Global Illumination Across Industries

Course
SIGGRAPH 2010

Jaroslav Křivánek
Cornell University & Charles University, Prague

Marcos Fajardo
Solid Angle SL

Per Christensen
Pixar Animation Studios

Eric Tabellion
PDI/DreamWorks

David Larsson
Illuminate Labs (now Autodesk)

Michael Bunnell
Fantasy Lab, Inc.

Anton Kaplanyan
Crytek GmbH

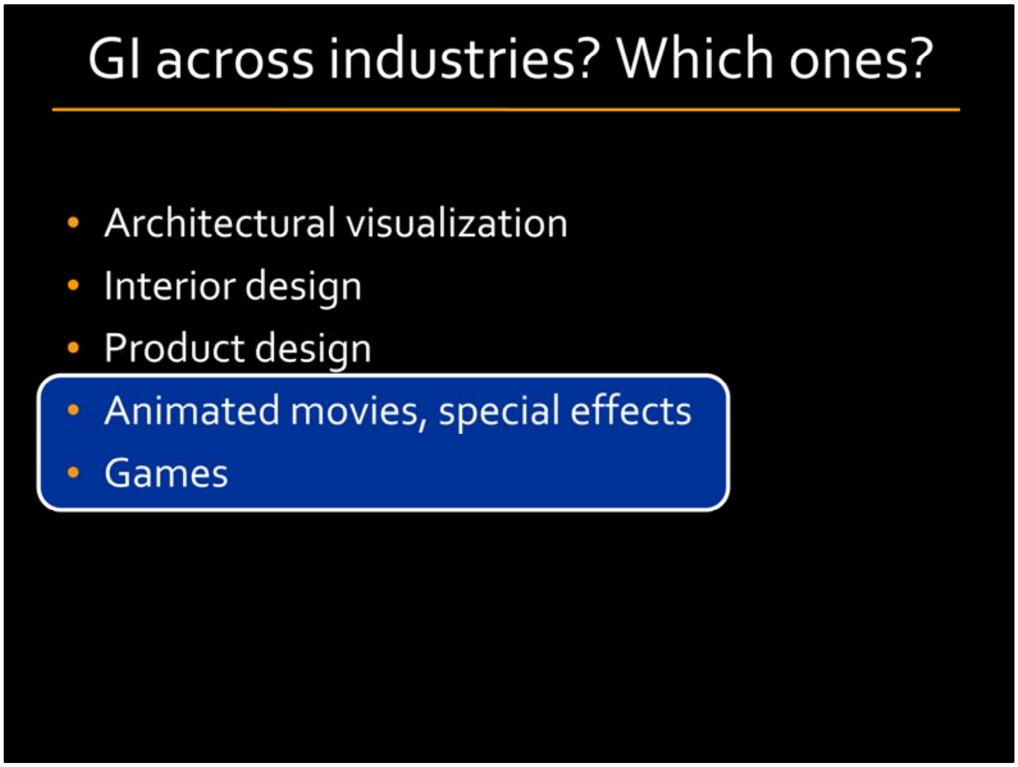
Global Illumination Across Industries

Introduction

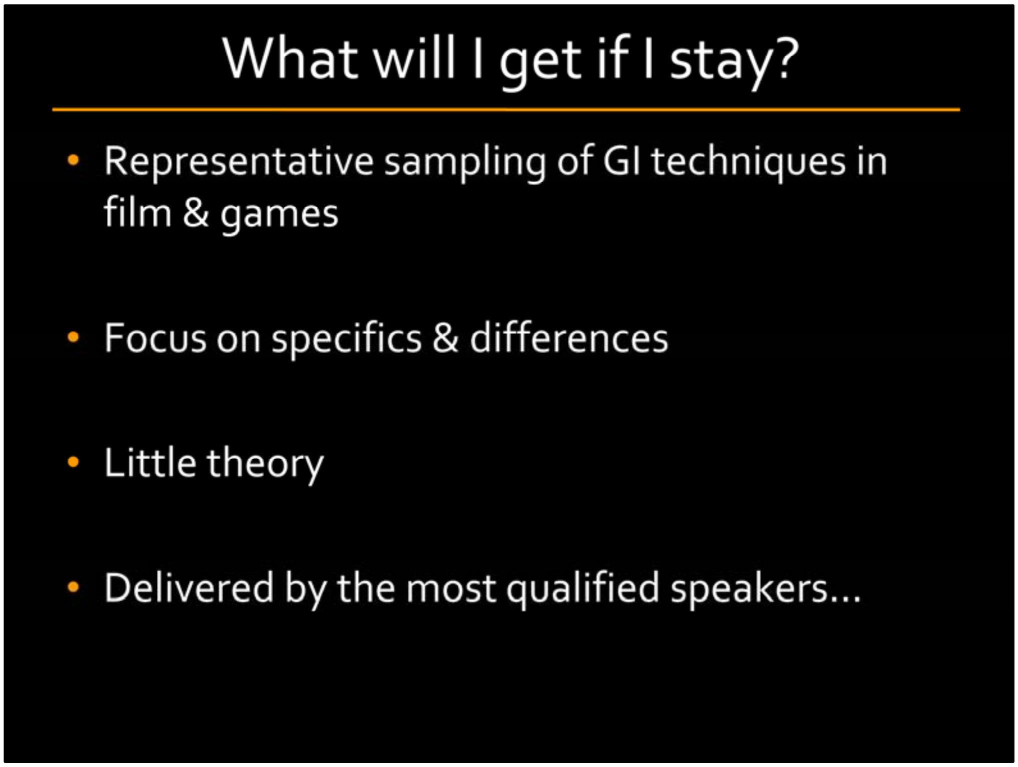
Jaroslav Křivánek
Cornell University &
Charles University, Prague

This course is about global illumination (GI), which, in the layman's terms, is due to light bouncing around in a scene. Depending on objects' shapes and materials, GI can give rise to all sorts of visual effects some of which are shown here. In this course, the term GI refers to the smooth diffuse inter-reflections shown in a somewhat contrived example on the left, where the reddish tint on the white dragon is due to the light diffusely reflected from the red book next to it.

