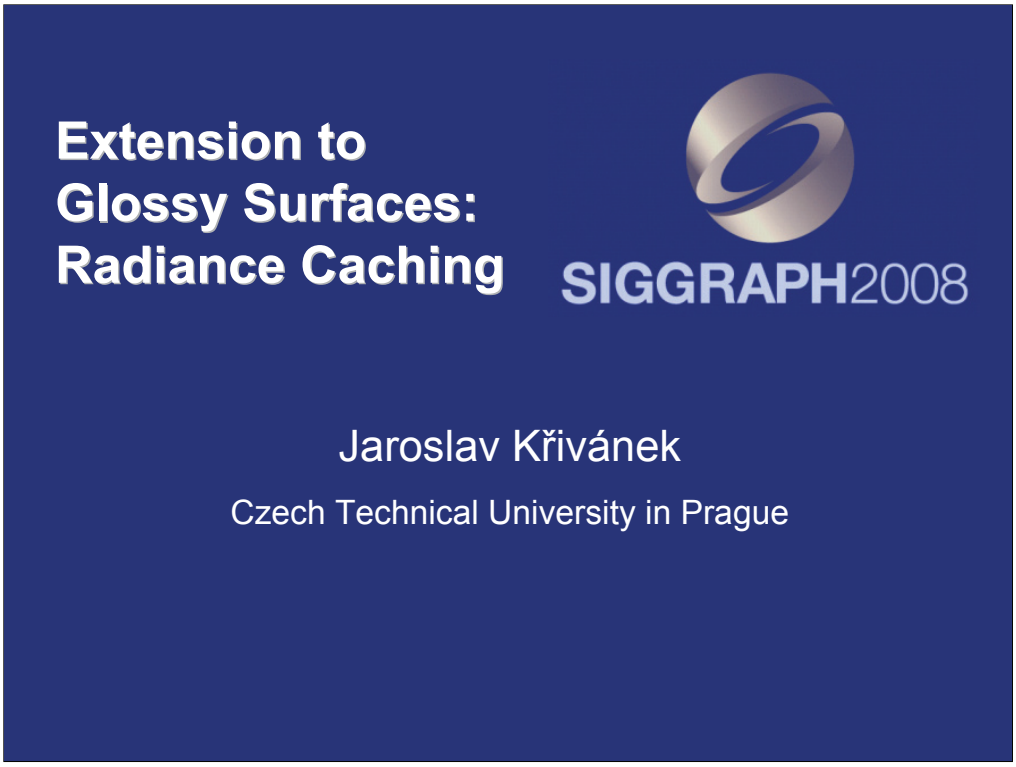
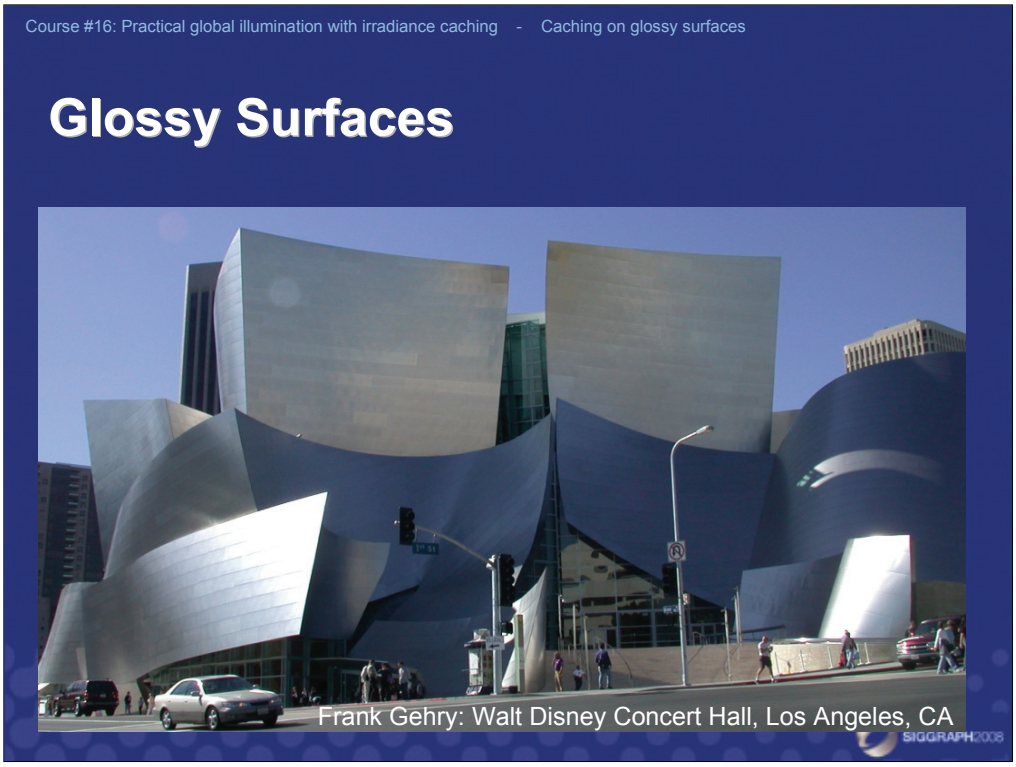
SIGGRAPH2008

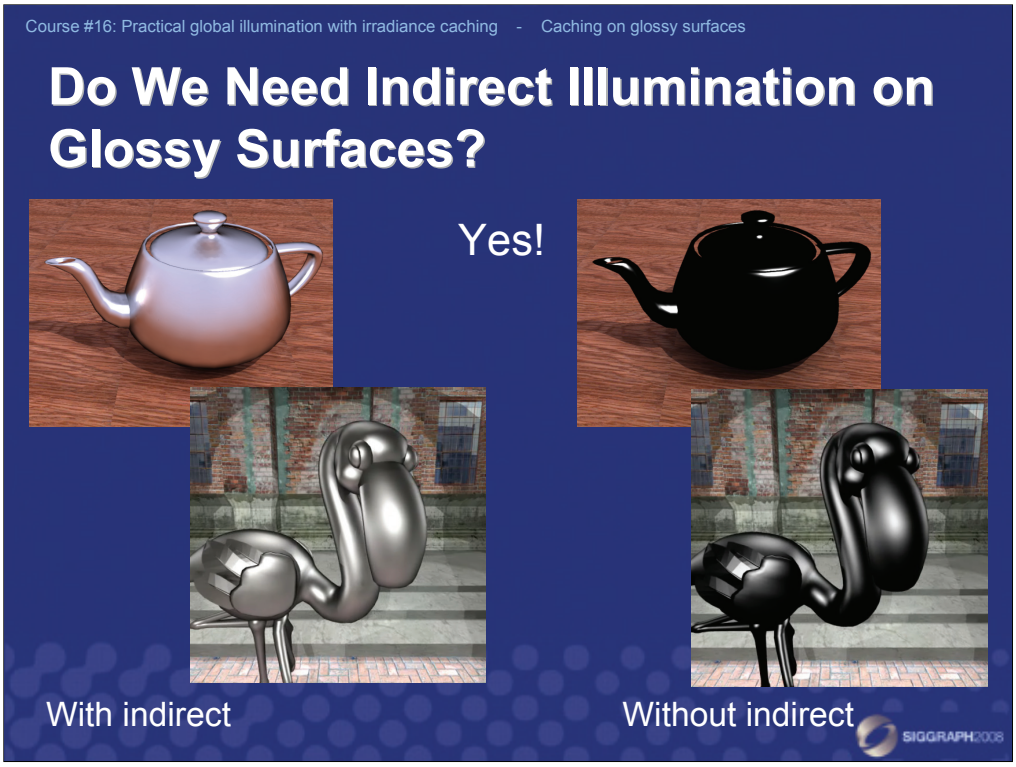


The interpolation scheme used in irradiance caching for diffuse surfaces can be also used on glossy surfaces. However, some modifications are necessary due to the view-dependence of reflection on glossy surfaces. These modifications will be discussed in this part of the course.

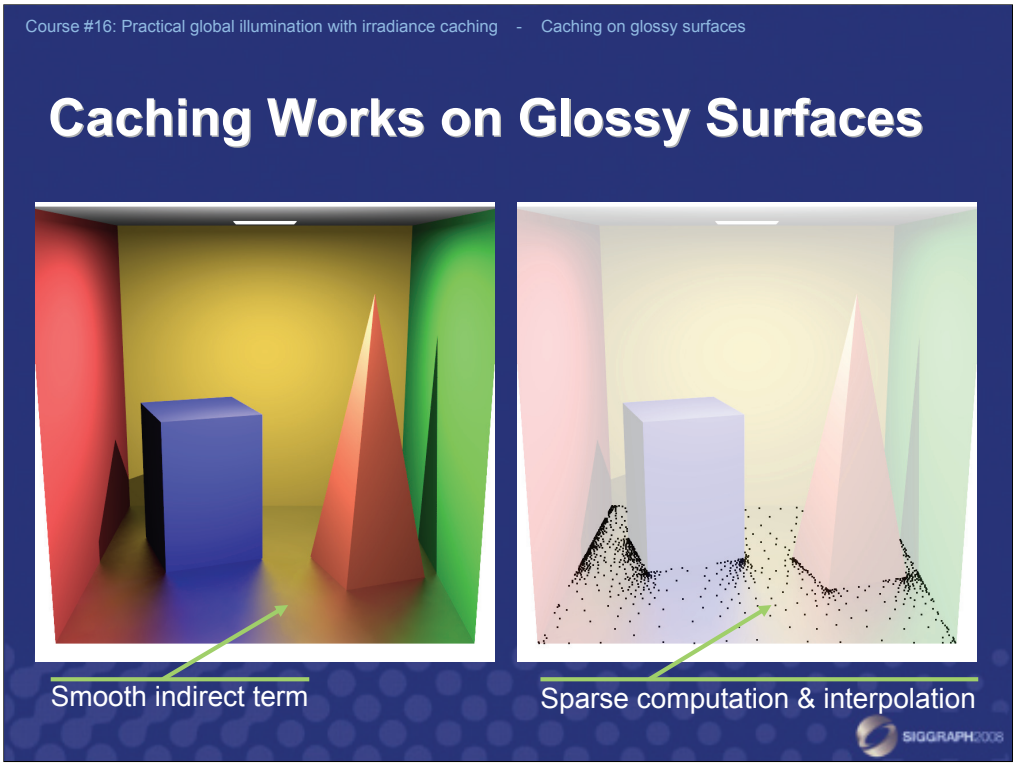


A nice real-world example of glossy surfaces are the walls of Frank Gehry's Walt Disney Concert Hall in Los Angeles (as well as Gehry's Guggenheim Museum in Bilbao, Spain). Other common examples of glossy surfaces include rough plastic surfaces or rough pottery.

What do we exactly mean by 'glossy surface' for the purpose of this talk? We are referring to rough surfaces that reflect their environment but the reflections are blurry. This is in contrast to 'specular' surfaces that show sharp reflections of their environment. Glossy surfaces have 'low-frequency' BRDFs while specular surfaces have 'high-frequency' BRDFs. Indirect illumination on specular surfaces can be quite efficiently computed by stochastic ray tracing with importance sampling (proportional to the BRDF) using only a couple of secondary rays. In what follows, we will focus on computing indirect illumination on glossy surfaces (with low-frequency BRDFs), for which importance sampling is not very effective.

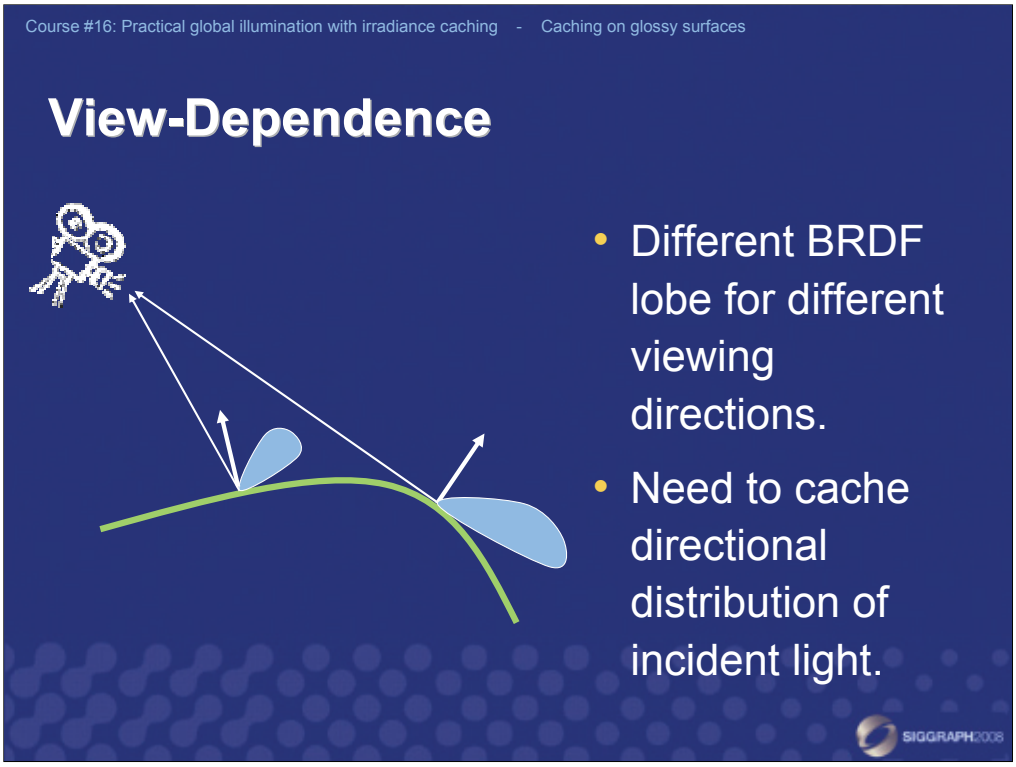


Why do we bother computing indirect illumination on glossy surfaces? Is it worth the effort? Yes, it certainly is. Actually, Fleming et al. [2003] have shown that correct perception of material properties depends very much on the illumination. Under a point light, we don't really know what an object is made of, while under natural illumination, it is much easier to recognize the object's material.

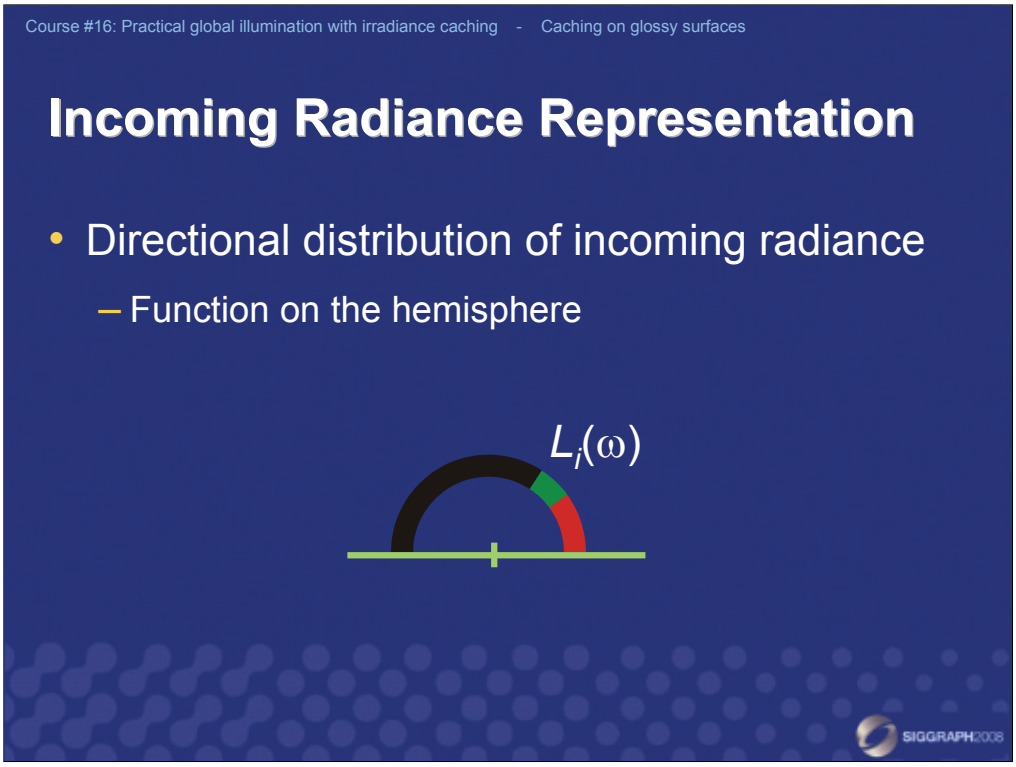


Fortunately, glossy surfaces show a very blurry reflections of their environment. In other words, indirect illumination on glossy surfaces changes slowly, similar to diffuse ones. This allows to use the same trick as in irradiance caching – compute illumination lazily at sparse locations in the scene and interpolate elsewhere.

