in a multiple importance sampling scheme. The models we chose to implement had well defined PDFs that we were able to rapidly integrate into our existing shader architecture.
MIS helped enormously in reducing noise in glossy reflections. By choosing the best sampling directions for the columns versus the characters, sampling settings approached a “one-size-fits-all” ideal.

Case Study: Lighting and Rendering The Final Battle

The final battle sequence of *Alice* was particularly challenging on a number of levels. In this section, we are going to discuss the approach to lighting this sequence with a new renderer and shading system.

Final Battle: Geometric Complexity

Traditional instancing strategies broke down when we introduced procedural displacements to geometry. While scene memory use was manageable for most situations, computation of indirect shading slowed down dramatically. Limited trace distances and making small objects invisible to secondary lighting effects reduced the rendering expense of computing a the indirect lighting components, and the expense of negligible accuracy in the energy transport simulation.

Final Battle: Lighting Strategies

The final battle sequence incorporated a significant amount of set, creature and live action interaction. At the outset, it was a given that the lighting of all aspects of the sequence had to be consistent. The initial approach was to create a broadly defined lighting rig that could be distributed to all of the shots.

The stage lighting informed how the initial lighting rig would be built. Generally, actors were lit with bounce cards to the right and left of camera which served to fill in darks and soften shadow areas significantly. This on-set approach was simulated in cg using the natural falloff of emissive geometry in the indirect diffuse light computations. The overall look of the sequence was that of an overcast morning. The artwork however had also depicted a high contrast between the darks and mid tones which became a balancing act for the lighting team.

While single pass rendering was generally encouraged and successful, it was not always practical in the live action visual effects context. Usually separate passes had to be generated to separate foreground and background elements relative to a plate. Sometimes special utility passes had to be generated to integrate live action elements more appropriately. These passes still had to integrate the entire scene geometry to accurately represent shadows, bounce light and reflections. Since render times were expected to be fairly long, it was common practice to light shots at half resolution and with lower sampling settings. We were still getting all of the secondary shading effects and subsurface scattering which helped us judge the lighting more accurately than if we had rendered without those effects at full resolution. Supervisors got used to these slightly noisy renders and could provide more creative feedback earlier in the lighting process.

Final Battle: Subsurface Scattering

Subsurface scattering proved to be a special challenge for this sequence as it was a key characteristic of the white knights look. We found that using point based subsurface was too unwieldy for the renderer as each character had to generate a separate point cloud per frame. This amount of data would fill up ram rather quickly. It became much more efficient to use the ray traced subsurface scattering implementation, following [JENSEN01], as the cost was significantly cheaper for many hundreds of creatures that were small in frame. The traced scattering solution also tended to preserve more geometric and texture detail over the point cloud solution.

Conclusions

Sony Pictures Imageworks is one of the largest visual effects facilities in the world. Over the past year and a half we have managed to transform the entire rendering strategy for feature film visual effects production. The effect of these changes are numerous and profound.

We have moved away from a biased multiple pass rendering system, to a largely single pass global illumination system incorporating modern ray tracing and physically based shading techniques. Our shaders for surfaces and light sources respond much more naturally than previous implementations. Robust global illumination methods combined with efficient production practices have allowed artists to experiment freely with simulating on-set lighting practices.

We have streamlined the users interaction with the shading system. We have reduced the number of parameters and associated them with real world surface phenomena that can be discussed with clarity. We have provided a set of tools such as material layering to intuitively add visual complexity.

The impact on shot production and lighting procedures has been equally profound. Master lighting setups translate predictably to different shot settings, and lighters can get a sense of shot-to-shot consistency within a sequence much faster than with previous methods.

And finally, since the first frame rendered incorporates the full complement of shading components, including indirect diffuse bounce, image based lighting, glossy reflections and ray traced refractions, lighters and supervisors are much better equipped to make creative lighting judgments much sooner in the production schedule.

Moving Forward

At Imageworks we continue to research, experiment with and engineer new rendering solutions. We are continually developing new techniques, both procedurally and technologically, to use these tools in production. Current areas of development interest in which we are extending physically based shading ideas are: specular to diffuse energy transfer (aka caustics), subsurface scattering using ray-traced lighting and global illumination, and efficient and realistic ray traced hair shading.

Imageworks' sponsored Open Shading Language open source development incorporates a lot of the work discussed here in a generic framework for integration into almost any rendering engine. In our use of OSL internally we have found the language to dramatically improve our ability to get shaders to users rapidly. We have also seen much better consistency in the resulting imagery, owing to a much more homogenous sampling environment, and the implementation of illumination models as closures. OSL Shaders themselves do not contain explicit lightloops, but rather make calls to library functions that will be evaluated at a later stage in the rendering process.

Because OSL abstracts the illumination integration process out of the shaders, the renderer is free to make decisions about what techniques are most appropriate to solve the

lighting . This continues to be an active area of development for the renderer developers and already has yielded excellent gains in the area of IBL importance sampling.
As always, we continue to search for solutions to long render-times, but ultimately the quality of the image is the top priority.

References

[1] [WALTER07] Walter, B., Marschner, S. R., Li, H., and Torrance, K. E. Microfacet Models for Refraction through Rough Surfaces. In *Rendering Techniques (Proc. EG Symposium on Rendering)*. (2007)

[2] [VEACH97] Eric Veach Robust Monte Carlo Methods for Light Transport Simulation, Stanford University, (1997)

[3] [JENSEN01] Jensen, H. W. , S R. Marschner , Levoy M. , Hanrahan P., A practical model for subsurface light transport, *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, (2001)

[4] [JENSEN04] Jensen, H. W. , A practical guide to global illumination using ray tracing and photon mapping, *ACM SIGGRAPH 2004 Course Notes*, (2004)