

Realistický rendering

© 2010-2017 Josef Pelikán, CGG MFF UK

<http://cgg.mff.cuni.cz/>

<http://cgg.mff.cuni.cz/~pepca/>

Festival fantazie, Chotěboř, 4. 7. 2017



Obsah přednášky

- co je realistický rendering?
- příklady použití
- Ray-tracing (historie)
- trocha teorie – radiometrie (fyzika)
- optický popis materiálu
- Monte-Carlo výpočty šíření světla (současnost)



[+1] [Front] [Wireframe]

[+] [Perspective] [Realistic + Edged Faces]

[+1] [Front] [Realistic]

[+] [Camera002] [Wireframe]

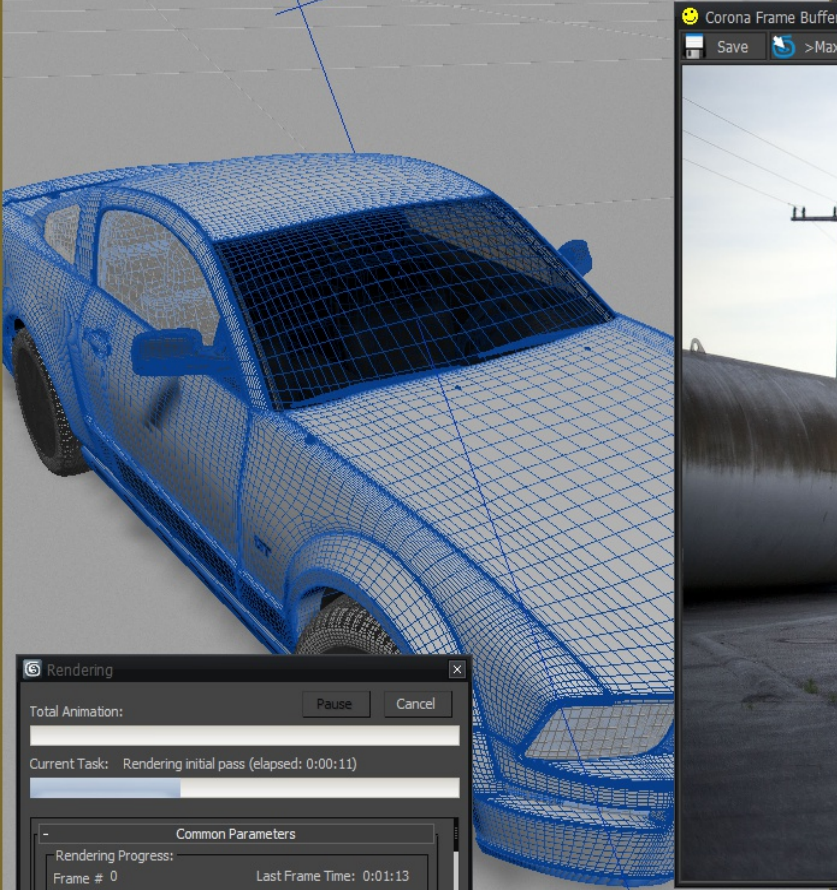
Render Setup: CoronaRenderer

Common Corona Settings Render Elements

About Corona Renderer

Corona Renderer Alpha
(c) Render Legion s.r.o., 2014
Homepage: <http://corona-renderer.com>

WARNING: this is slower ASSERT version.



Corona Frame Buffer (1:2)

Save >Max Ctrl+C Refresh Erase Tools BEAUTY

A rendered image of a red sports car (Ford Mustang) in a scene. The car is positioned in a perspective view, facing towards the right. The scene includes a large cylindrical tank, a utility pole, and a building in the background. The car has white racing stripes on its hood and roof.

Stats DR ColorMap

TIMES

Estimated remaining: ---

Scene parsing: 0:00:05

Geometry: 0:00:00

Secondary GI: 0:00:00

Rendering: 0:00:05

TOTAL elapsed: 0:00:10

SCENE

Primitives uniq.: 786,015

Primitives inst.: 1,236,063

Geometry groups: 111

Instances: 114

Portals: 0

Lights (groups): 1 (1)

GI CACHES

HD records: 0

HD success rate: 0.0

PERFORMANCE

Passes total: 0

Rays/s total: 1,226,086

Rays/s actual: 1,266,861

Sampl/s total: 185,059

Sampl/s act.: 190,371

Rays/sample: 6.6

VFB refresh time: 341ms

Rendering

Pause Cancel

Total Animation: [Progress Bar]

Current Task: Rendering initial pass (elapsed: 0:00:11)

Common Parameters

Rendering Progress:

Frame # 0 Last Frame Time: 0:01:13

1 of 1 Total Elapsed Time: 0:00:00

Pass # 1/1 Time Remaining: ????:??