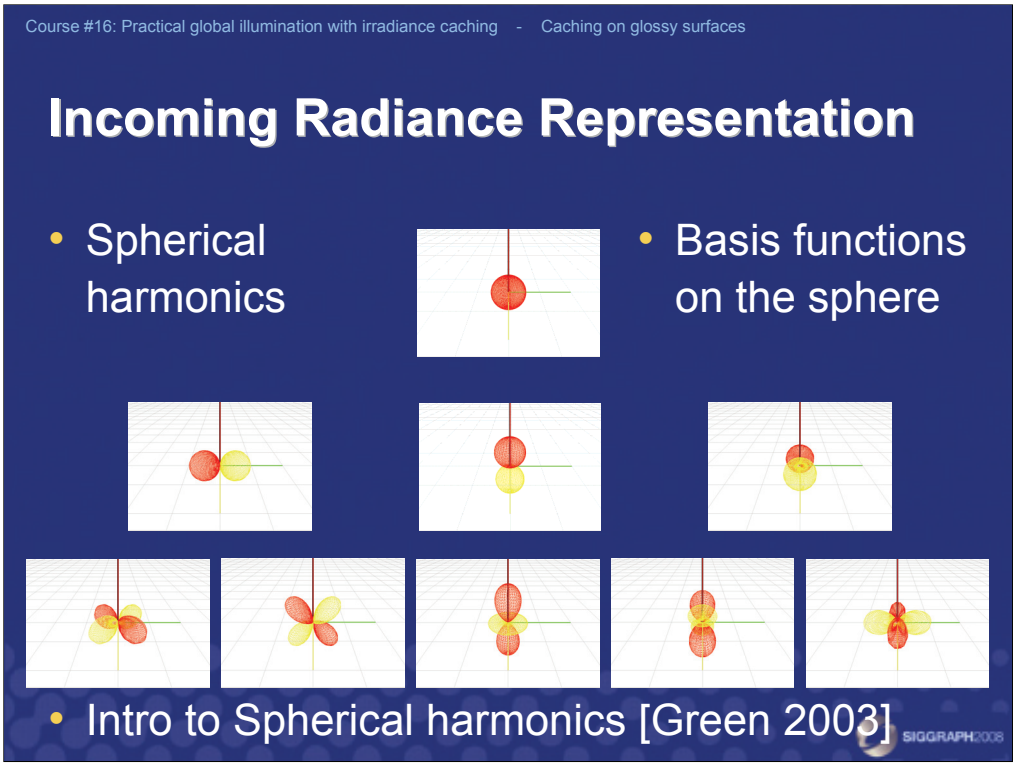
Unlike for diffuse surfaces, appearance of glossy surfaces is view-dependent. In other words, the surface looks different when observed from different viewing angles. This makes interpolation of illumination more complicated than on purely diffuse surfaces.



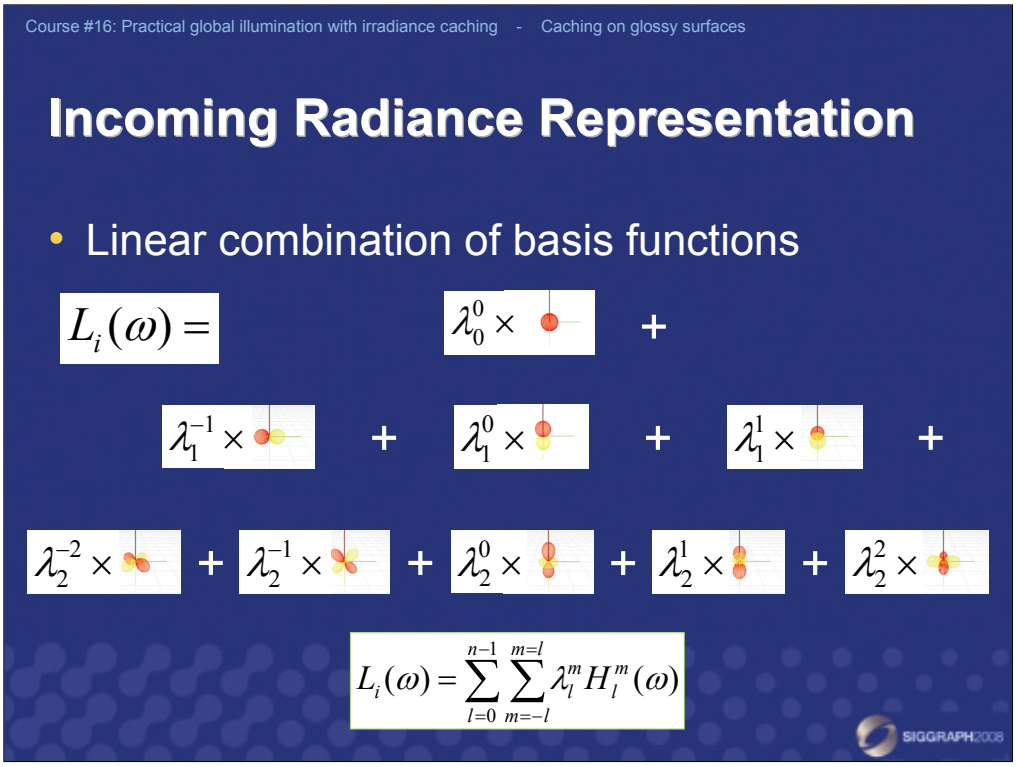
The view-dependence of glossy surfaces is due to the fact that different part of incident light is reflected towards the camera for different viewing directions. So if we want to interpolate illumination on glossy surfaces, we need to cache the full directional distribution of incoming radiance – i.e. we cache a representation of incoming radiance at a point for all possible directions on the hemisphere. Then, at each interpolation point, we 'extract' the part of the incoming light which is reflected towards the camera. This 'extraction' is done by evaluating the illumination integral, i.e. by integrating incoming light multiplied by the BRDF lobe over the hemisphere. This will be discussed in more detail later.



The cached quantity – directional distribution of incoming radiance - is a function defined on the hemisphere. We need to find a suitable representation of this function to make caching possible.



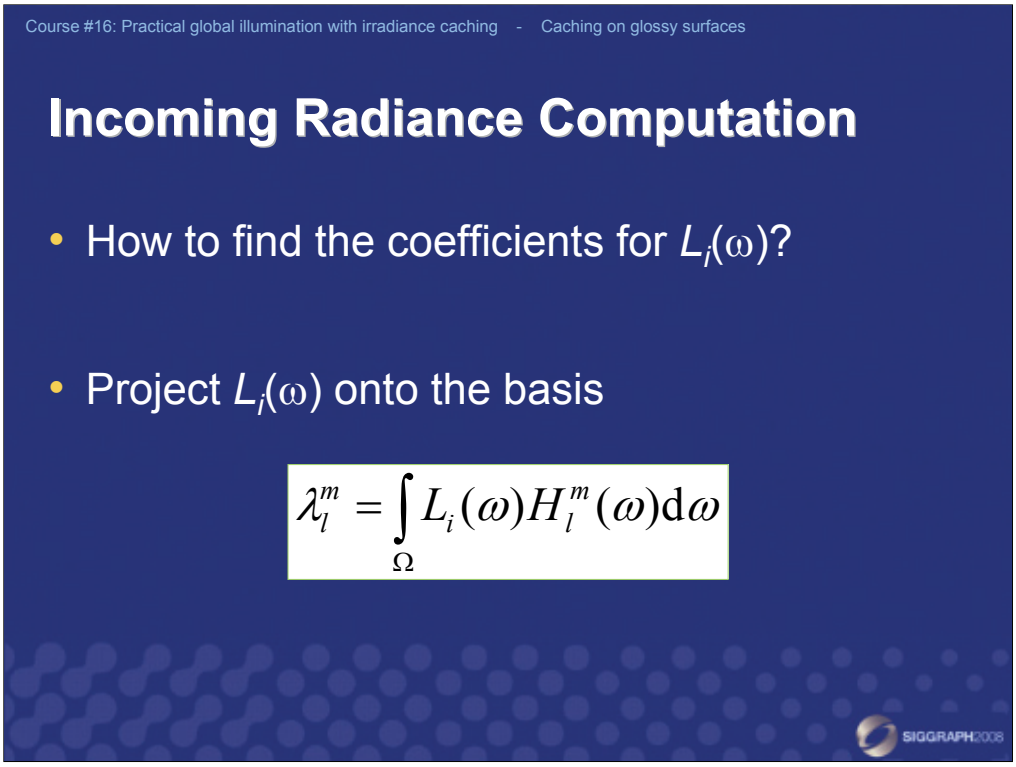
For various reasons that will become clearer later, we use spherical harmonics to represent the incoming radiance at a point. Spherical harmonics are basis functions defined on the unit sphere.



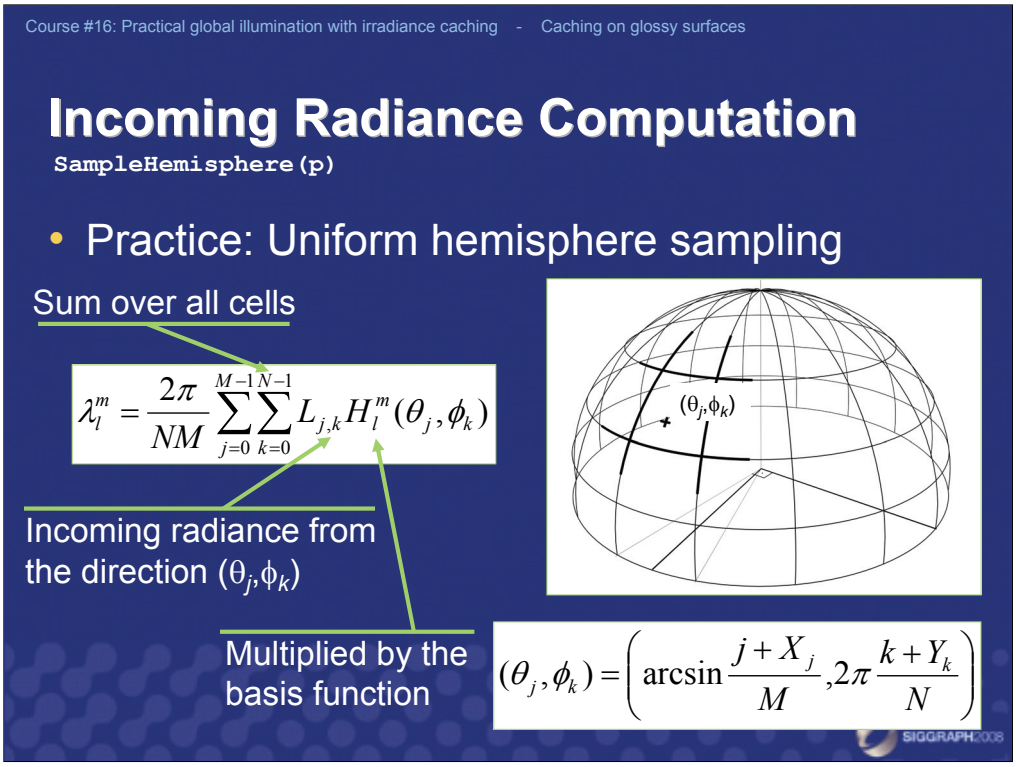
Incoming radiance, as well as any other function defined on the hemisphere, can be represented as a linear combination of the basis functions. The coefficients in this linear combination make up the representation of our function.

For spherical harmonics, the coefficients are indexed by two indices, *l* (lower index) and *m* (upper index). The *l*-index runs from 0 to *n*-1, where *n* is the order of the spherical harmonics representation. The higher the order, the better (more accurate) representation. Spherical harmonics with the same *l*-index form a band (a row on the slide above). Within one band, the *m* index goes from –*l* to +*l*. So there is one harmonic in the first band, three harmonics in the second band, five in the third band etc. There are n^2 coefficients in total for an order-*n* representation. The double sum in the formula on the slide simply stands for summing over all harmonics up to order *n*. It could also be expressed as a single sum for *i* from 0 to n^2-1, with *i* given by : $i = l(l+1) + m$.

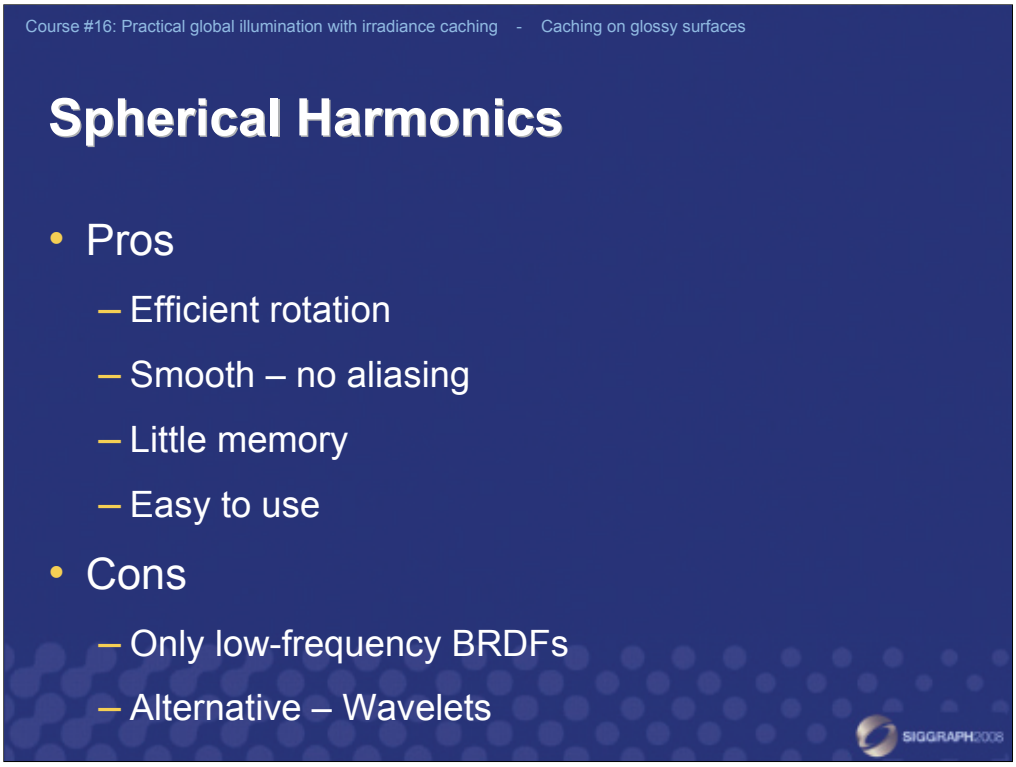
To summarize, an order-*n* representation of a function consists of n^2 coefficients, which is what we to store in the cache. (Actually, we store one coefficient vector for each color component, R, G, and B). Typically, we use order up to 10 in radiance caching, corresponding to 100 coefficients.