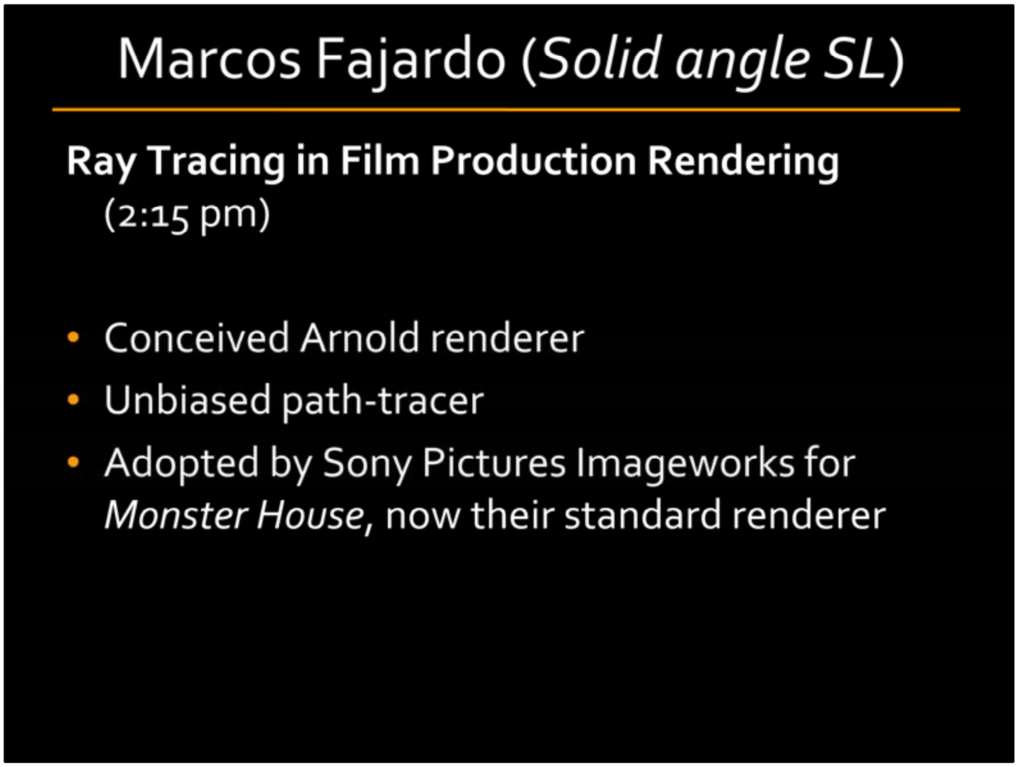
A less contrived example of global illumination, due to Michael Bunnell, is shown here. The diffuse inter-reflections provide subtle details in illumination that may often go unnoticed, but without which the image would look strange: smooth illumination gradients, colors in shadows, and color-bleeding.



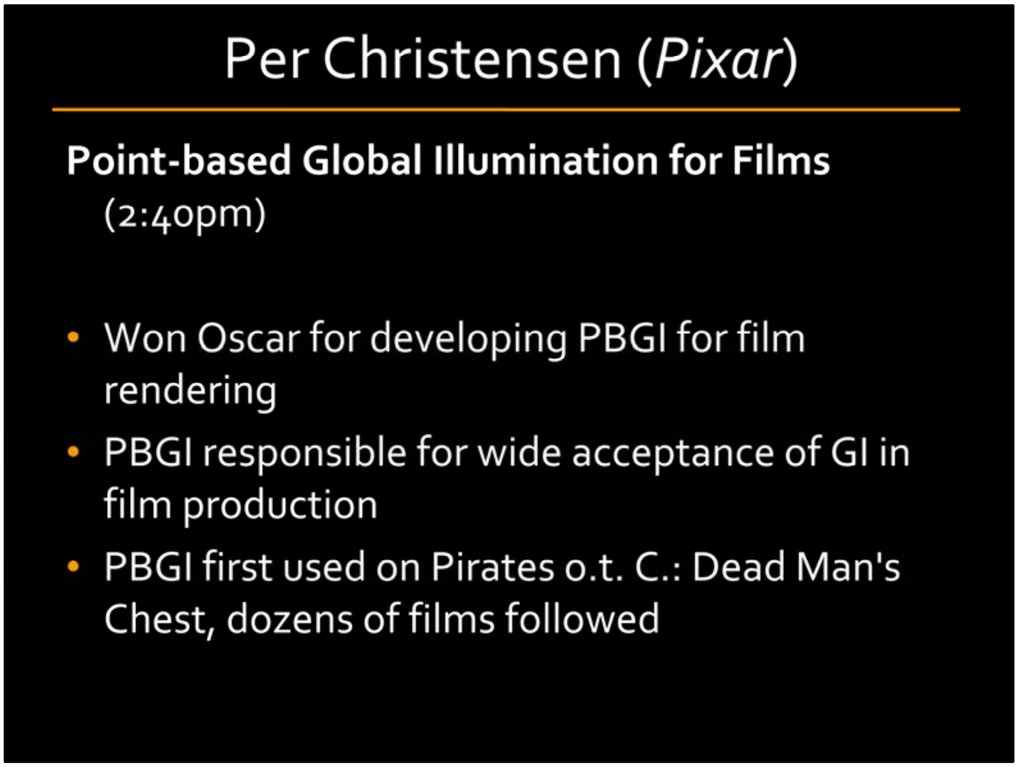
The course title is “Global illumination across industries.” Which industries are we talking about? Essentially any application that uses CG rendering also uses global illumination. This course, however, focuses on two specific areas -- film production and video games.