Render Settings:

Solver: Object-based CombSolver local frac: 0,33

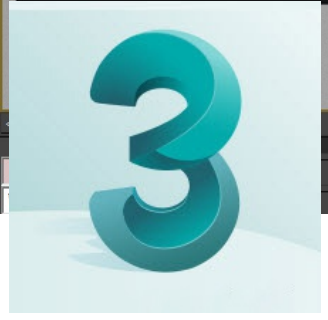
Sampling mode: MIS - both (bes) CombSolver global frac: 0,33

Portal fraction: 0,75

Preset: [Dropdown]

View: Quad 4 - Camer

Render



Reklama



Architektonická vizualizace



Cíle realistického zobrazování



- věrně **napodobit přírodu**
 - ◆ virtuální scéna reprezentovaná v počítači
- přesně **simulovat šíření světla** ve scéně
 - ◆ „predictive rendering“
- nebo jen „**důvěryhodné zobrazení**“
 - ◆ laický pozorovatel nemá poznat, že je obrázek umělý...
- **rychlost** vykreslování
 - ◆ „off-line“ rendering (reklama, film /výpočetní farmy/)
 - ◆ „real-time“ (min. 25 fps)

Dnešní přednáška



- věrně **napodobit přírodu**
 - ◆ virtuální scéna reprezentovaná v počítači
- ~~■ přesně **simulovat šíření světla** ve scéně
 - ◆ „predictive rendering“~~
- nebo jen „**důvěryhodné zobrazení**“
 - ◆ laický pozorovatel nemá poznat, že je obrázek umělý...
- **rychlost** vykreslování
 - ◆ „off-line“ rendering (reklama, film /výpočetní farmy/)
 - ~~◆ „real-time“ (min. 25 fps)~~



Dnes ne!

- grafika v počítačových hrách
- grafické karty (GPU)
- modelování 3D scény (3DS MAX, Blender, ...)
- animace
- simulace fyzikálních jevů
 - ◆ vlny na vodě, výbuchy, obloha, mraky, ...
- analýza obrazu a počítačové vidění
 - ◆ ani použití neurosítí nebo obecného AI v renderingu

Aplikace

- design, architektura, umění
 - ◆ šíření světla v interiéru, kokpitu, ...
- zábavní průmysl
 - ◆ filmy (IL&M, Pixar, DreamWorks, ... „off-line“)
 - ◆ videohry („real-time“)
- média
 - ◆ televize (virtuální studia, ...)
 - ◆ reklamy



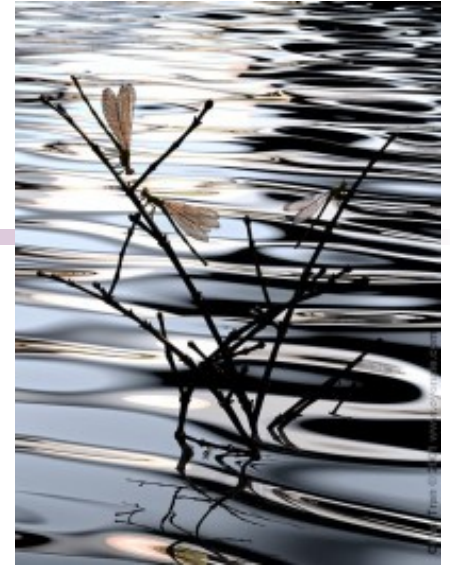
Historie – klasické vykreslování



- Straßer, Catmull 1974: Z-buffer
- **ploškový model**
 - ◆ nejčastěji trojúhelníkové sítě
- **výpočet viditelnosti**
 - ◆ Z-buffer
- **přibližné světelné poměry**
 - ◆ lokální osvětlovací model, vržené stíny
- dnes na **GPU: textury, shadery**



Historie – Ray-tracing



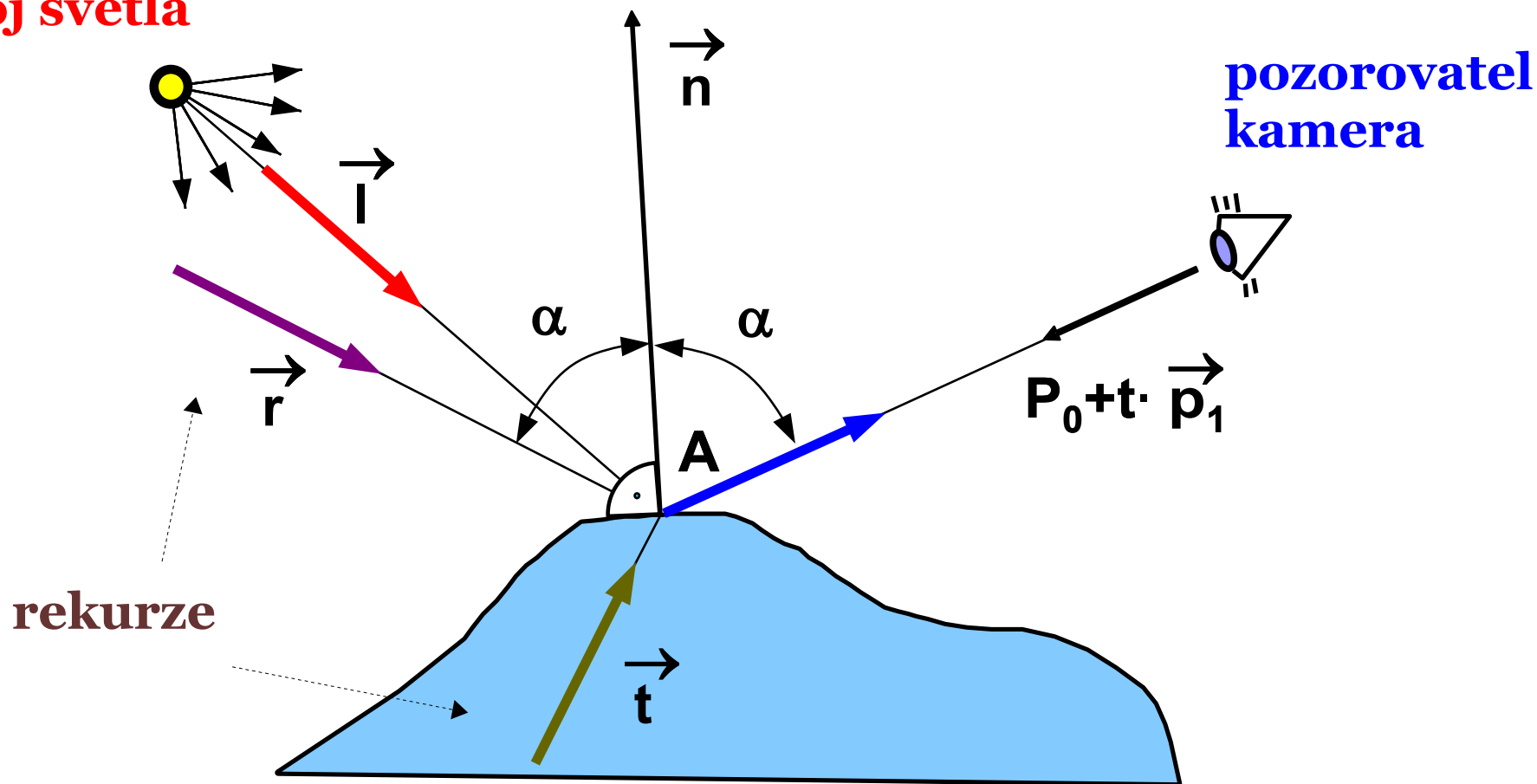
- Whitted 1980: základní Ray-tracing
- **geometrický přístup**
 - ◆ sleduje se jenom ideálně odražený paprsek
- výpočetně velmi náročný **výpočet průsečíku** paprsku se scénou
 - ◆ 90% času → urychlovací metody
- snadné **vylepšení vzhledu**
 - ◆ textury, anti-aliasing, shadery
 - ◆ distribuovaný RT (Monte-Carlo)