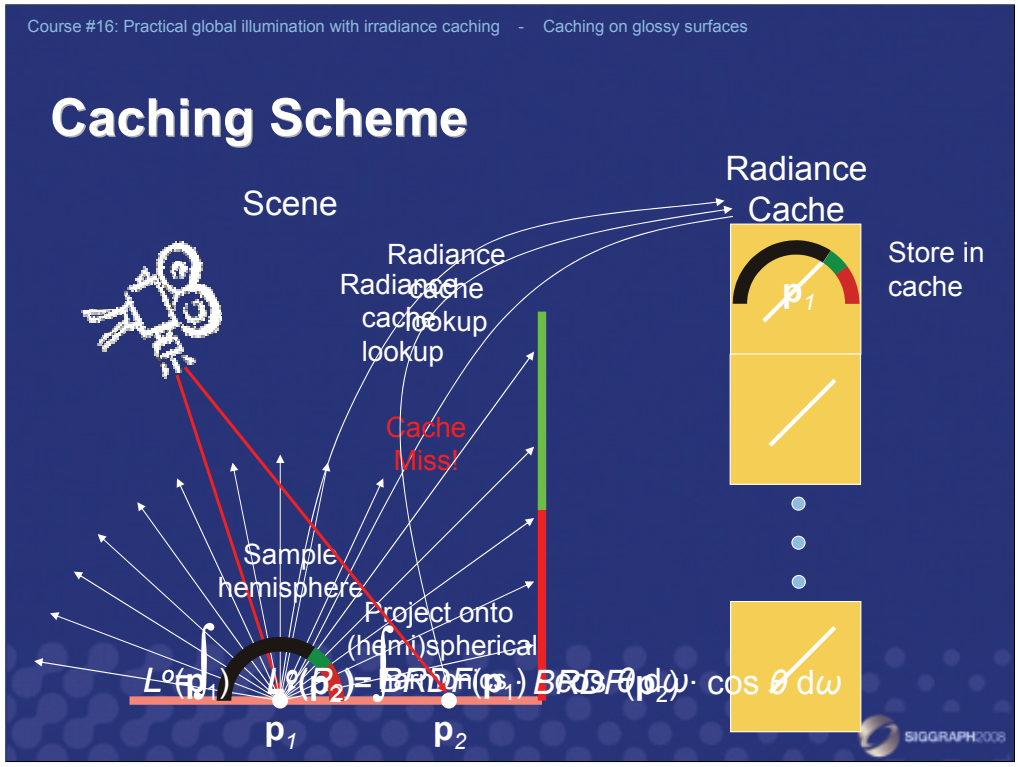


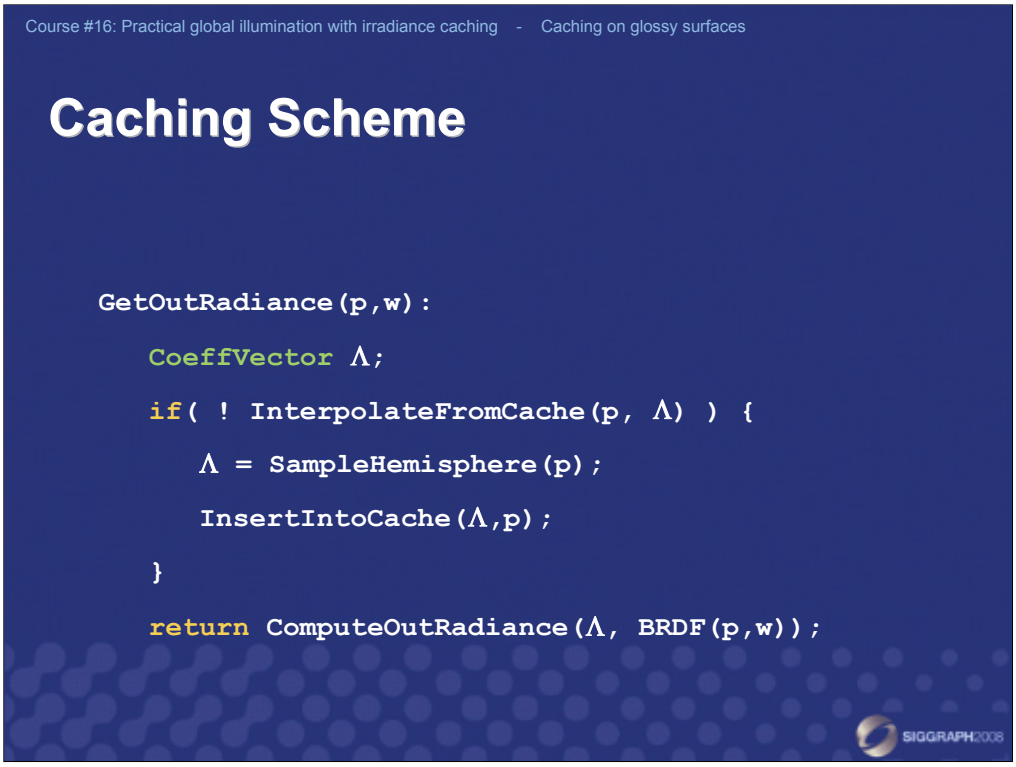
The coefficients representing a function are found by *projection*. For our purposes, it is sufficient to say that the projection is computed by integrating the product of the function being projected with the basis function, as shown on the slide.



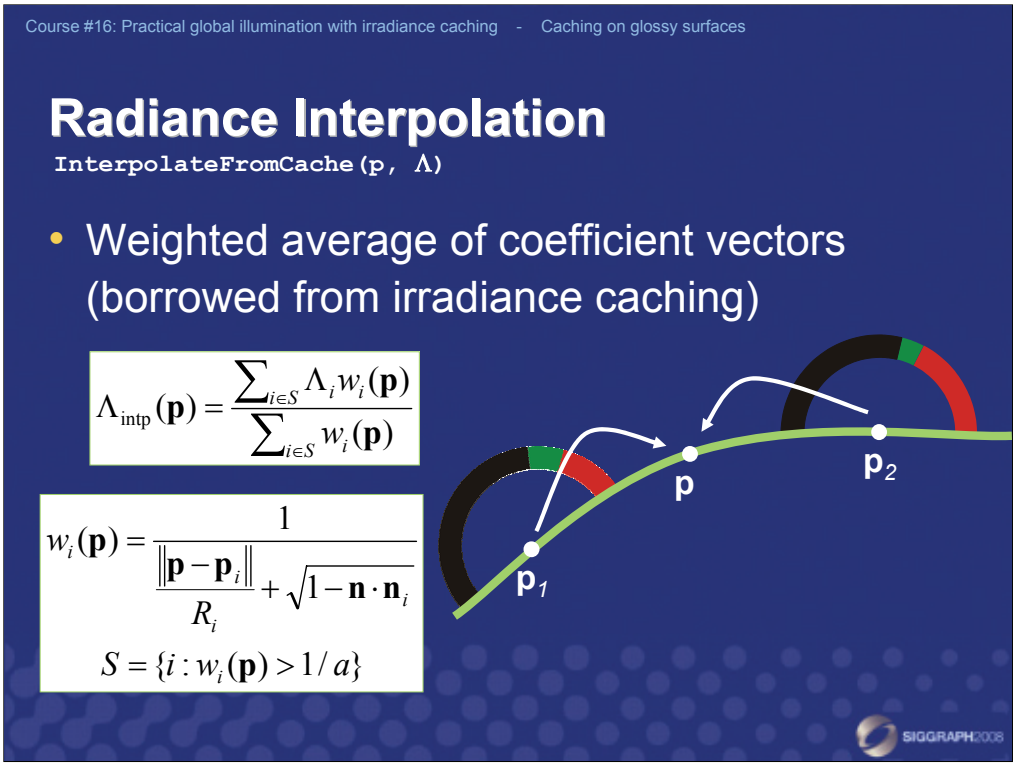
To find the coefficients for the incoming radiance at a point, we need to numerically estimate the projection integral. Since our only means to evaluate the incoming radiance is point sampling – i.e. casting secondary rays – the obvious choice is Monte Carlo quadrature. We divide the hemisphere *uniformly* into *M* x *N* cells, where *M*.*N* is the desired number of rays (we use 500 – 8000 our renderings), and $N \approx 4M$. For each cell [j,k] we pick a random direction using the formula above. Note that this is slightly different from irradiance caching, since we are using uniform sampling (as opposed to cosine-proportional sampling in irradiance caching). For each sampling direction, we evaluate the basis functions, multiply them by the incoming radiance from that direction and accumulate to the coefficient vector.


Course #16: Practical global illumination with irradiance caching - Caching on glossy surfaces
Spherical Harmonics
• Pros
– Efficient rotation
– Smooth – no aliasing
– Little memory
– Easy to use
• Cons
– Only low-frequency BRDFs
– Alternative – Wavelets
SIGGRAPH2008

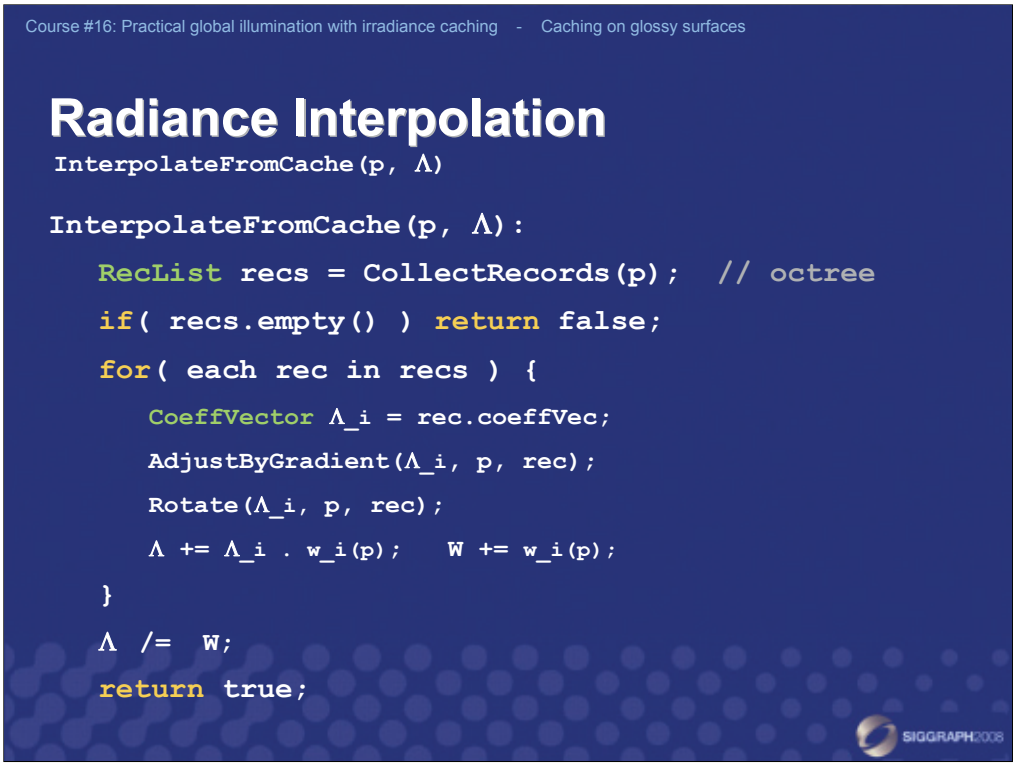

Course #16: Practical global illumination with irradiance caching - Caching on glossy surfaces
Caching Scheme
Scene
Radiance Cache
Store in cache
Radiance cache lookup
Radiance cache lookup
Cache Miss!
Sample hemisphere
Project onto (hemi)spherical
p1
p2
cos θ dω
SIGGRAPH2008



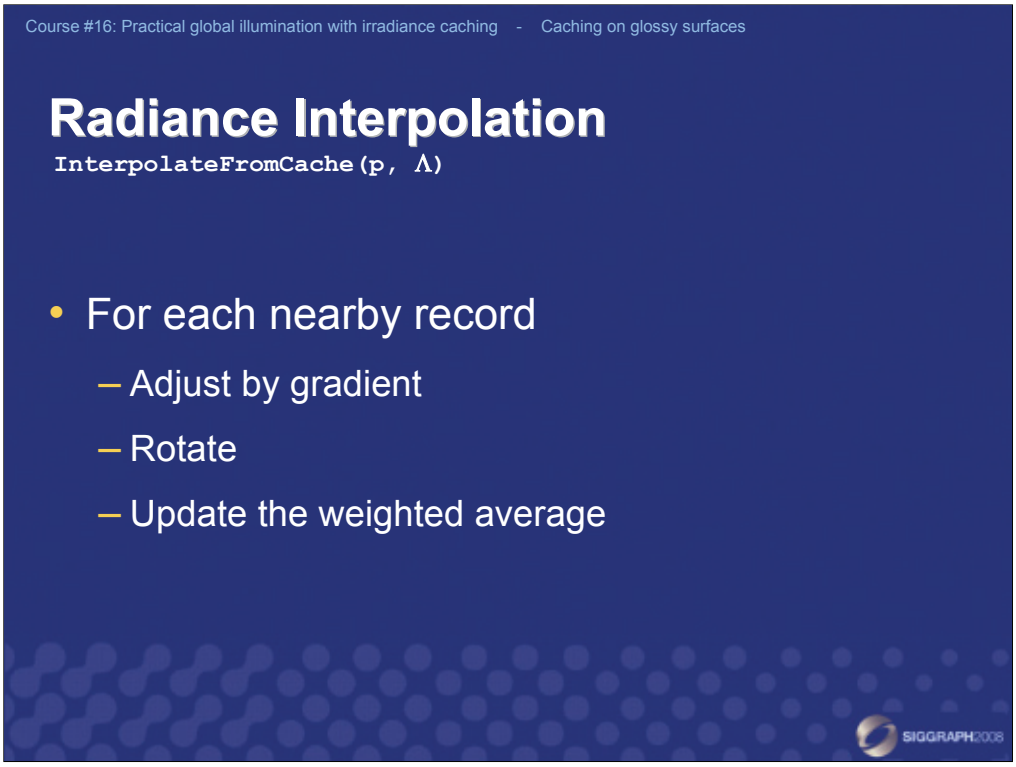
The overall caching scheme on glossy surfaces is the same as on diffuse ones. There are some important differences, though, that have to be taken care of. The procedure to compute outgoing radiance at a point due to glossy indirect illumination in a given (viewing) direction proceeds as shown on the slide. First, we query the radiance cache for previously computed records available for interpolation. If some are available, the coefficients for incoming radiance are interpolated from the cache. Otherwise, they are computed using hemisphere sampling as described on previous slides and stored in the cache. Finally, the incoming radiance is integrated against the BRDF to get the outgoing radiance. Various sub-routines of this procedure will be discussed on the following slides.