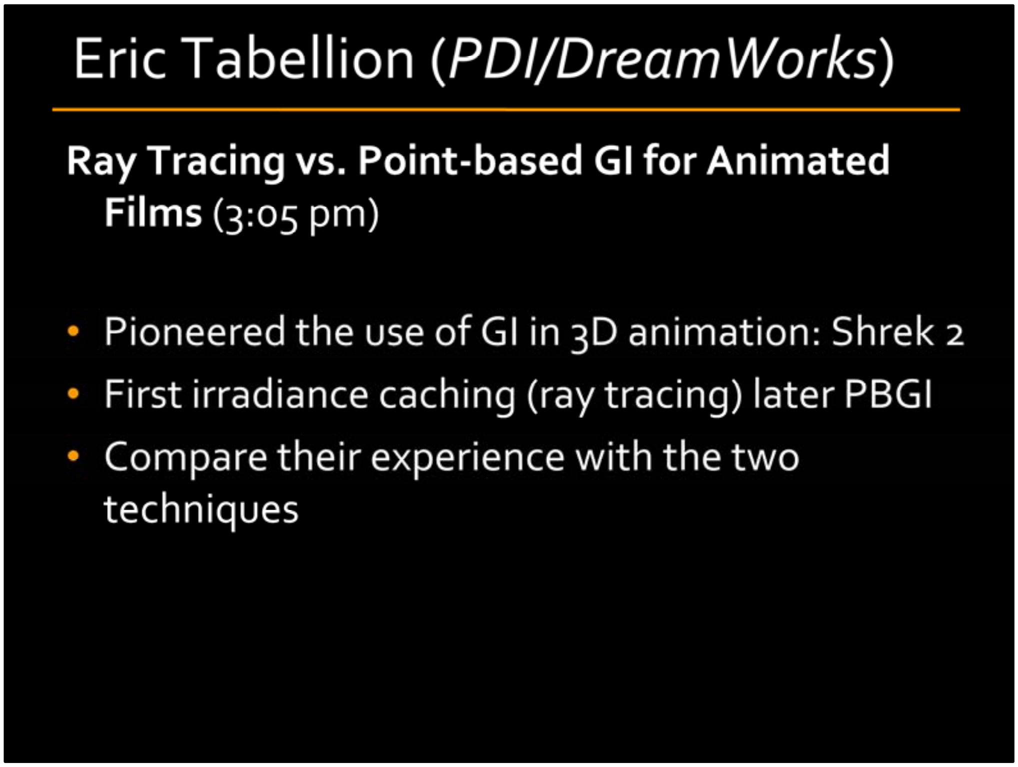
The course offers a representative sampling of GI techniques used in film and games. We'll be focusing specifics and comparisons. There won't be much theory, so nothing to worry about. But most importantly, the information is delivered by the most qualified speakers.



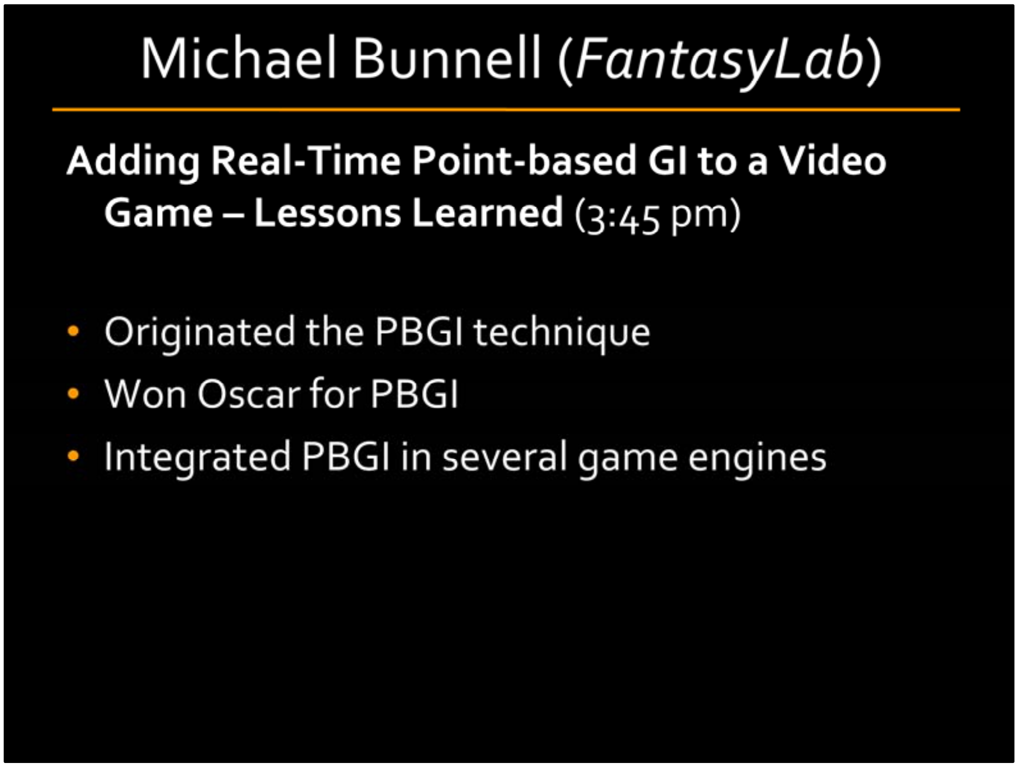
Our first speaker is Marcos Fajardo, the main developer of the Arnold renderer, an unbiased path tracer adopted by Sony Pictures ImageWorks for rendering of all of their films.



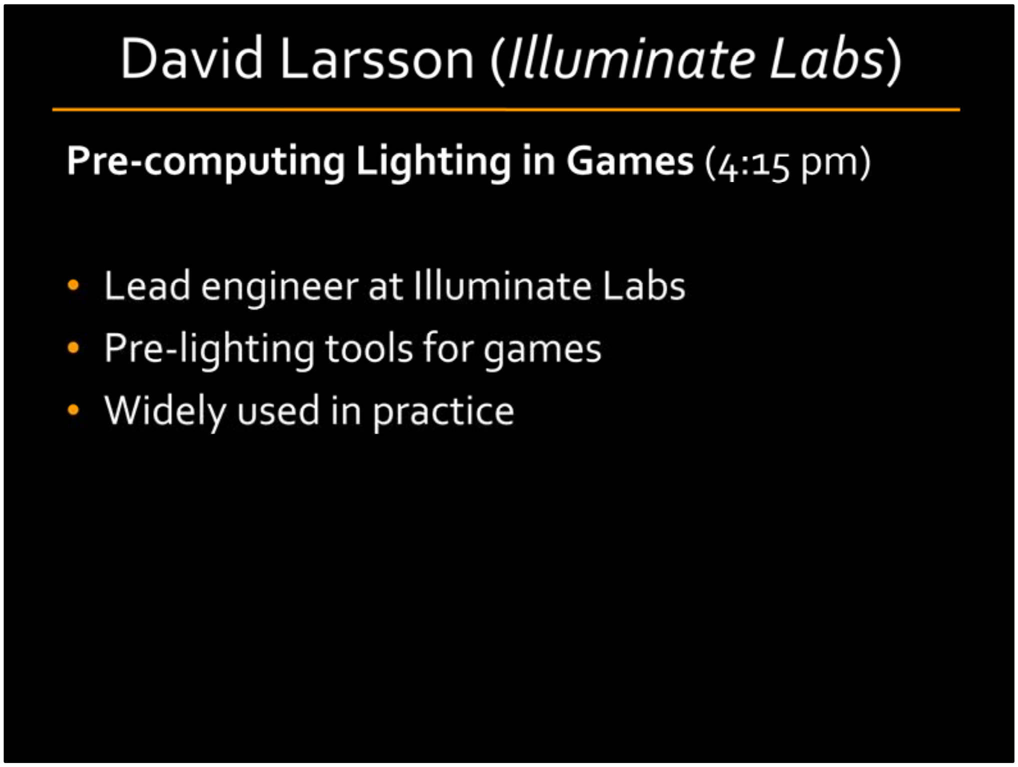
Next, we have Per Christensen from Pixar. Per won an Academy Award for developing the point-based global illumination technique which he will talk about in his contribution.