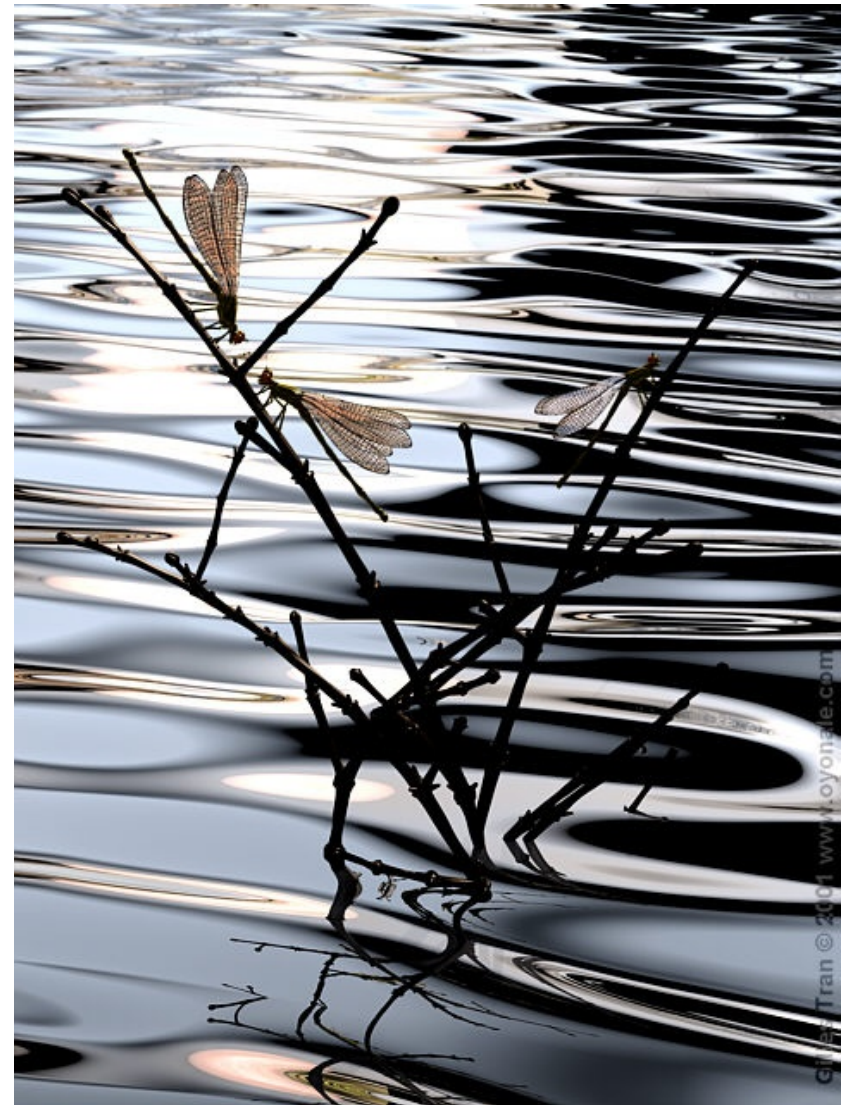
Ray-tracing - princip



zdroj světla



Ray-tracing – příklady



Ray-tracing – příklady



Ray-tracing – příklady

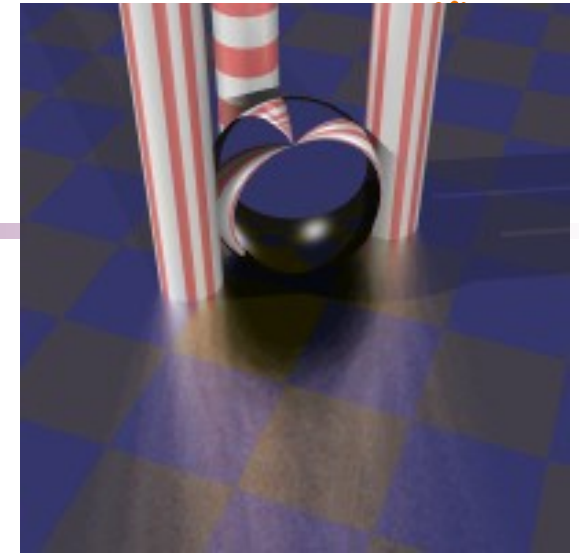


Ray-tracing – příklady



Historie – Distributed RT

- Cook 1984: Distributed Ray Tracing
- **vylepšení kvality výsledku**
 - ◆ integrál nahrazuje původně jediný vzorek
 - ◆ měkké stíny, odrazy, lomy, difrakce
 - ◆ rozmazání pohybem
 - ◆ hloubka ostrosti kamery
- výpočetně náročnější metody
 - ◆ Monte-Carlo algoritmy
 - ◆ stonásobně víc paprsků...