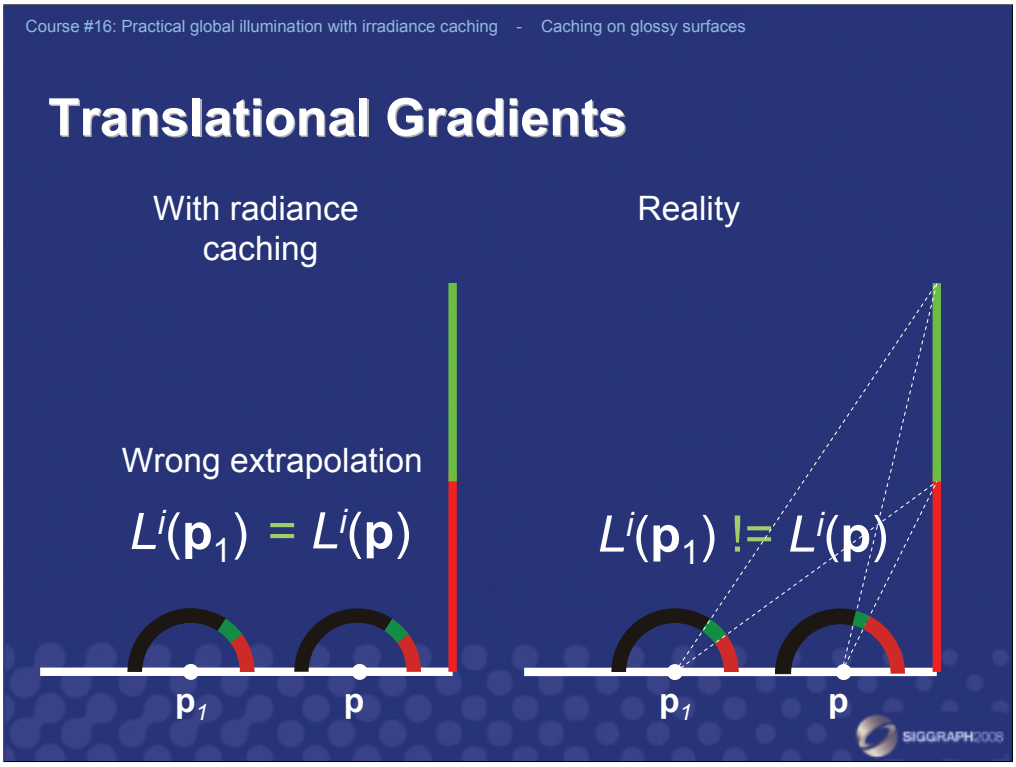


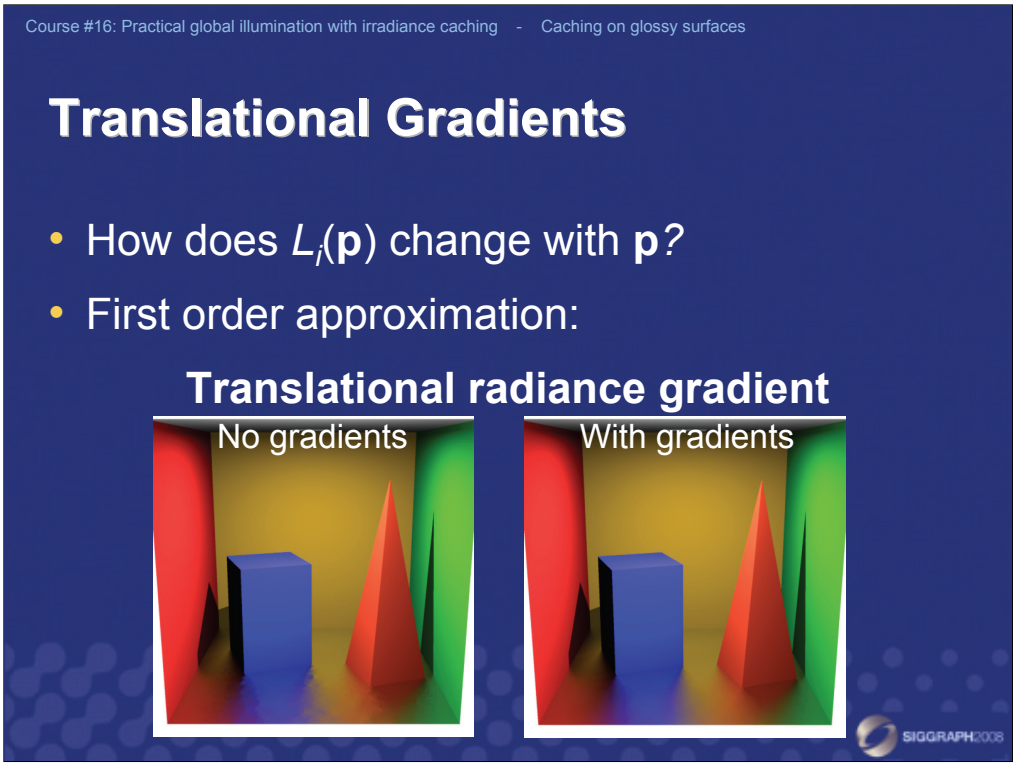
To get the coefficient of interpolated incoming radiance at a point, we use the same weighted sum as used in irradiance caching. Instead of interpolating cached irradiance values, we interpolate the coefficient vectors. However the coefficient vectors have to be adjusted by translational gradients and by rotation as described later.



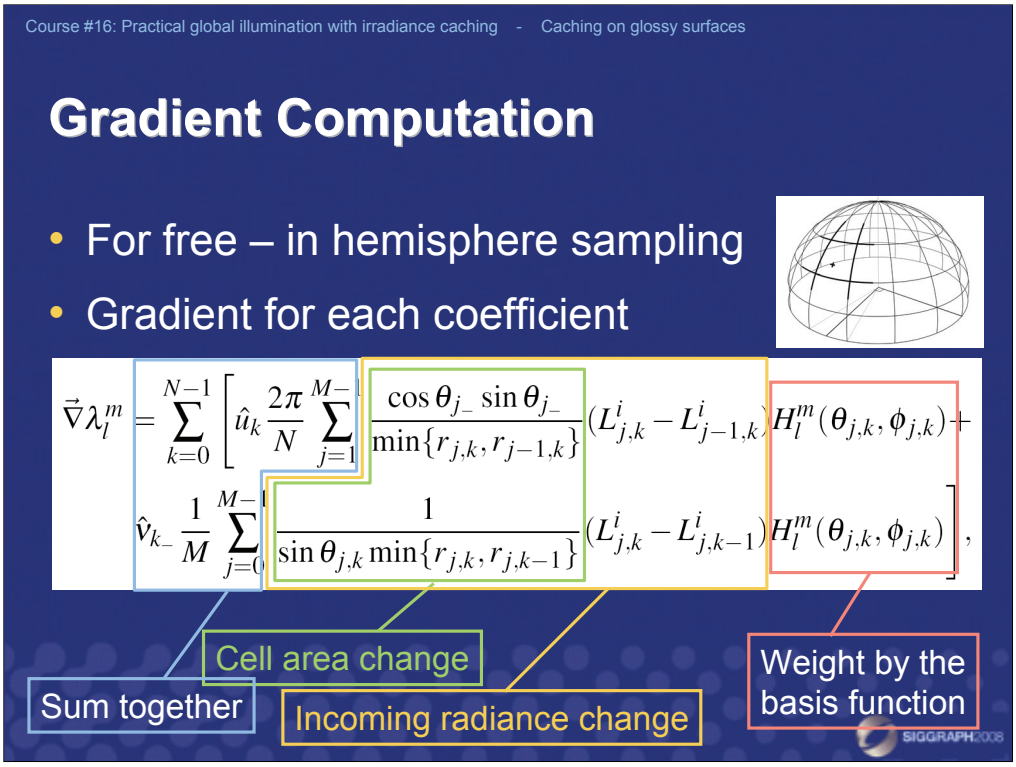
The radiance interpolation procedure starts by locating the existing records available for interpolation at **p**. This is uses the same octree structure as the irradiance cache. Then we loop over the records available for interpolation. For each record, we first adjust its coefficient vector by the translational gradient. The gradients are computed during hemisphere sampling and stored with the records in the cache. After the gradient adjustment, we apply a rotation to align coordinate frames at the point of interpolation and the location of the record.


Course #16: Practical global illumination with irradiance caching - Caching on glossy surfaces
Radiance Interpolation
InterpolateFromCache(p, Λ)
• For each nearby record
– Adjust by gradient
– Rotate
– Update the weighted average
SIGGRAPH2008


Course #16: Practical global illumination with irradiance caching - Caching on glossy surfaces
Translational Gradients
With radiance caching
Reality
Wrong extrapolation
$L^i(\mathbf{p}_1) = L^i(\mathbf{p})$
$L^i(\mathbf{p}_1) != L^i(\mathbf{p})$
$\mathbf{p}_1$
$\mathbf{p}$
$\mathbf{p}_1$
$\mathbf{p}$
SIGGRAPH2008



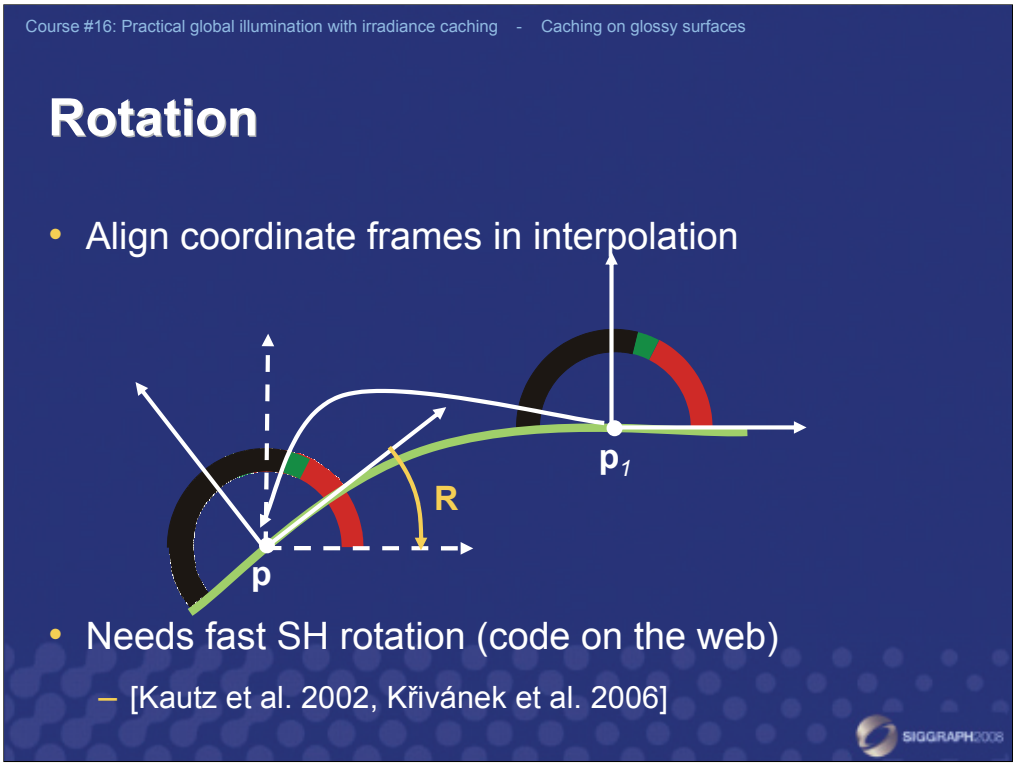
Translational gradients are used in interpolation to compensate for the change of incoming radiance with translation. If radiance is interpolated without the use of gradients, we see undesirable discontinuities in the images. The discontinuities are suppressed by the use of gradients.



The gradients represent a first-order approximation of the change of incoming radiance with translation. For each coefficient of incoming radiance, there is one translational gradient, which describes how this coefficient changes with translation. Gradients are computed in hemisphere sampling along with the coefficient vectors using a formula shown above, and stored in the record.

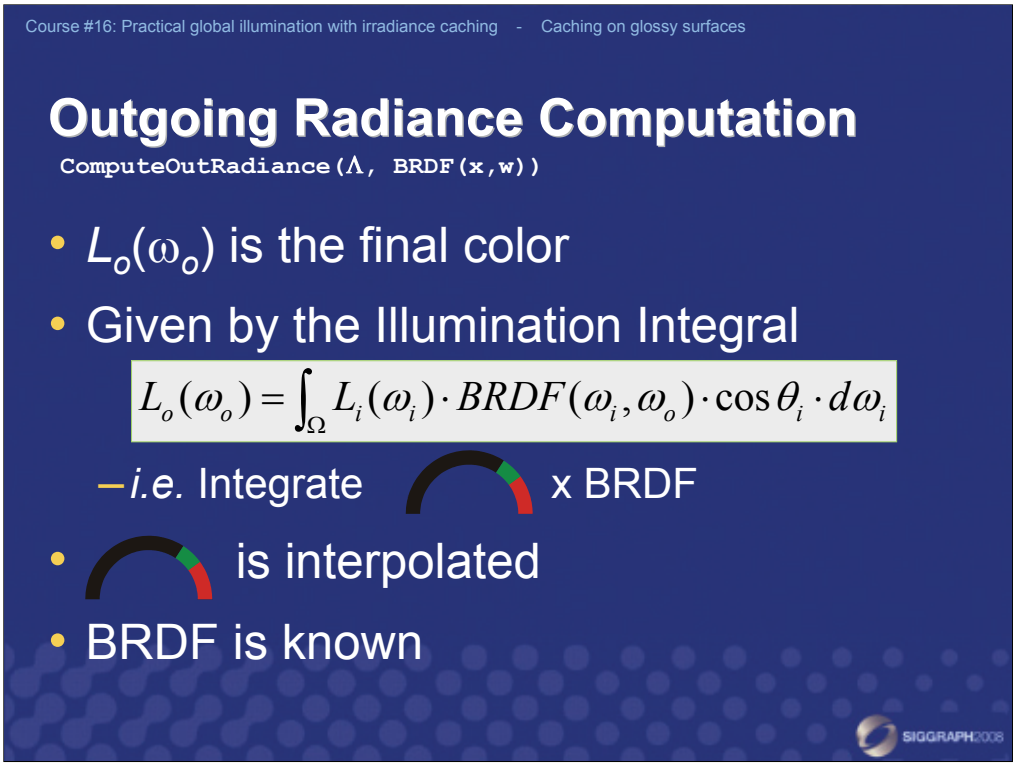
To save space, we store the gradients in form of two coefficient derivative vectors – with respect to *x* and *y* axes of the local coordinate frame at the record. We disregard the gradient with respect to the local *z* axis, since the displacements along *z* are small in the interpolation. The *xy* plane is the local tangent plane and *z* is the normal vector.

The gradients are then used in interpolation to adjust the coefficient vector stored in the record.

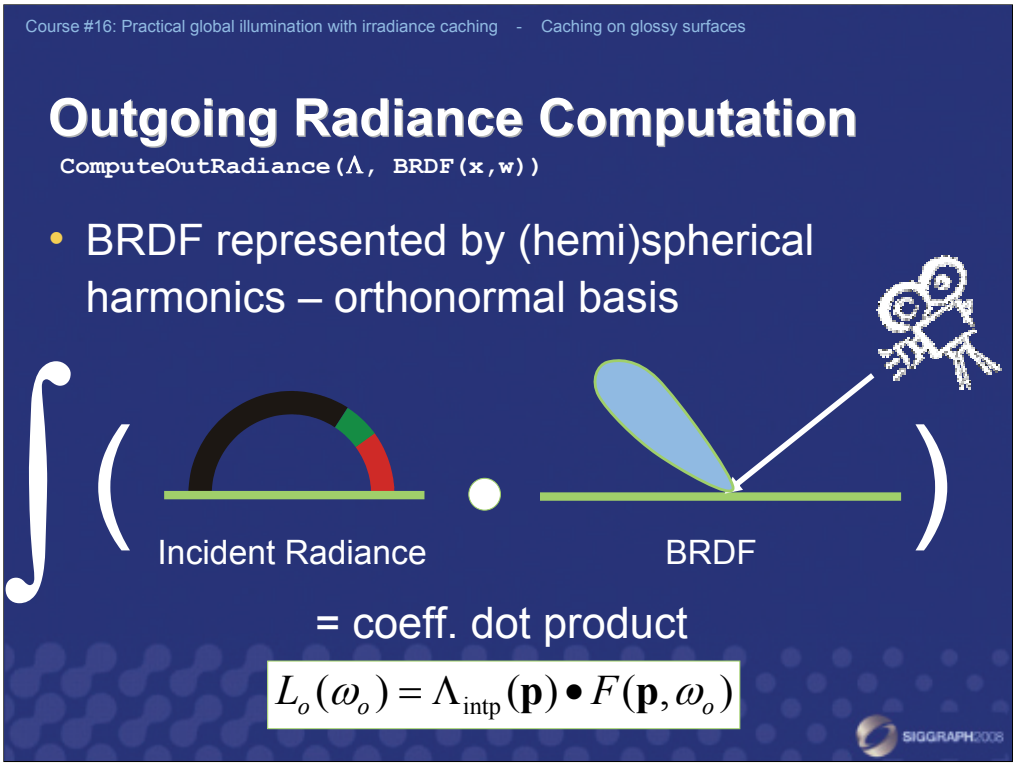


After the gradient adjustment, the interpolation procedure proceeds by applying a rotation. This is required to align the coordinate frame at the point of interpolation with the coordinate frame with respect to which the coefficients of the incoming radiance were computed. The rotation procedure takes a vector of spherical harmonic coefficients for the incoming radiance and a 3x3 rotation matrix, and outputs a coefficient vector representing the rotated incoming radiance. This rotation procedure is the bottleneck of in the radiance interpolation and hence it is essential to have an optimized procedure for this. We won't go into detail here and rather refer to the radiance caching web page where the rotation code can be downloaded.