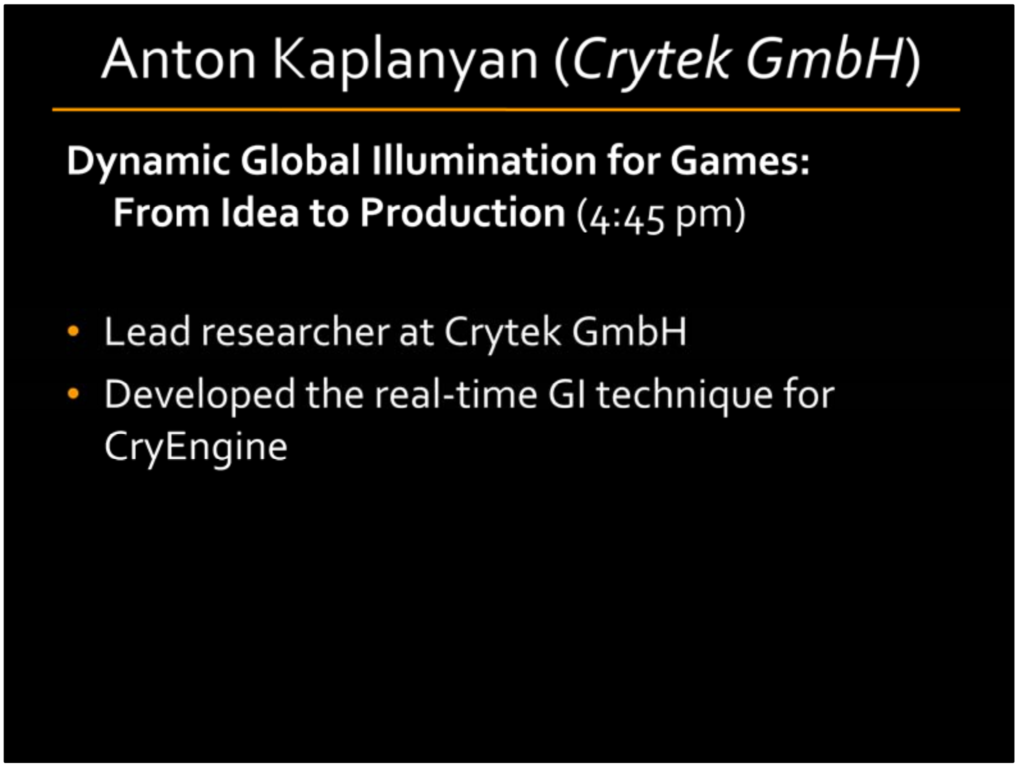
Our next speaker is Eric Tabellion from PDI/DreamWorks. Eric pioneered the use of GI in feature animation on Shrek 2. He will share his experience with ray traced and point based global illumination.



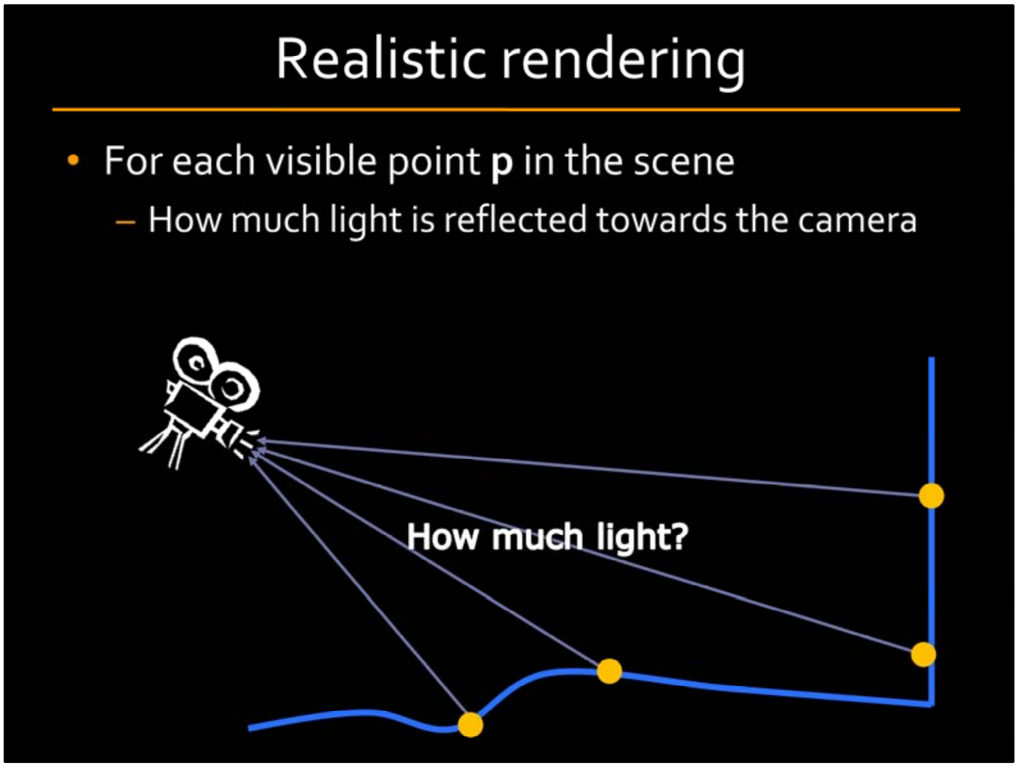
Moving on to the real-time domain, Micheal Bunnell will be our next speaker. Michael originated the point-based GI and was also awarded an Academy Award for his this work. He will share his experience with including PBGI in game engines.