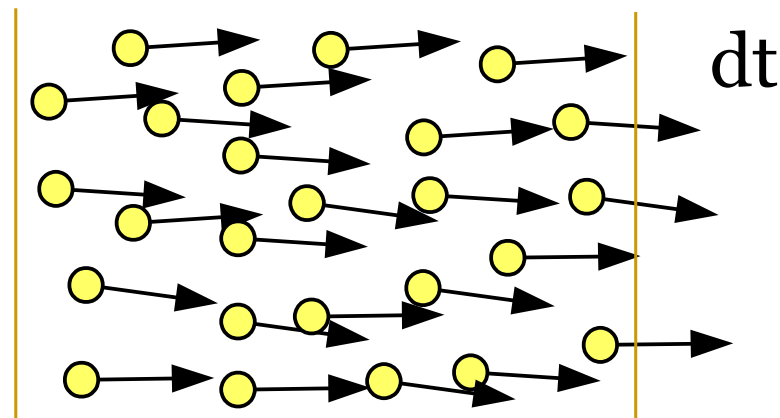
Základy radiometrie I



Zářivý tok, výkon (*Radiant flux, Radiant power*)

$$\Phi = \frac{dQ}{dt} \quad [W]$$

Počet fotonů (přepočtených na energii) za jednotku času
(100W žárovka – cca 10^{19} fotonů/s, oko z monitoru – 10^{12} f/s).



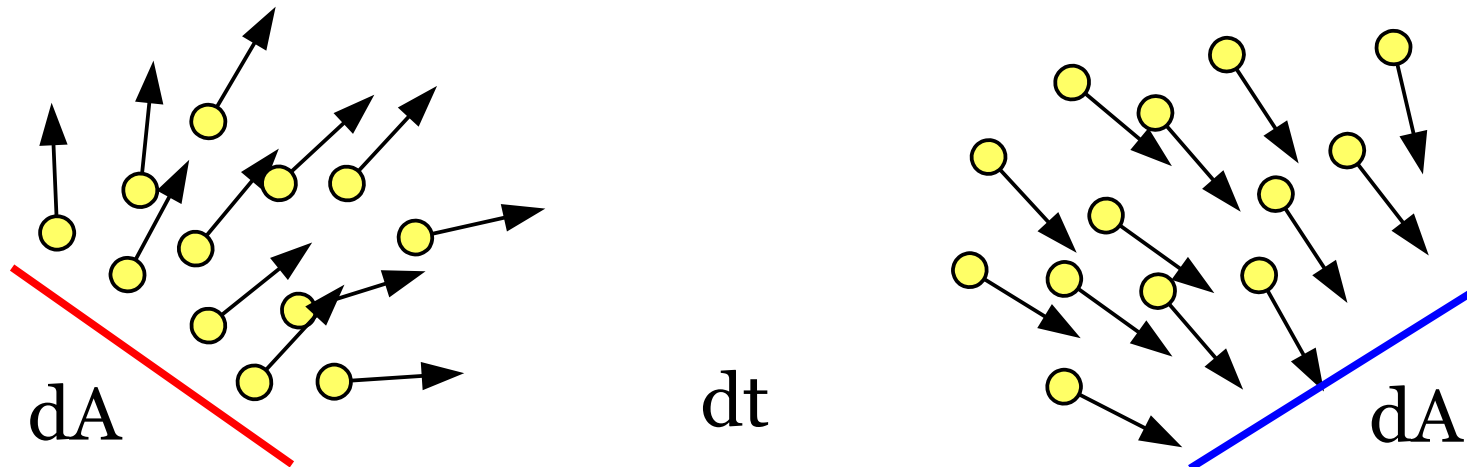
Základy radiometrie II



Hustota zářivého toku (*Irradiance, Radiant exitance, Radiosity*)

$$E(x) = \frac{d\Phi(x)}{dA(x)} \quad [\text{W/m}^2]$$

Plošná hustota fotonů (přepočtených na energii) dopadajících nebo vyzařovaných za jednotku času.