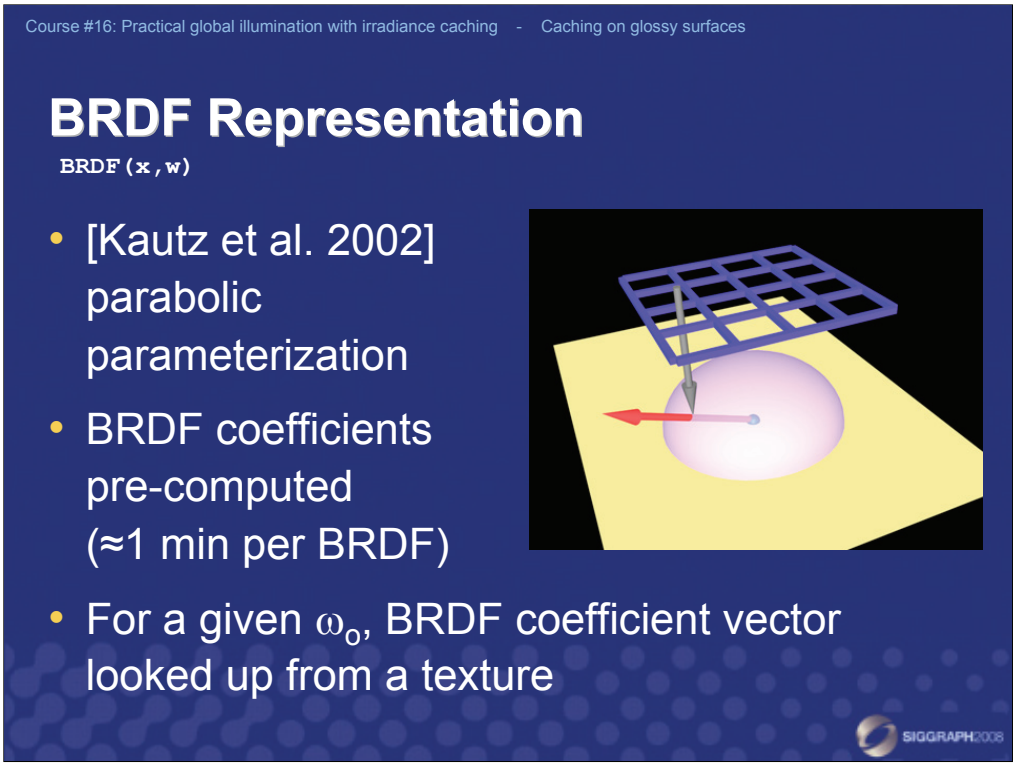
This completes the interpolation procedure.



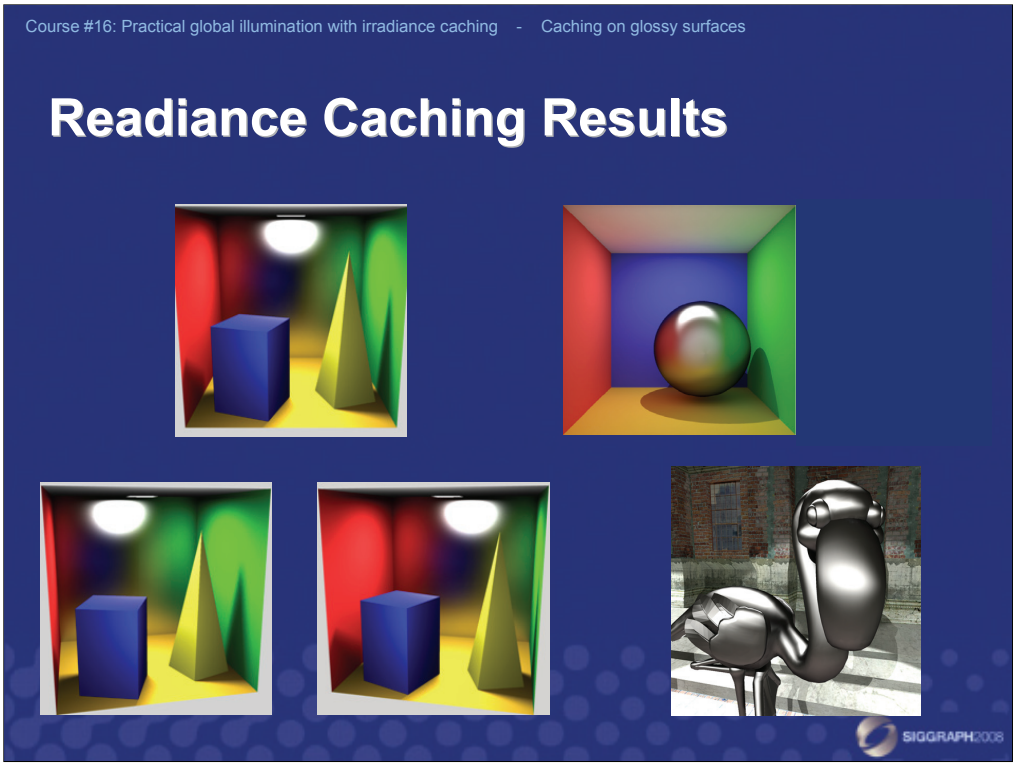
The final result we want to compute is the outgoing radiance at a point for a given outgoing direction – this is the color that corresponds to glossy indirect illumination. So far, we know how to compute or possibly interpolate the coefficient vector representing the directional distribution of incoming radiance at a point. Now we turn this into the outgoing radiance by evaluating the illumination integral, shown on the slide. Since the incoming radiance is interpolated and the BRDF can be looked up from the scene database, all the information required to evaluate the illumination integral is readily available.



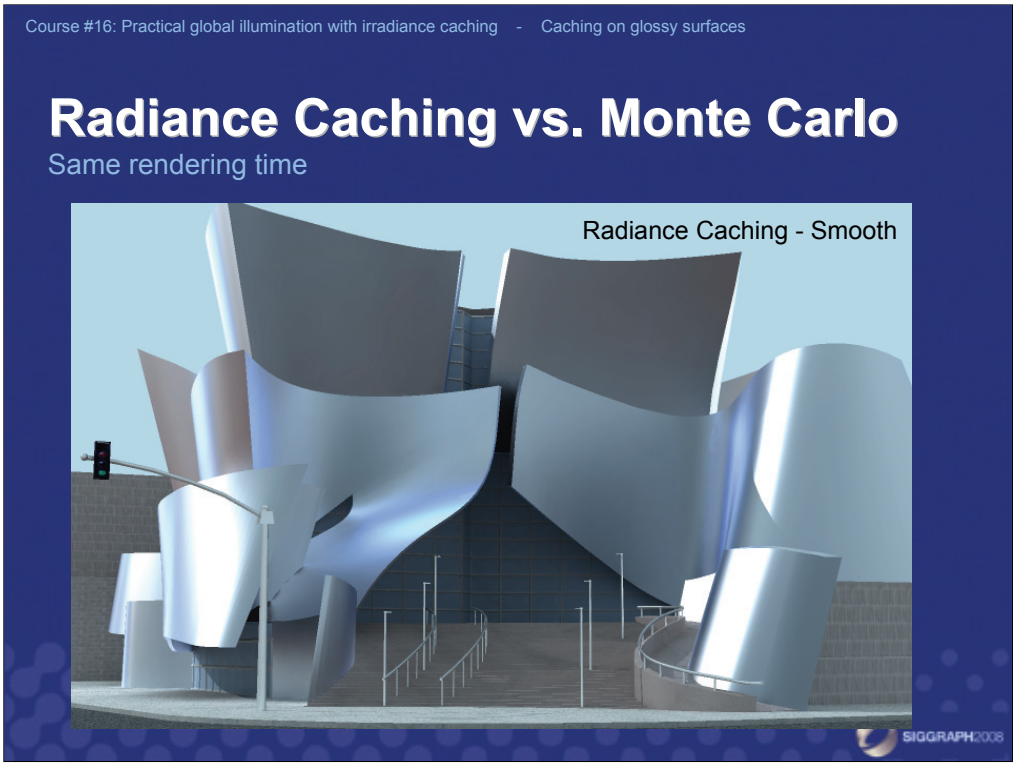
To make the evaluation of the illumination integral fast, we take advantage of the dot-product property of orthonormal bases (which spherical and hemispherical harmonics are): Integral of the product of two functions can be computed as the dot product of the coefficient vectors of these function with respect to the basis. So if we represent the BRDF times the cosine term using (hemi)spherical harmonics, all we need to do in order to compute the outgoing radiance is to dot the BRDF coefficients with the incoming radiance coefficients.



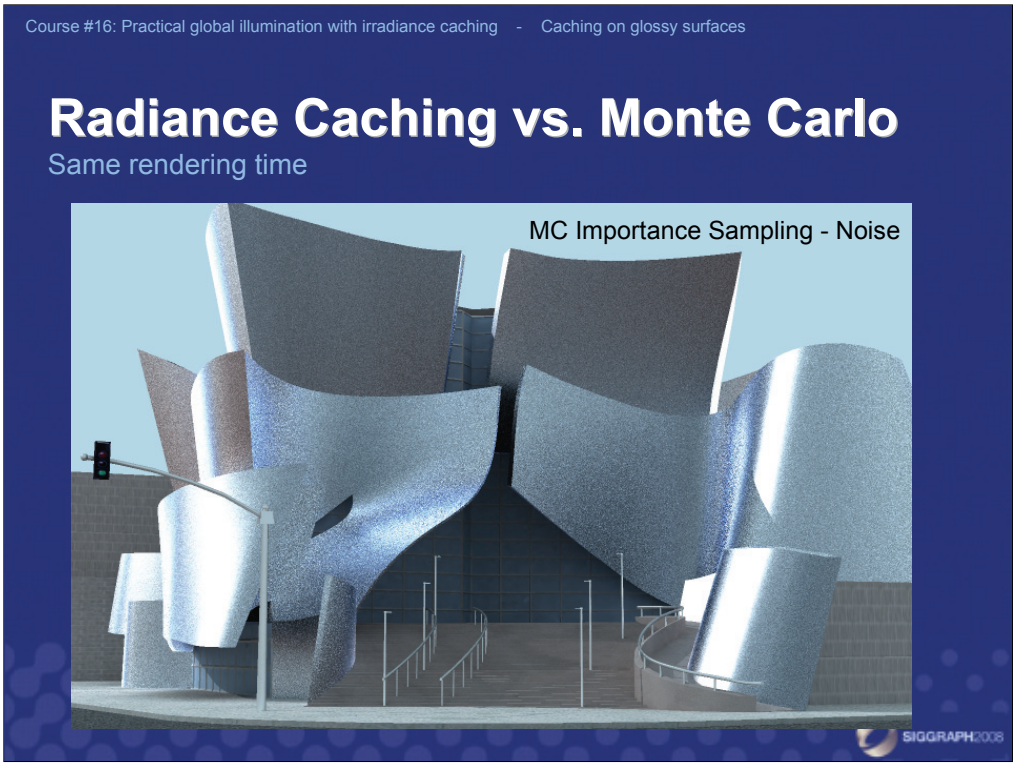
For a given, fixed, outgoing direction, the BRDF lobe is a hemispherical function. We represent this function by (hemi)spherical harmonics. We discretize the outgoing directions and for each discrete outgoing direction, we compute the coefficient vector representing the BRDF lobe. This is done in pre-process.



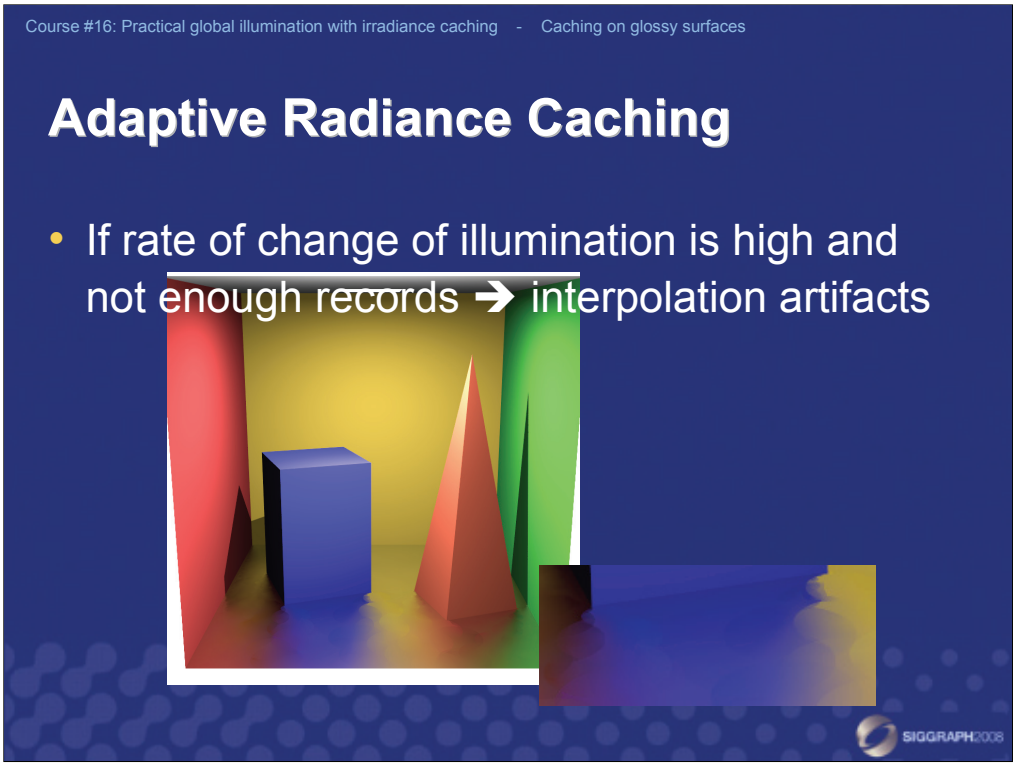
This slide shows some images rendered with radiance caching. The images show that radiance caching can handle fairly sharp glossy reflections (boxes on the left), anisotropy (sphere) and curved geometry (flamingo). For sharper or more anisotropic reflections and for more complex geometry, radiance caching is usually outperformed by Monte Carlo importance sampling. As such, radiance caching complements importance sampling.



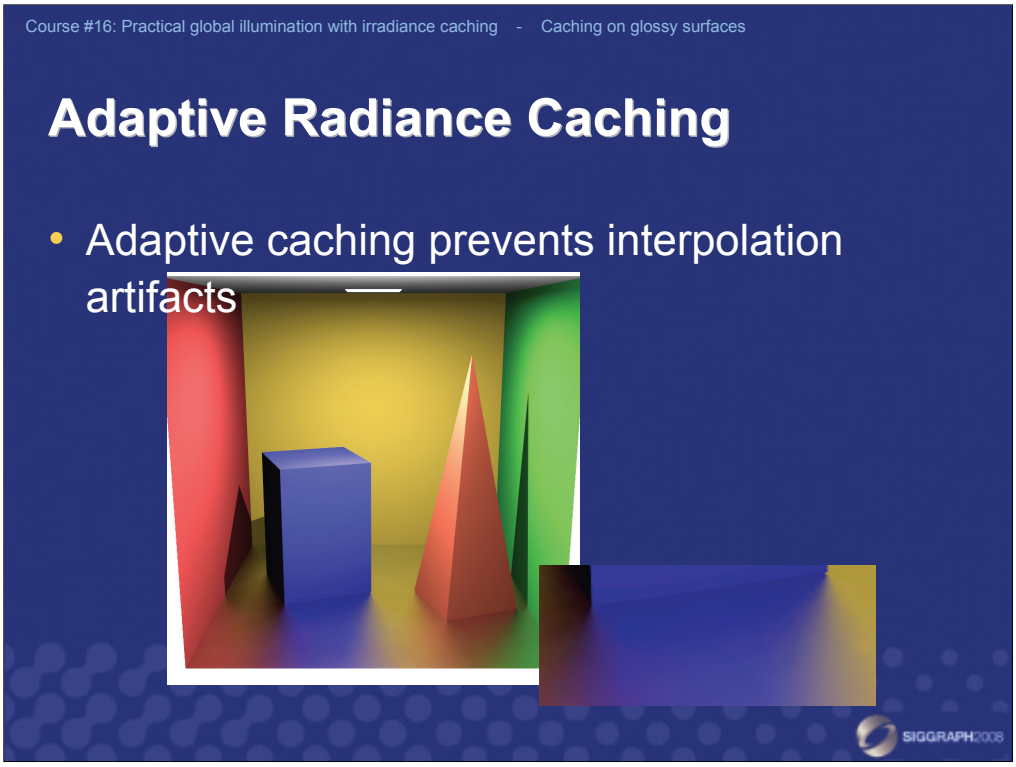
Given the same rendering time, radiance caching, produces smooth rendering.