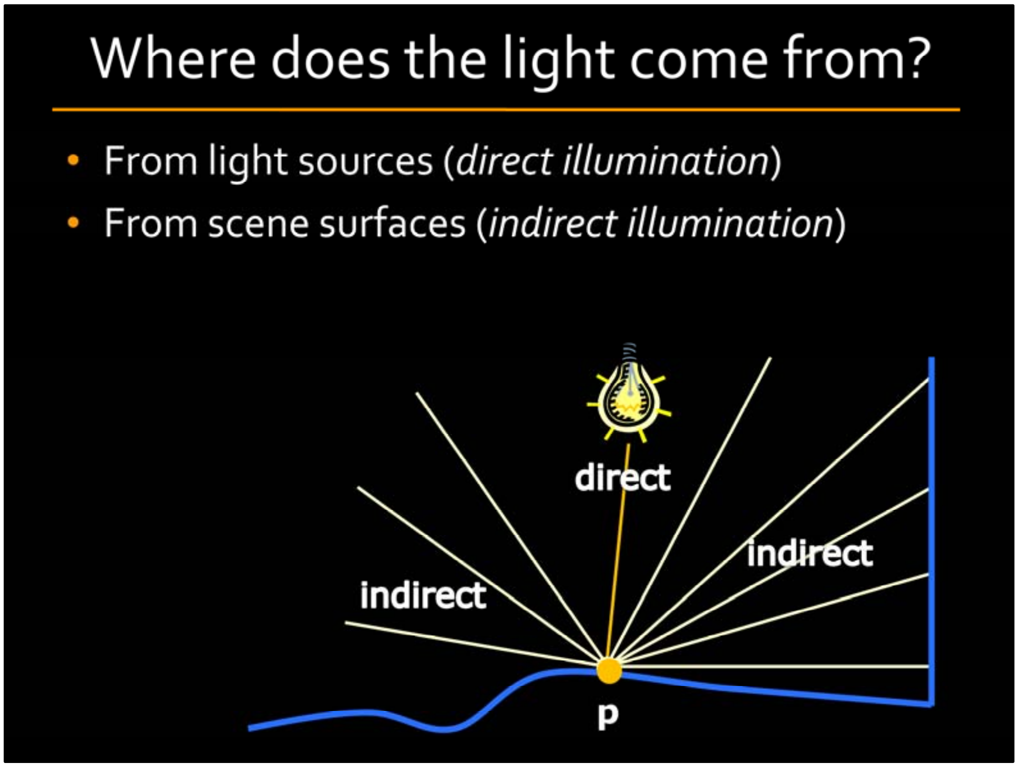
Next, David Larsson, the lead engineer in Illuminate Labs, now Autodesk, will talk about pre-computed lighting in games.



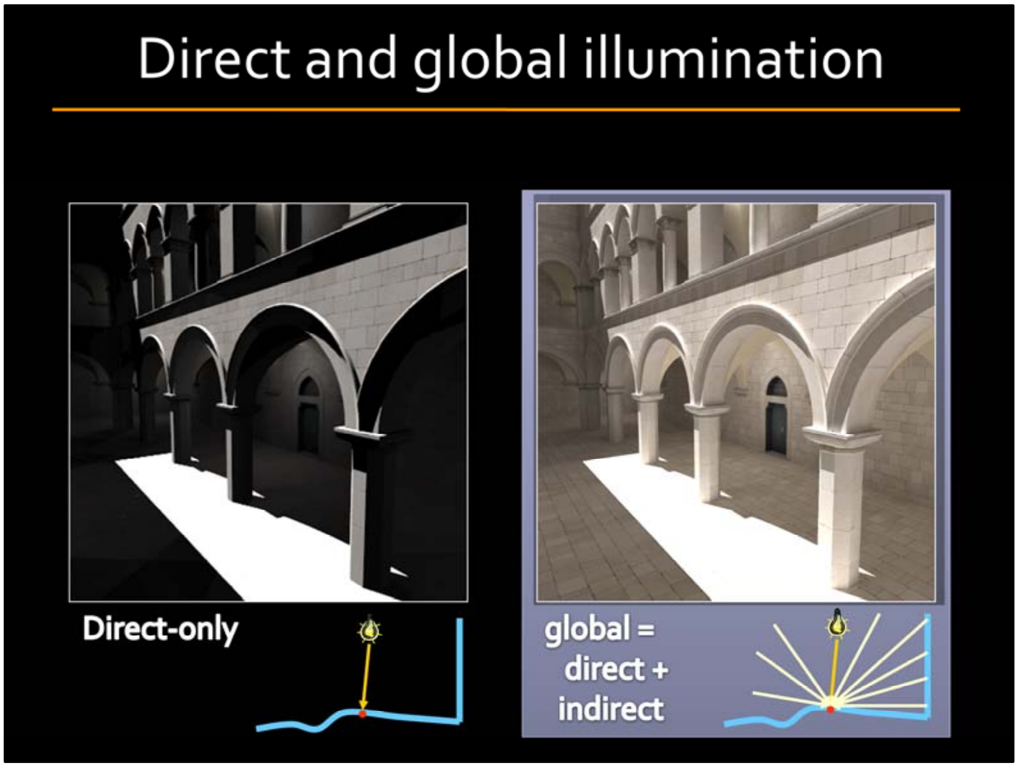
And finally, Anton Kaplanyan, the lead researcher at Crytek GmbH will describe the real-time GI techique that he developed for Crytek's CryEngine.



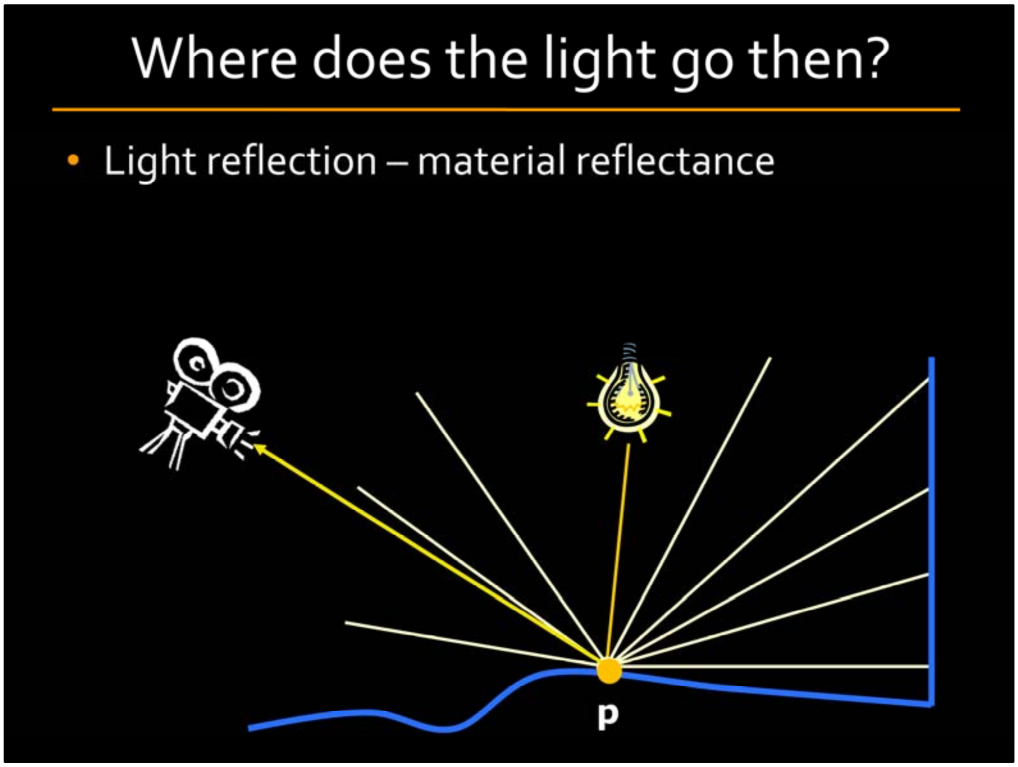
Let us now briefly review some basic terms of GI computation. The goal of realistic rendering is to compute the amount of light reflected from visible scene surfaces that arrives to the virtual camera through image pixels. This light determines the color of image pixels.