Základy radiometrie III



Zář (*Radiance*)

$$L(x, \omega) = \frac{d^2 \Phi(x, \omega)}{d A_{\omega}^{\perp}(x) d \sigma(\omega)} \quad [\text{W/m}^2/\text{sr}]$$

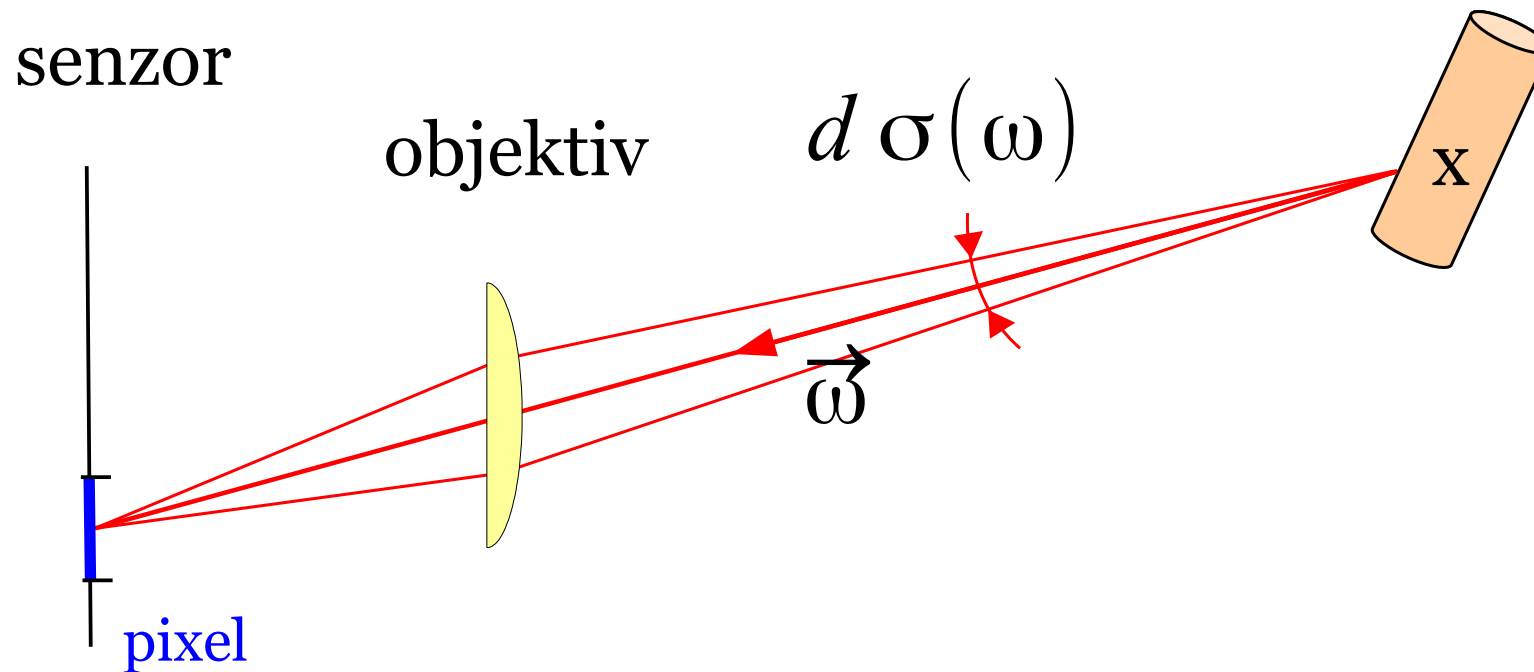
Počet fotonů (přepočtených na energii) procházejících za jednotku času malou ploškou kolmou na směr ω . Záření míří do malého kužele kolem daného směru ω .

Zář je veličina definovaná jako **hustota** vzhledem k $d\mathbf{A}^{\perp}$ a současně vzhledem k prostorovému úhlu $d\sigma(\omega)$.