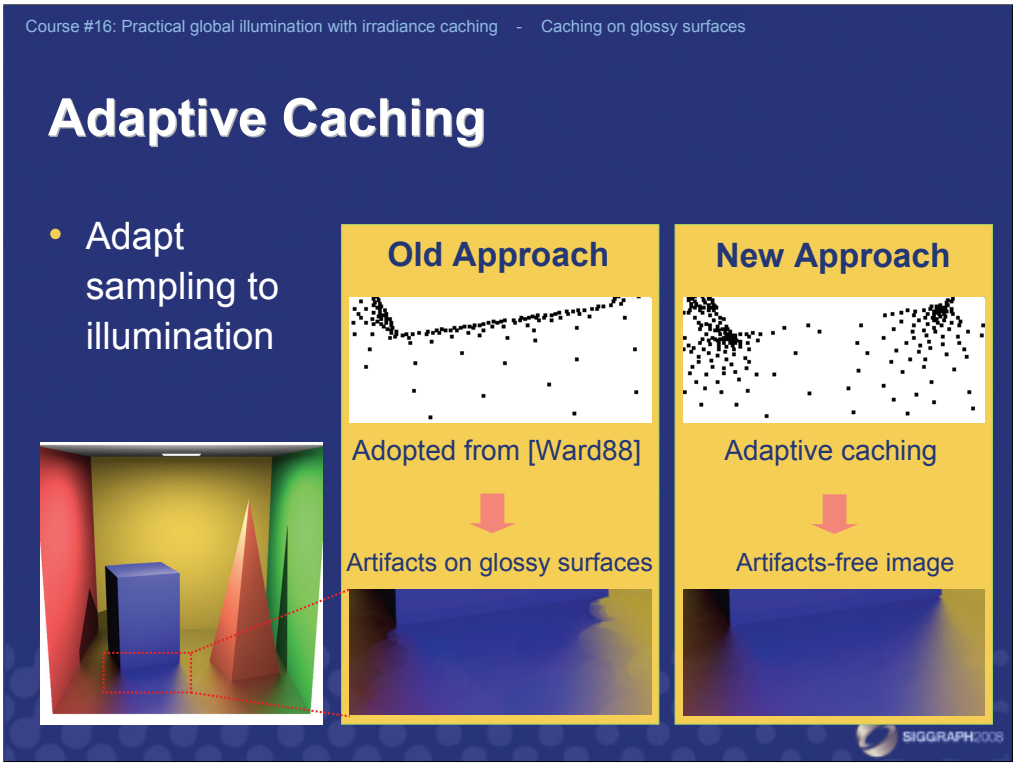
This and the following slide compare image quality delivered by MC importance sampling and radiance caching given the same rendering time. Here we see that Monte Carlo produces noisy image.



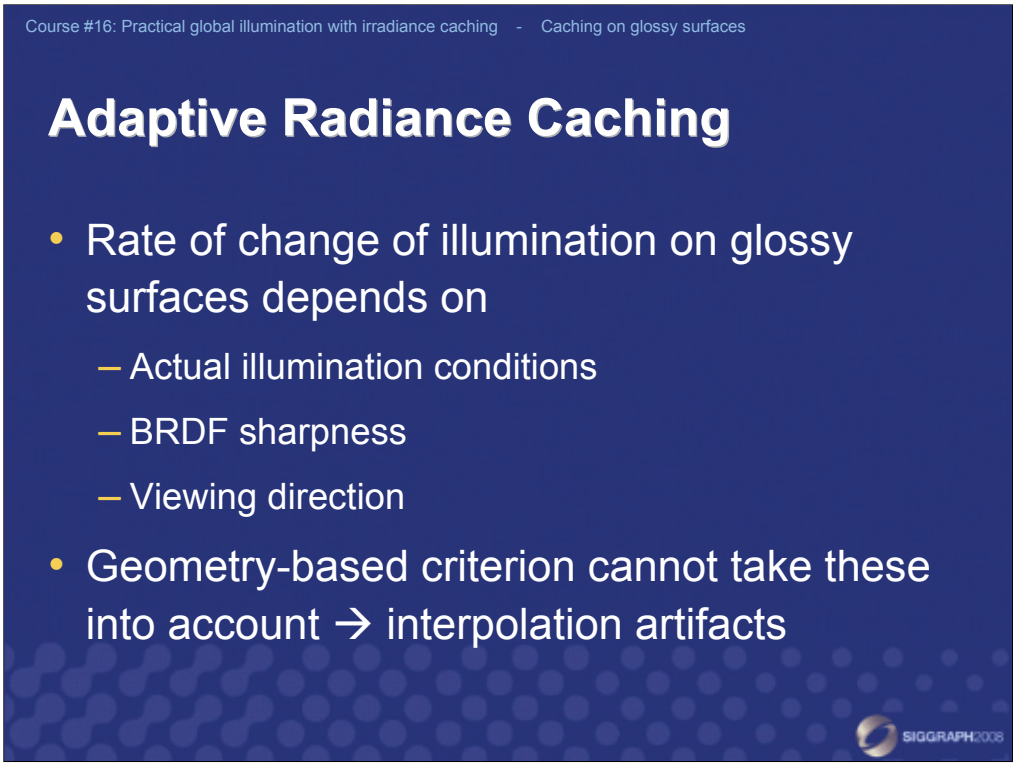
The basic radiance caching algorithm as described so far works fine in many cases, but it fails more often than irradiance caching.



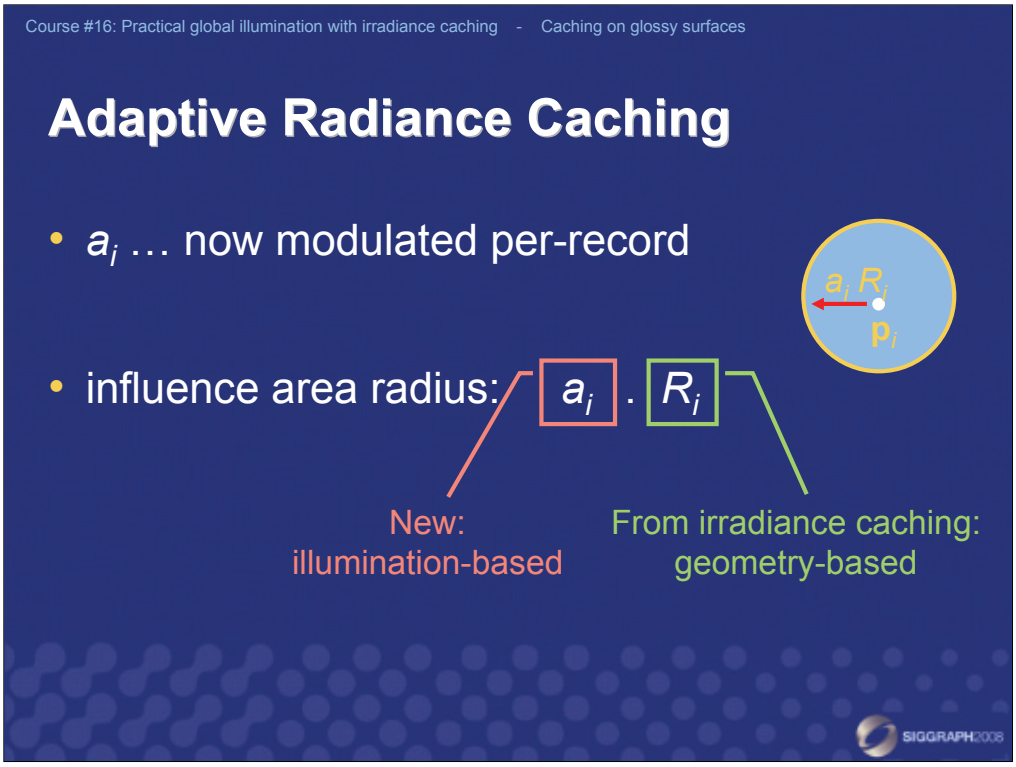
We use adaptive radiance caching to handle the failure cases.



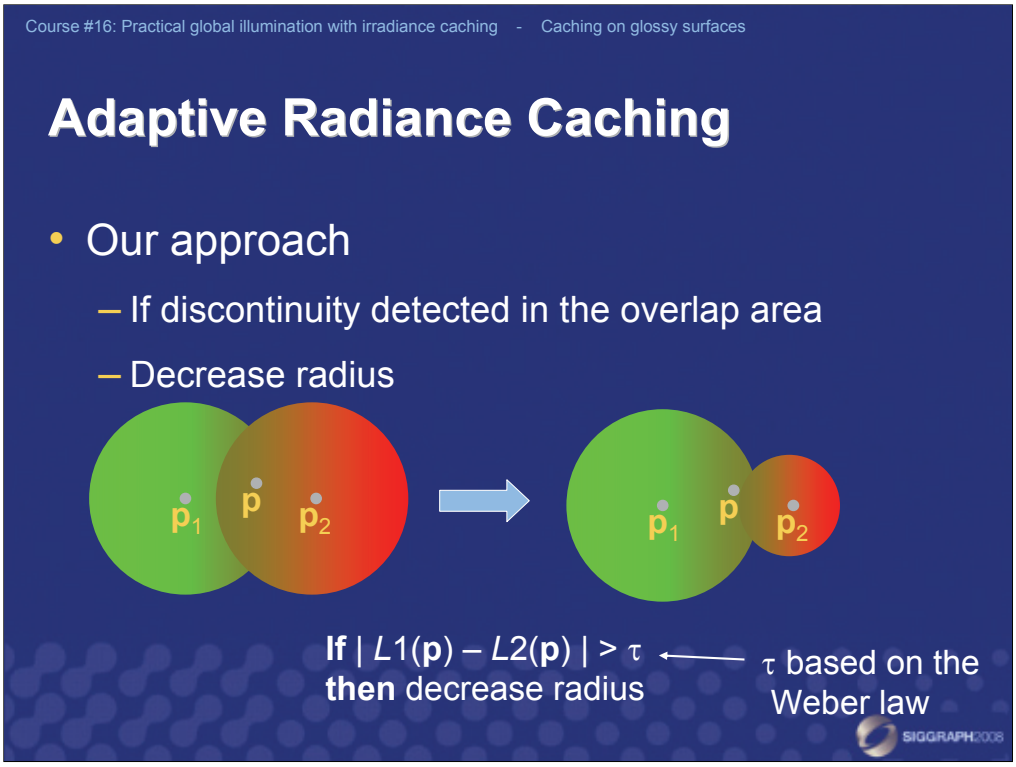
In particular, we adapt the density of spatial sampling of the indirect illumination to the actual local illumination conditions. This is in contrast to irradiance caching, where the sampling density is adapted based solely on the scene geometry.



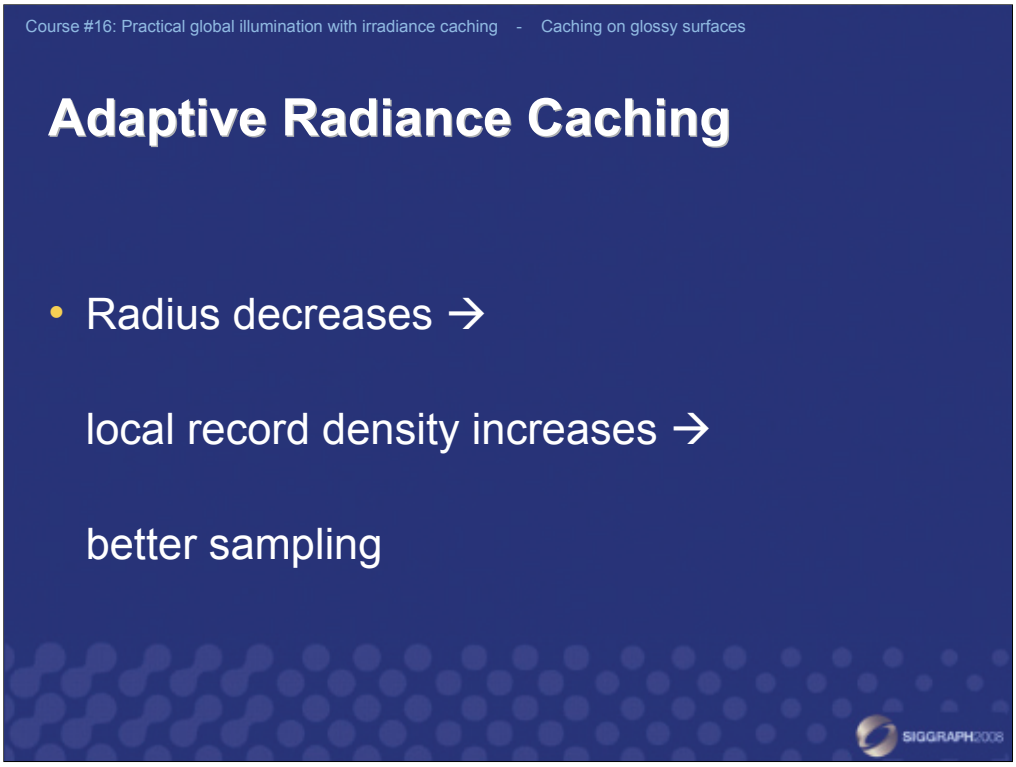
When we take the geometry-based criterion from irradiance caching and use it on glossy surfaces, we may fail to predict important illumination changes. This is because indirect glossy illumination (as opposed to indirect diffuse) very much depends on the actual illumination conditions in the scene. In addition, it depends on the BRDF sharpness and the viewing direction.



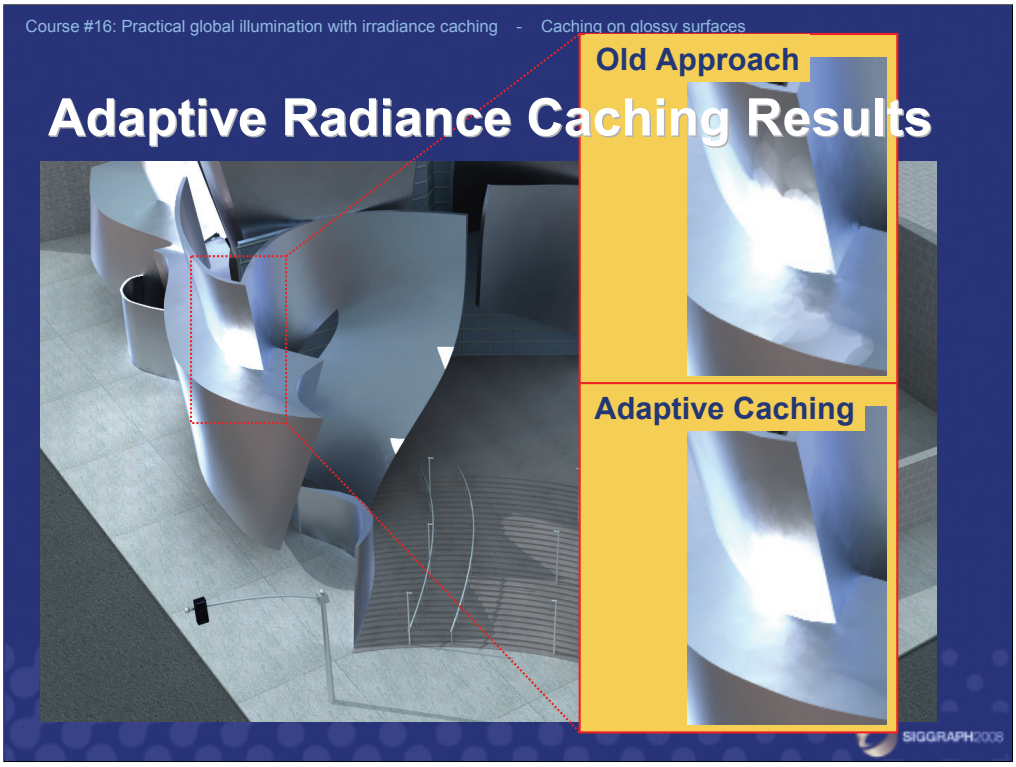
Remember that in irradiance caching and in the basic radiance caching, the record spacing is determined by the mean distance to the surrounding geometry, denoted R_i, multiplied by a user supplied constant a. In adaptive radiance caching, we extend this by also modulating the allowed error a automatically on a per-record basis in order to produce higher sampling in areas of high lighting complexity.