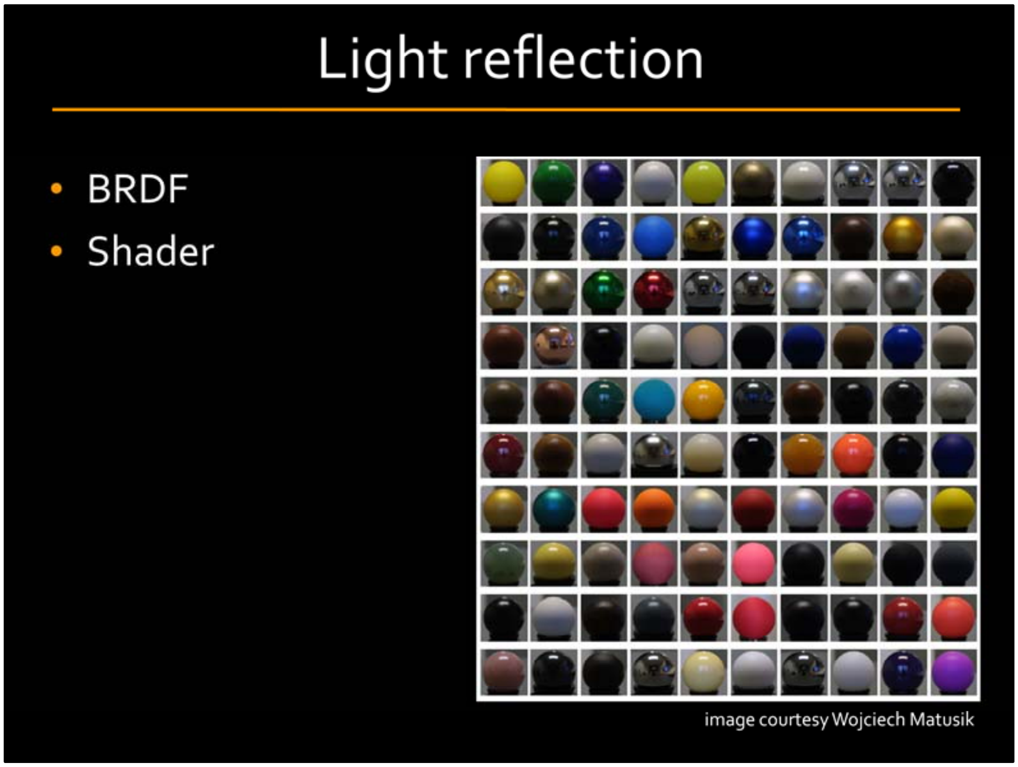
Consider one such point, **p**. Where does the light at **p** arrive from? Surely, it can come directly from the light sources – this is called the *direct illumination*. But light can also arrive from any scene surface, after being reflected several times. This is the *indirect illumination*.



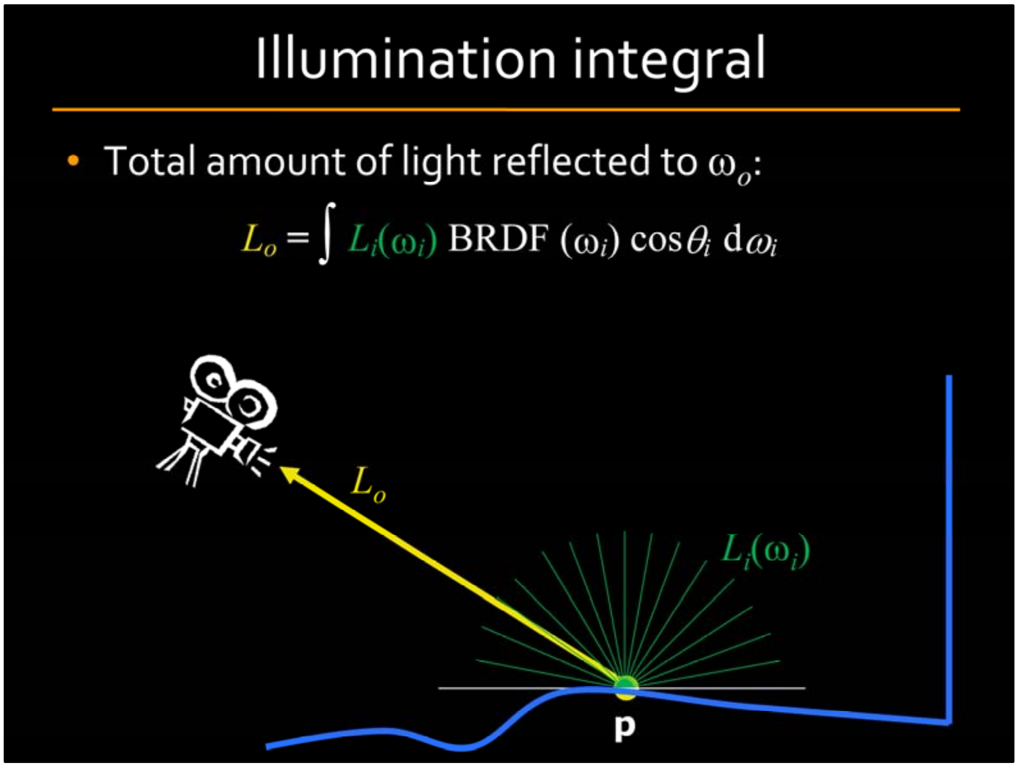
On the left is an image generated by taking into account only direct illumination. On the right is the result of global illumination computation where light is allowed to bounce around in the scene before it is finally reflected towards the camera.