Radiance I



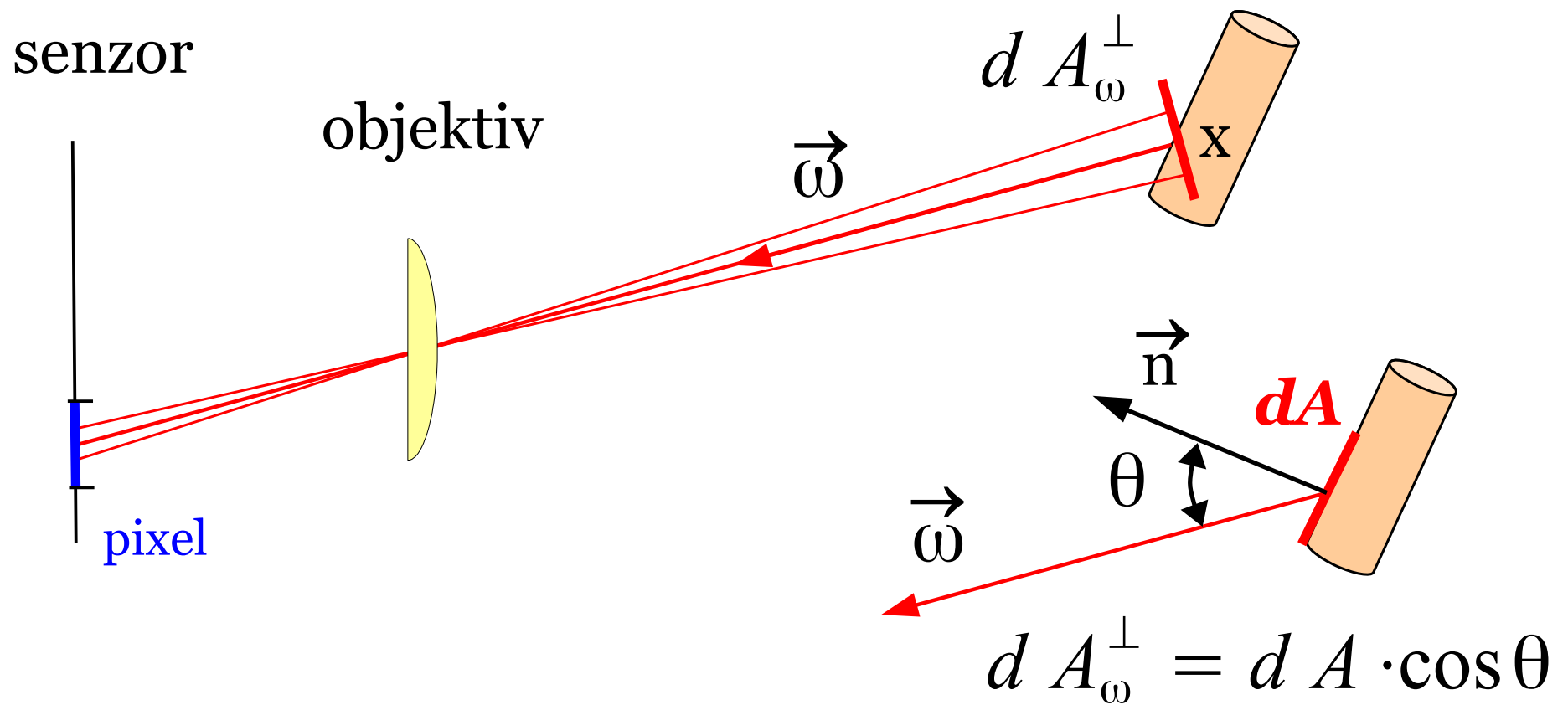
$$\Phi(x, \omega) \propto d\sigma(\omega)$$



Radiance II



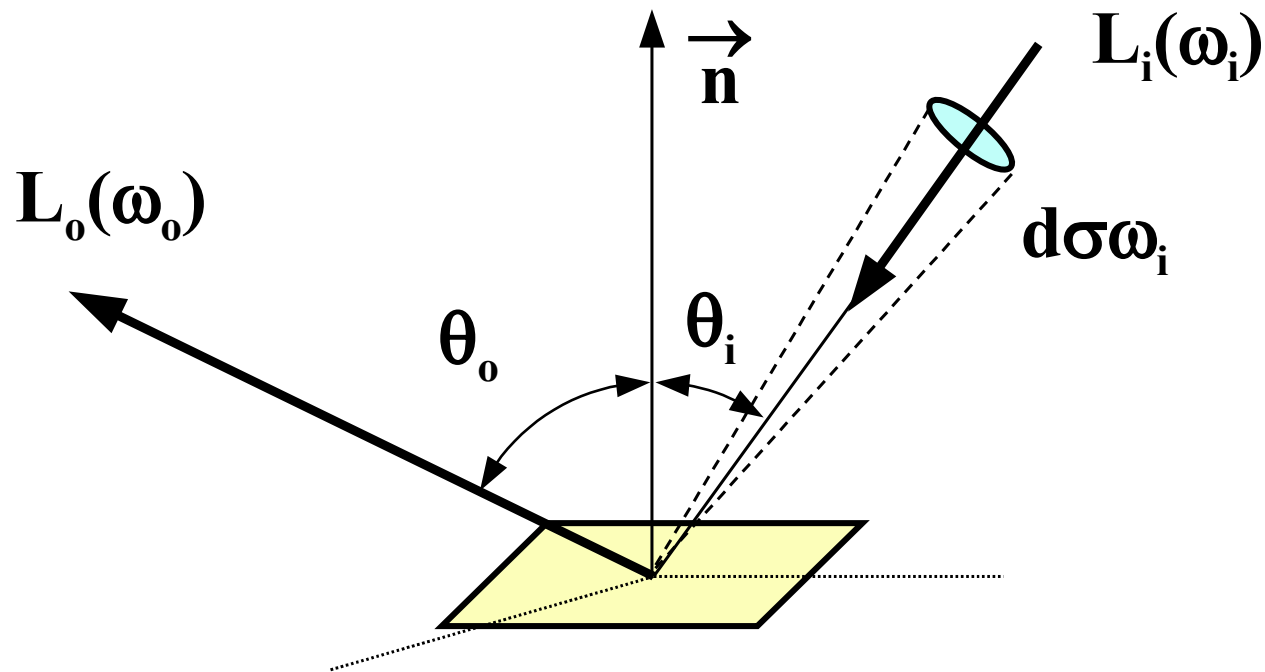
$$\Phi(x, \omega) \propto dA_{\omega}^{\perp}(x)$$



BSDF (lokální přenosová funkce)