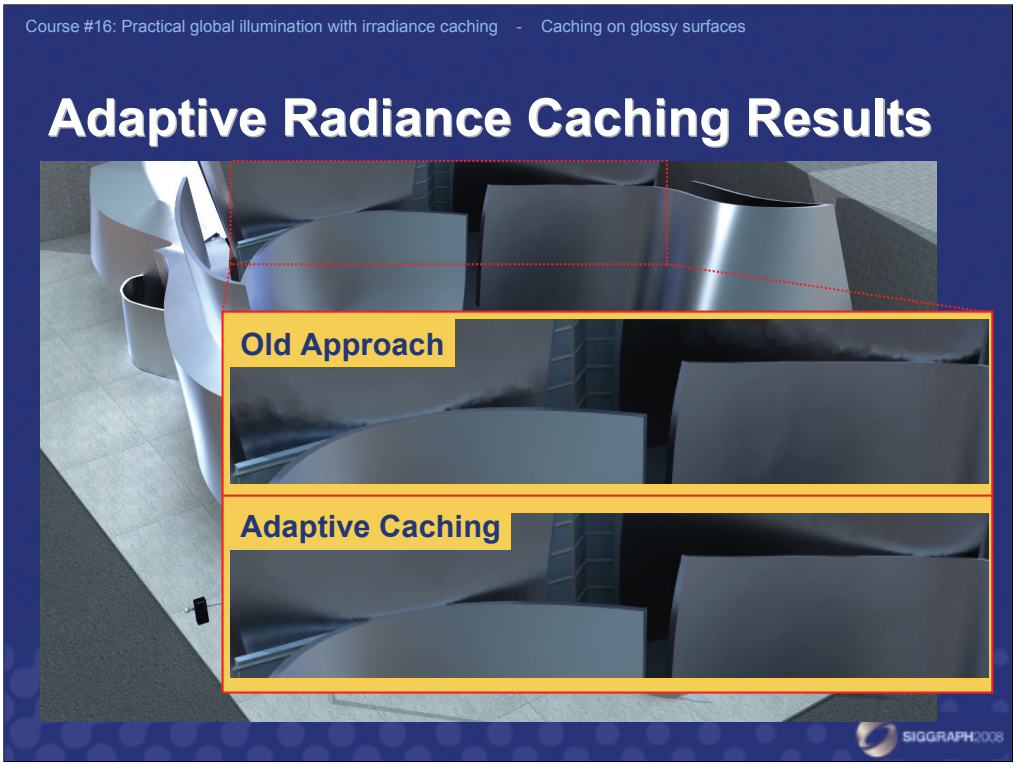
To adapt the a_i-value to the local illumination conditions, we proceed as follows. During the radiance interpolation, we perform an additional check on the difference of radiance values of the contributing records. If this difference is perceptually important, we conclude that a visible discontinuity would be produced by interpolation. To determine the perceptual importance, we use the simplest possible metric – the Weber law. It states that the minimum perceivable difference of luminance is given by a fixed fraction of the luminance value. We use the threshold of 2 to 3 %. If a potentially visible discontinuity is detected, we decrease the a_i value associated with one of the records in order to exclude it from interpolation. This 'shrinks' the area over which that record can be reused for interpolation.

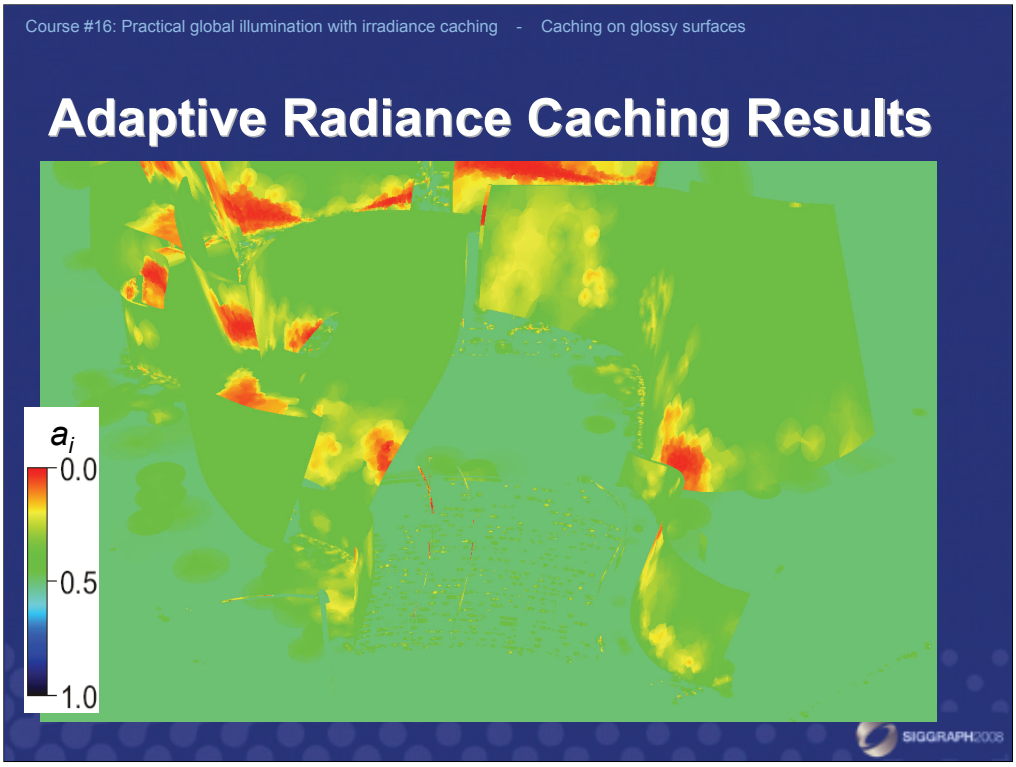


Shrinking the influence area of a records has for consequence increased local density of radiance records and thus better local sampling of indirect illumination.

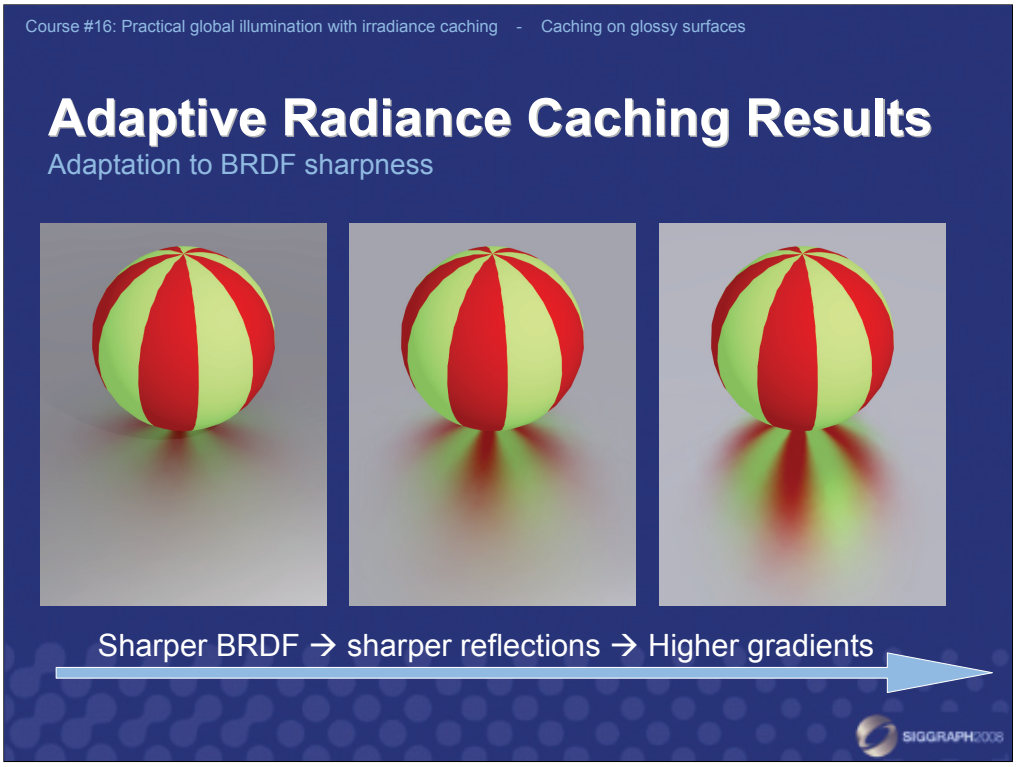


This and the following slide show the effect of adaptive radiance caching on the rendered images. The two rendering were created in the same rendering time. Without adaptive radiance caching, we see some discontinuities because of the interpolation, which are successfully suppressed by adaptive caching.

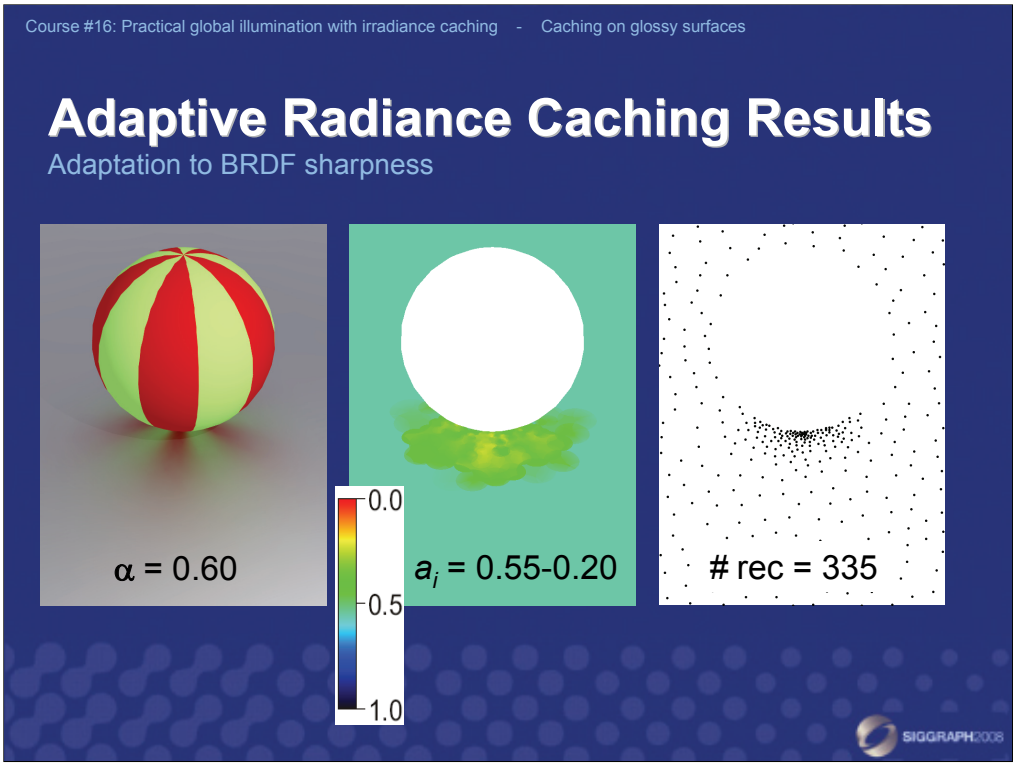

Course #16: Practical global illumination with irradiance caching - Caching on glossy surfaces
Adaptive Radiance Caching Results
Old Approach
Adaptive Caching
SIGGRAPH2008



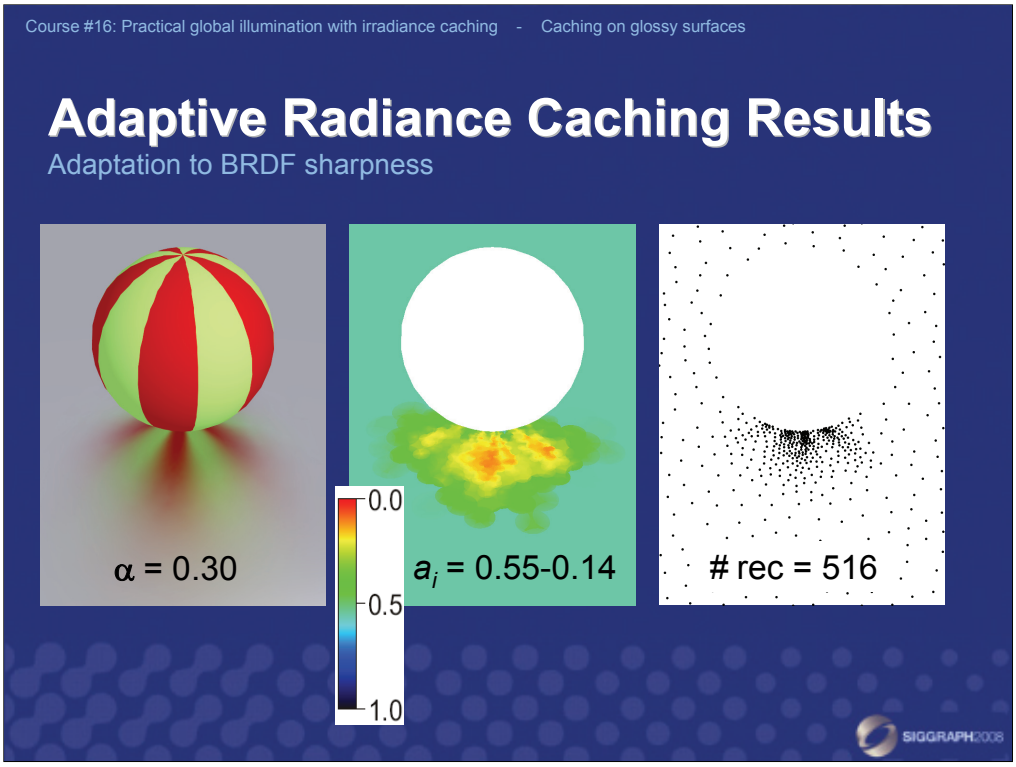
This image shows the locally adapted *ai* value for the rendering of the Walt Disney Hall.



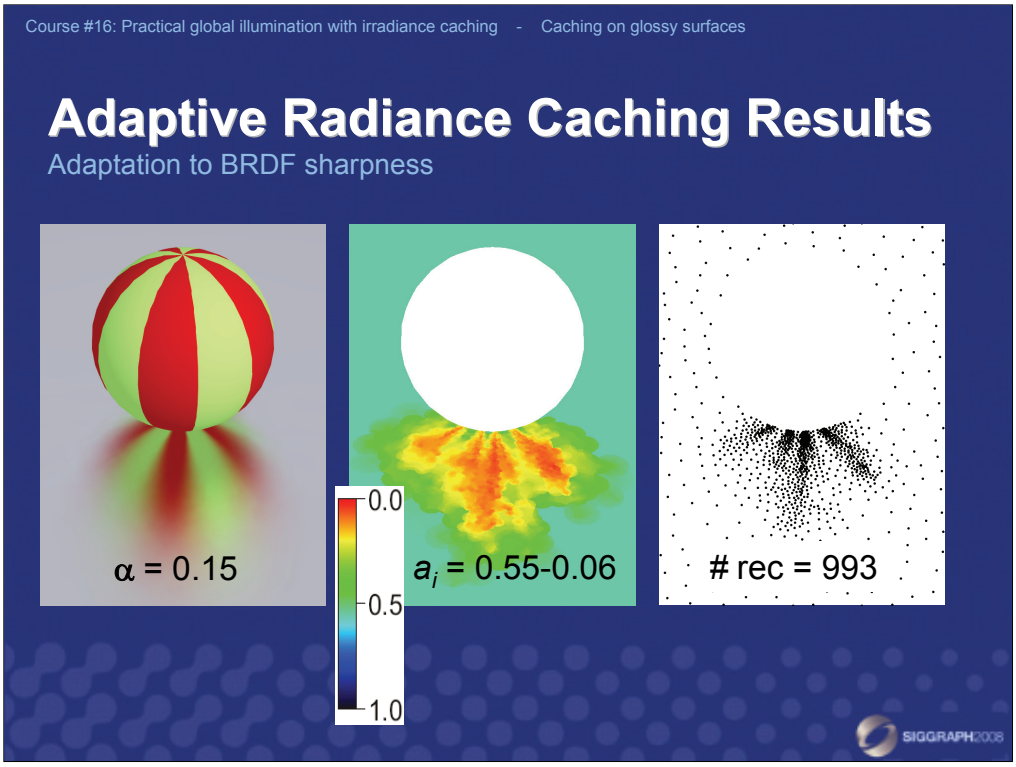
Adaptive caching also automatically adapts the record density to the BRDF sharpness. On these three images, the BRDF sharpness of the glossy floor gradually increases from left to right, giving sharper and sharper reflections. Sharper reflections result in higher illumination gradients.