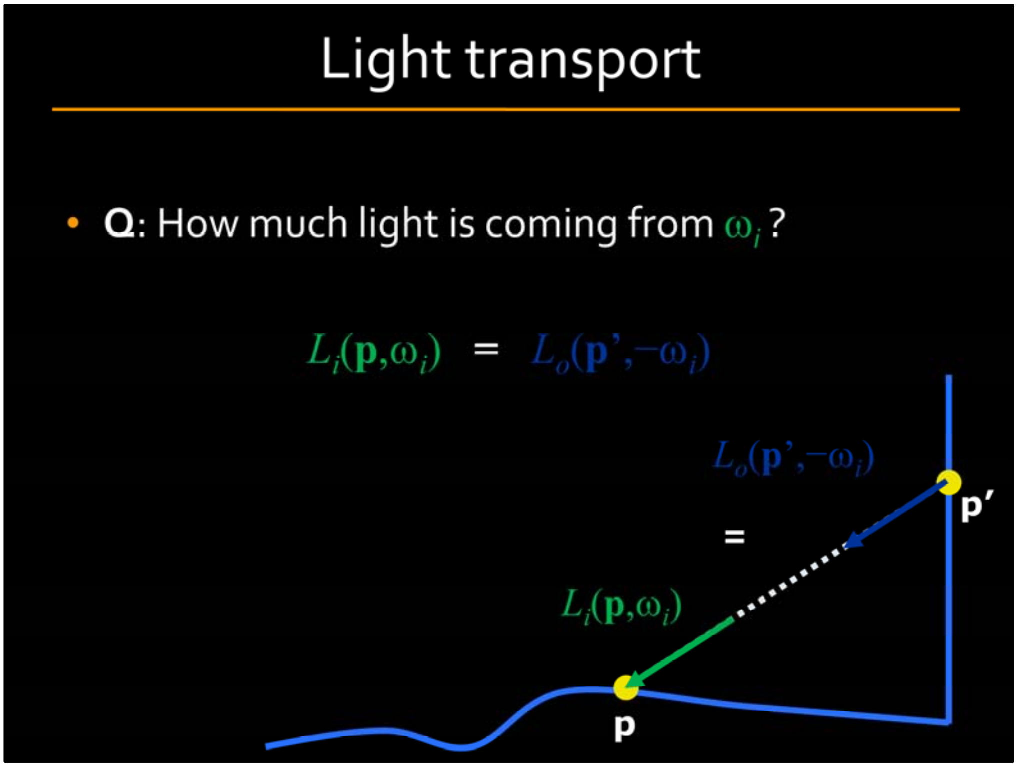
Now we know where the light comes from at **p**. But we want to compute how much of it is reflected towards the camera. This is determined by the material reflectance properties.



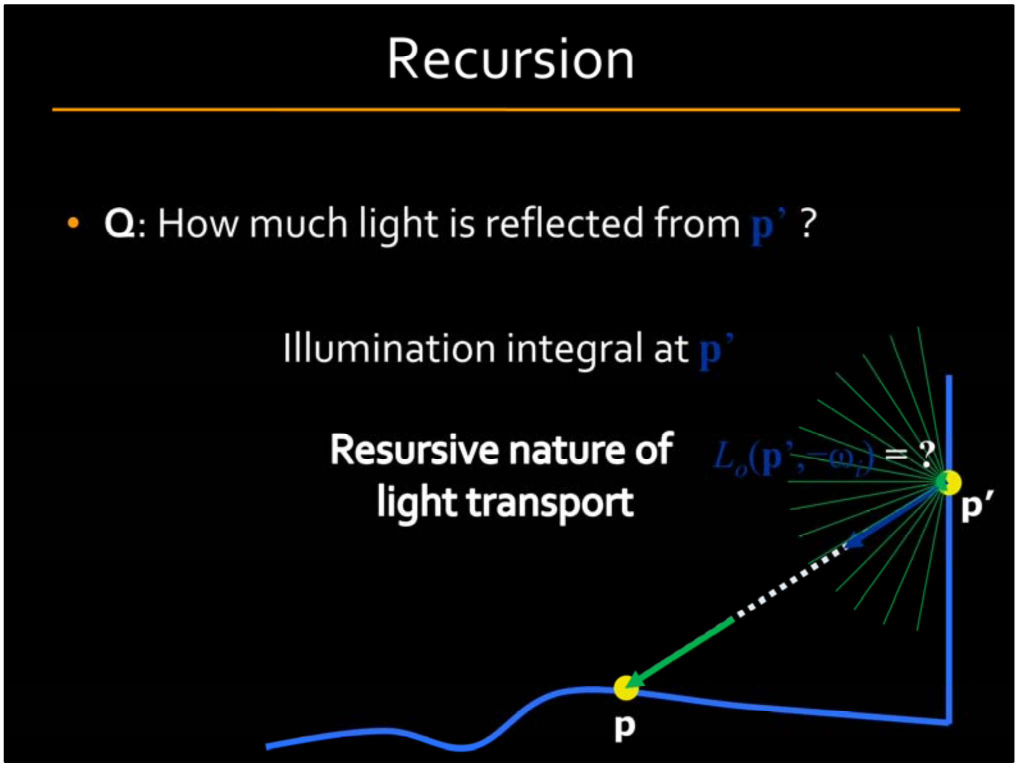
The material reflectance determines the surface’s response to incoming light. The spheres on the right are all illuminated the same – their varying appearance is entirely determined by their different material properties. A matmatical description of reflectance characteristics at a point is the BRDF – bidirectional reflectance distribution function.