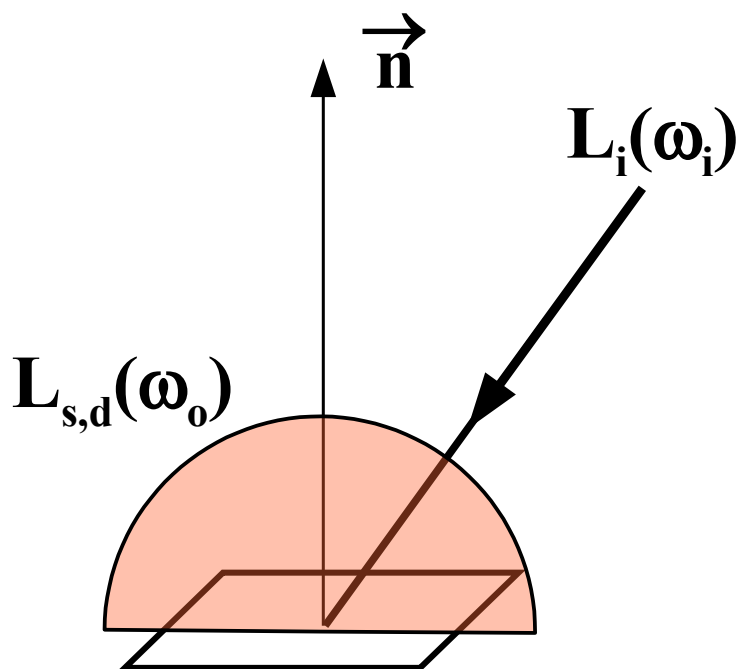


(„Bidirectional Scattering Distribution Function“)

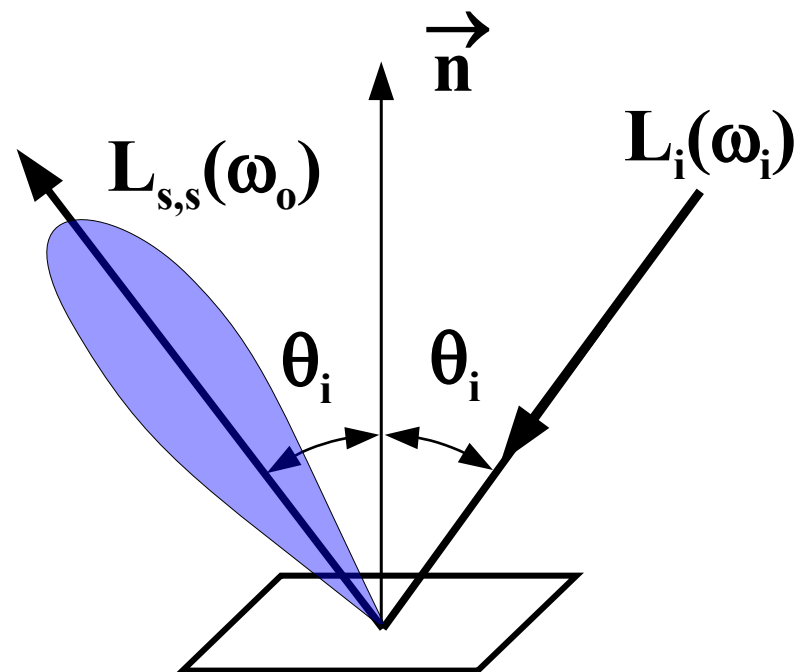


$$f_s(\omega_i \rightarrow \omega_o) = \frac{d L_o(\omega_o)}{d E(\omega_i)} = \frac{d L_o(\omega_o)}{L_i(\omega_i) \cos \theta_i d \sigma^\perp(\omega_i)}$$

Klasické složky odrazu světla



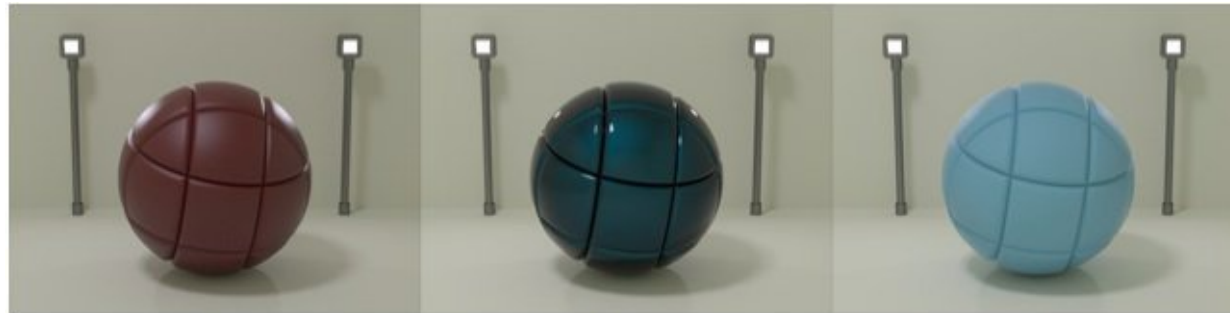
Difusní („diffuse“)



Lesklý („specular“)

$$f_s(\omega_i \rightarrow \omega_o) = f_{s,d}(\omega_i \rightarrow \omega_o) + f_{s,s}(\omega_i \rightarrow \omega_o)$$