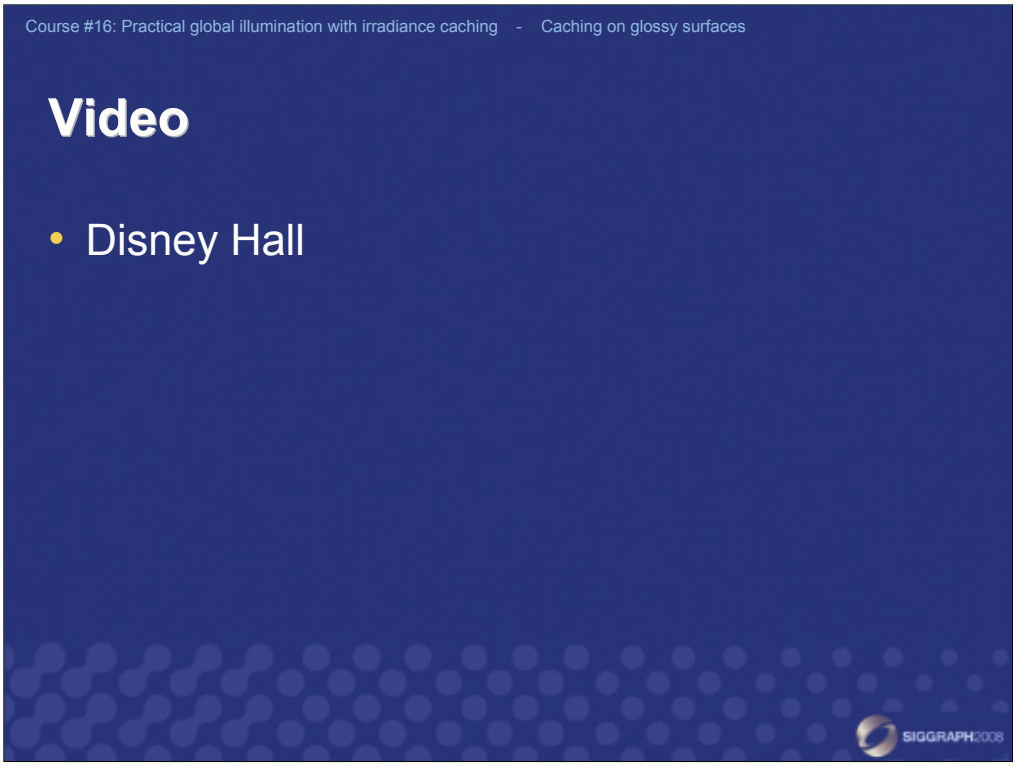
When the BRDF sharpness is small, we can get away with very sparse sampling.

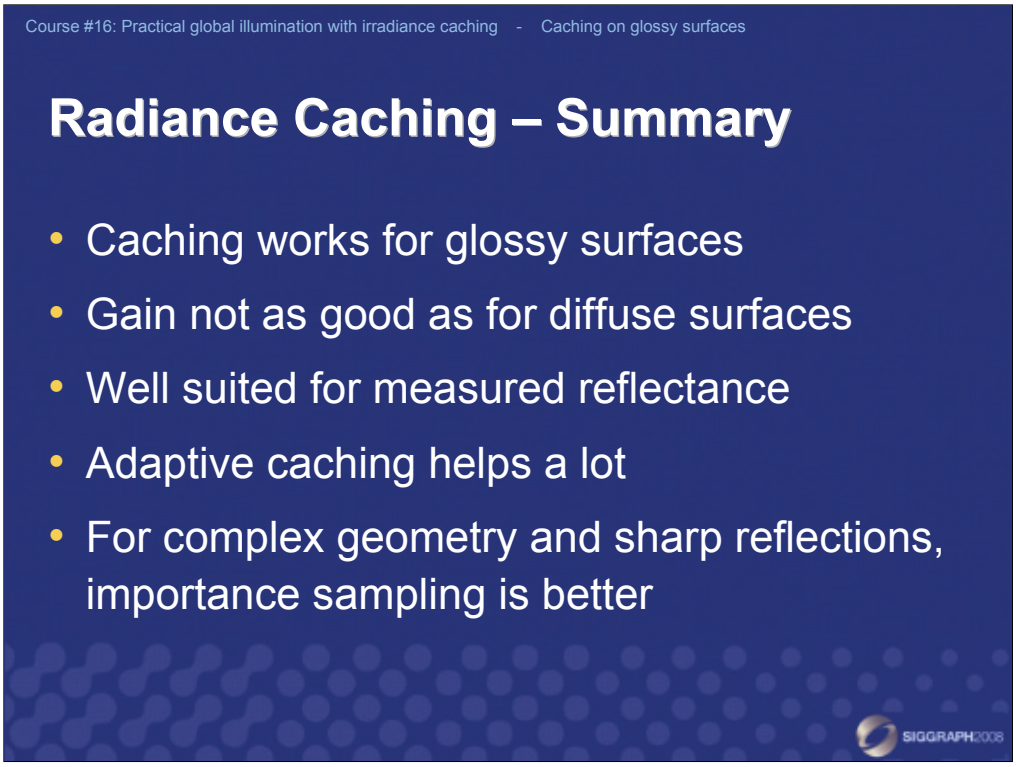


As the BRDF sharpness increases, the adaptive caching increases the record density accordingly, in order to avoid interpolation artifacts.

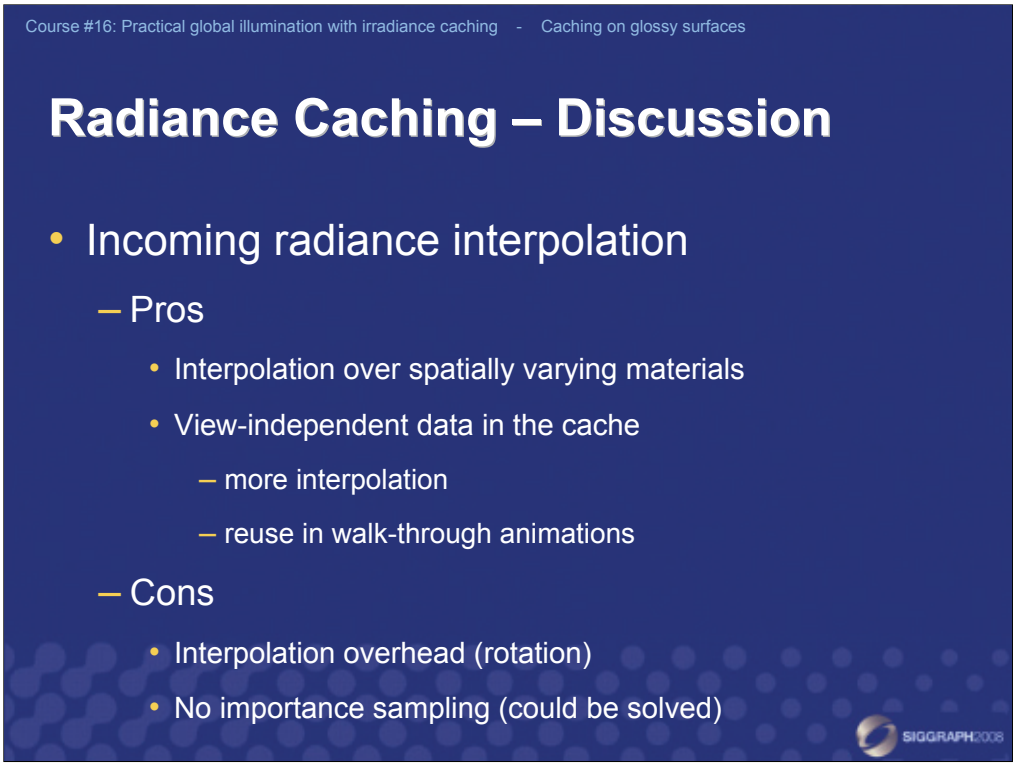


Increasing the BRDF sharpness further, still more records are automatically added by adaptive caching to preserve smoothness of indirect illumination. Notice that if we based the interpolation criterion a priori on the BRDF sharpness, we would have to increase the sampling density all over the glossy floor, which would result in a overly dense sampling and a significant performance drop.


Course #16: Practical global illumination with irradiance caching - Caching on glossy surfaces
Video
• Disney Hall
SIGGRAPH2008

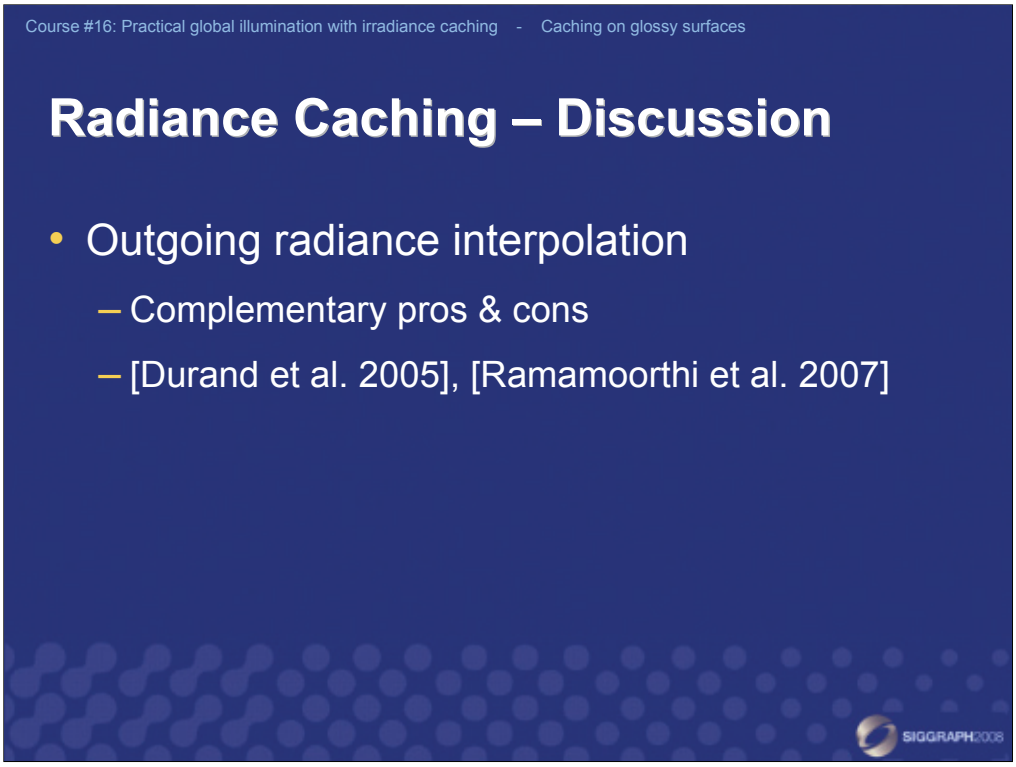


We have shown that lazy illumination caching can be used not only for diffuse surfaces, but even for glossy ones. The performance gain is not as substantial as for irradiance caching, though. The main overhead in radiance caching is due to uniform hemisphere sampling in radiance caching – this requires many sampling rays and, for a certain threshold of BRDF sharpness, importance sampling out-performs radiance caching. In addition, the interpolation performance is crippled by the spherical harmonic rotation. Nevertheless, radiance caching brings significant performance gains over Monte Carlo importance sampling when rendering fairly smooth geometry with low-frequency BRDF. In particular, it effective for rendering with measured reflectance data, for which importance sampling is difficult.

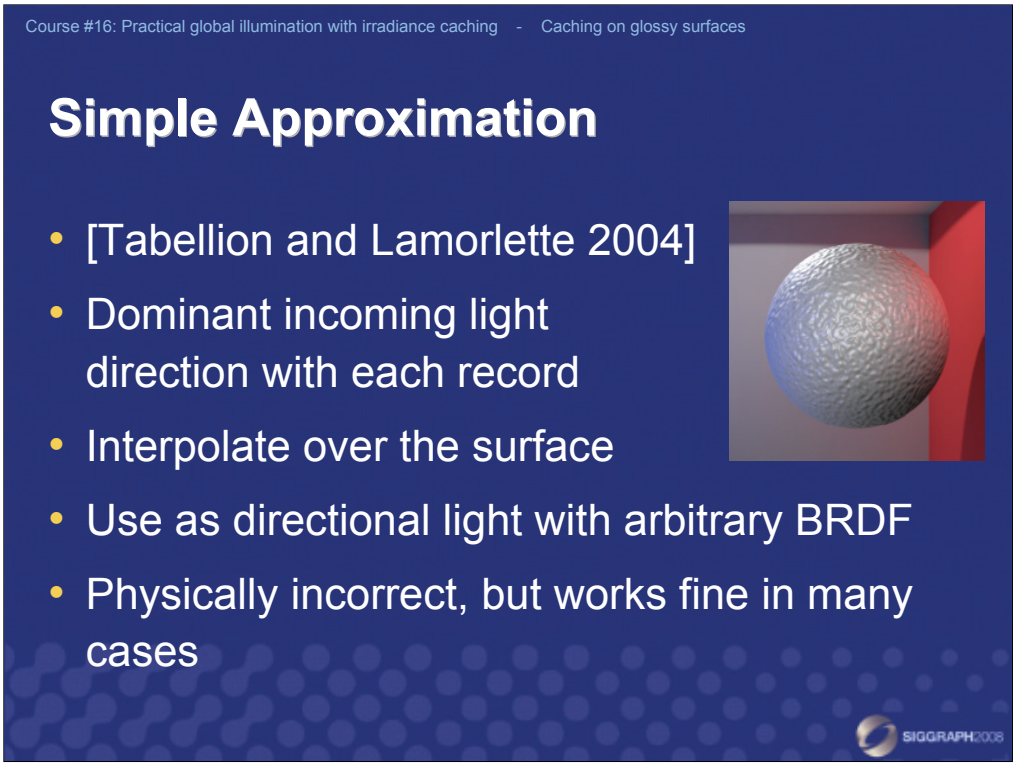


The essential design choice in radiance caching is to interpolate the directional distribution of incoming radiance (as opposed to interpolation outgoing radiance for the viewing direction). Since the cached quantity is view-independent, we can reuse cached values over larger areas and we can re-use the them in animations. In addition, we can also interpolate over spatially varying materials.

On the other hand, the overhead related to caching and interpolating a directional function may be quite large as discussed on the previous slide.



An alternative to caching incoming radiance is to interpolate outgoing radiance for the viewing direction.



An alternative method for illumination interpolation on glossy surfaces was proposed by Tablellion and Lamorlette [2004].

It is a simple extension of irradiance caching. When sampling the hemisphere, the dominant incoming light direction is computed weighted average of sample directions weighted by the radiance coming from that direction. This dominant direction is stored along with irradiance in the cache for possible interpolation. When shading a surface with arbitrary, non-diffuse BRDF, the interpolated irradiance is interpreted as coming all from the dominant direction, i.e. as if it was a directional light source. A standard surface shader is then used to compute color.

Their techniques is much lighter-weighted than radiance caching. It is an arbitrary approximation, but works fine in many cases.

ɔurse #16: Practical global illumination with irradiance caching - Caching on glossy surfaces
Thank you
?
Source code:
http://www.cgg.cvut.cz/show_research.php?page=01

Radiance Caching References

- P. Gautron, J. Křivánek, S. Pattanaik, and K. Bouatouch, A Novel Hemispherical Basis for Accurate and Efficient Rendering, Eurographics Symposium on Rendering, 2004.
- J. Křivánek, P. Gautron , S. Pattanaik, and K. Bouatouch, Radiance Caching for Efficient Global Illumination Computation , IEEE TVCG, Vol. 11, No. 5, Sep./Oct. 2005
- J. Křivánek, K. Bouatouch, S. Pattanaik, and J. Žára, Making Radiance and Irradiance Caching Practical: Adaptive Caching and Neighbor Clamping, Eurographics Symposium on Rendering, 2006.

Source code:
http://www.cgg.cvut.cz/show_research.php?page=01