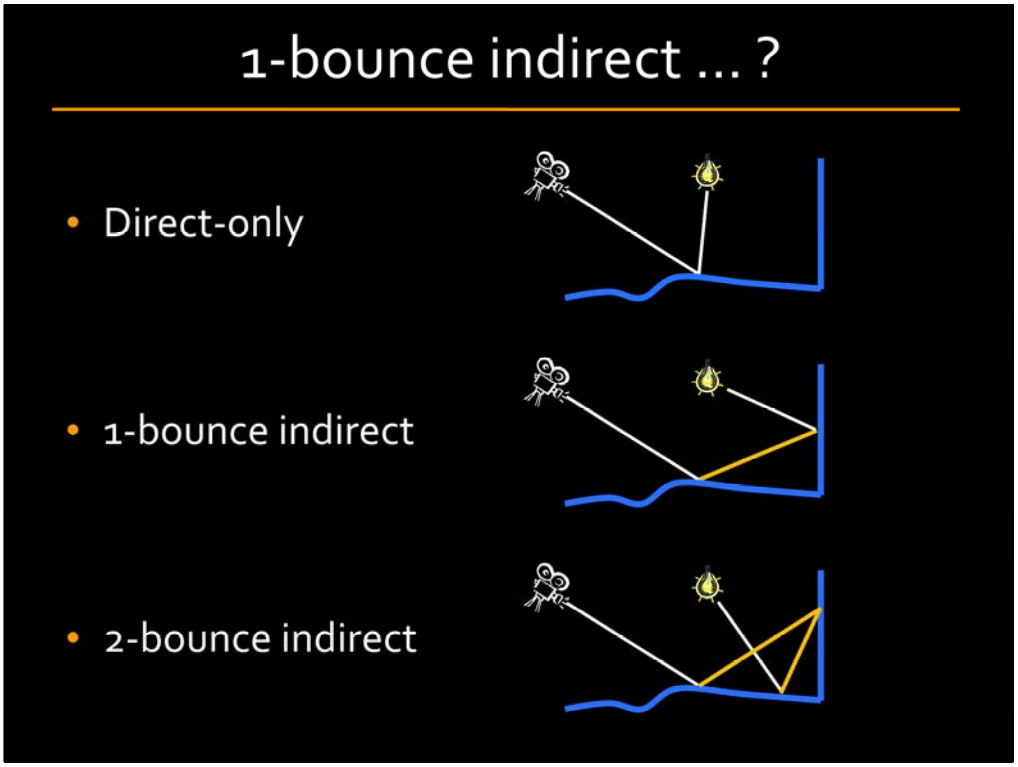
We want to compute the amount of light going from **p** towards the camera. Assuming we know how much light is coming to **p** from all possible directions (the incoming hemisphere), the reflected light can be computed by evaluating the illumination integral: For each incoming direction, we multiply the radiance from that direction by the BRDF and the cosine term. To get the total amount of reflected light, we sum (integrate) these contributions over all incoming directions. Then we add the self emitted light at **p** (in case **p** is on the light source) and we obtain the total amount of light going from **p** towards the camera.



We made an important assumption in computing the illumination integral. We assumed that we know how much light is coming to **p** from each direction. But how do we find this out? Taking advantage of the fact that *radiance does not change along straight lines*, we see that the incoming radiance from direction w_i is equal to the outgoing radiance at a point **p**' visible from **p** along w_i.

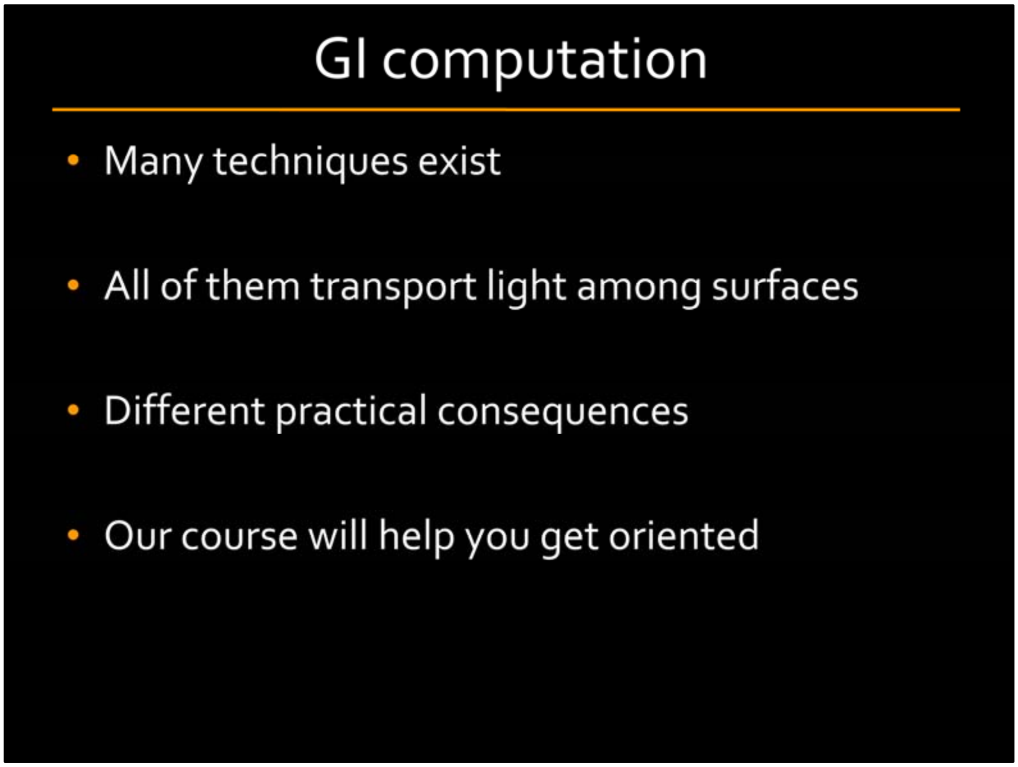


We have transformed the question of “how much light is coming to **p** from some direction” into “how much light is leaving **p’** into the opposite direction”. To answer the latter question, we apply the illumination integral once again at **p’**, in a recursive fashion. We can see that light transport is a recursive procedure, where each recursion level corresponds to one reflection of light.



In theory light can be reflected an infinite number of times on its way from the light source to the camera. However, due to light absorption, it is rarely useful to compute more than four bounces (reflections). One indirect bounce is often sufficient to capture the visual qualities of GI (smooth gradients, color bleeding, contact shadows).

The slides explain what exactly does the "one-bounce-indirect" jargon mean. Light paths with one single reflection correspond to direct illumination. "One-bounce-indirect" corresponds to light paths with two reflections (one "direct" and one "indirect"). And so forth for "two-bounce indirect".



There is a plethora of techniques for GI computation. The purpose of all of them is to simulate (or approximate in a visually plausible manner) the inter-reflections of light between surfaces. These techniques have different advantages and disadvantages that make them more or less suitable for specific applications. The very purpose of this course is to summarize some of the most important GI computation techniques used in film production and video games.