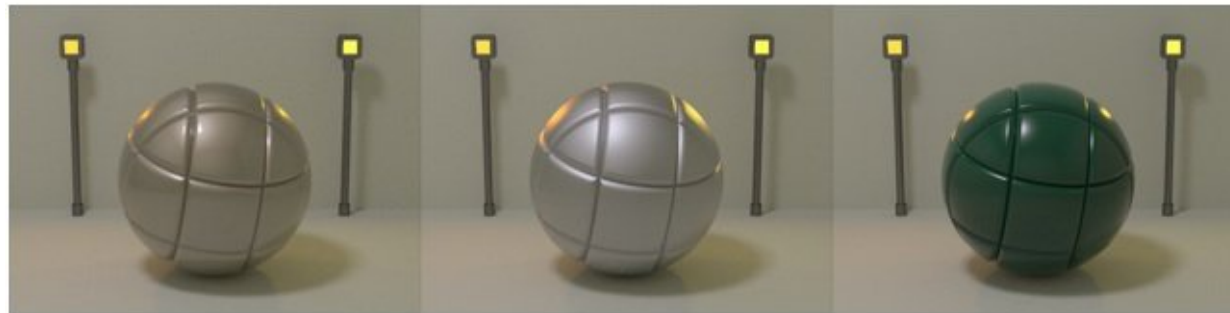
Příklady BSDF I



(a)

(b)

(c)



(d)

(e)

(f)



(g)

(h)

(i)

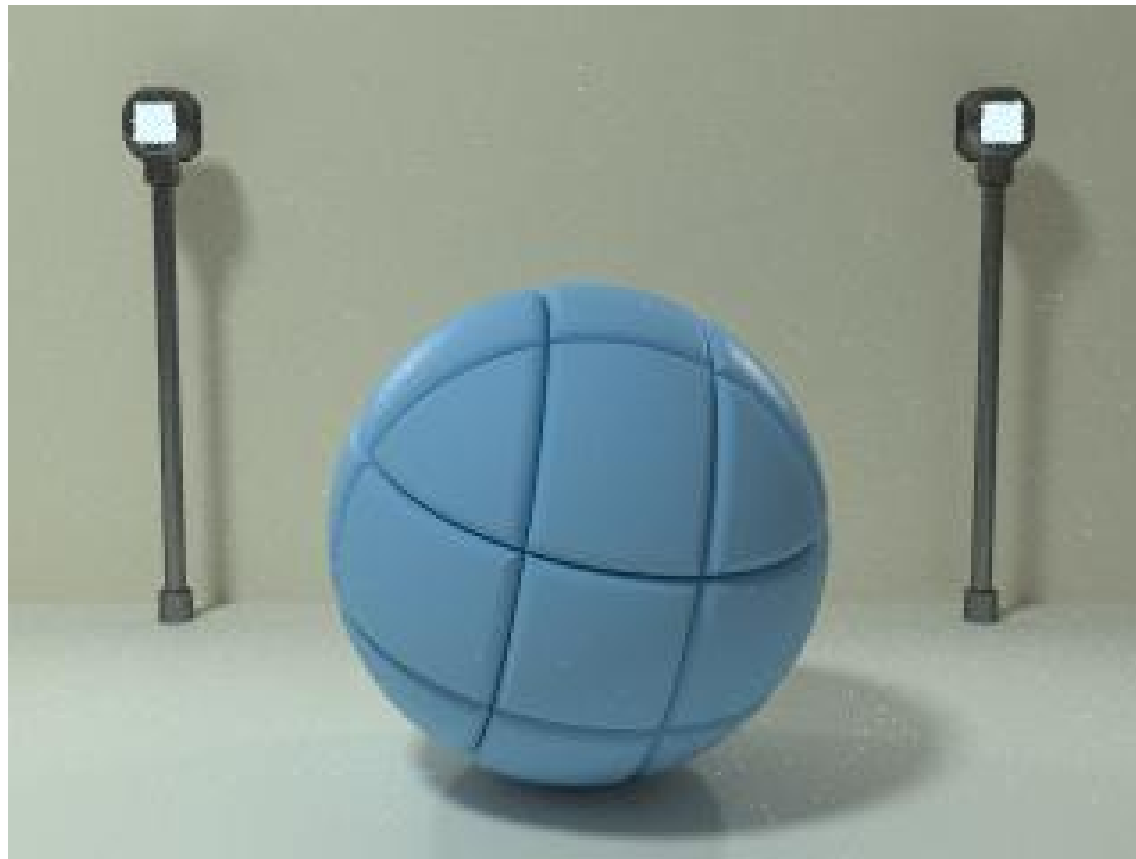
Příklady BSDF II



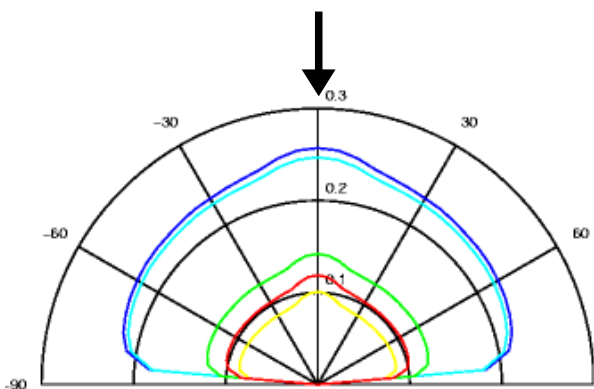
Příklady BSDF III – latexový nátěr



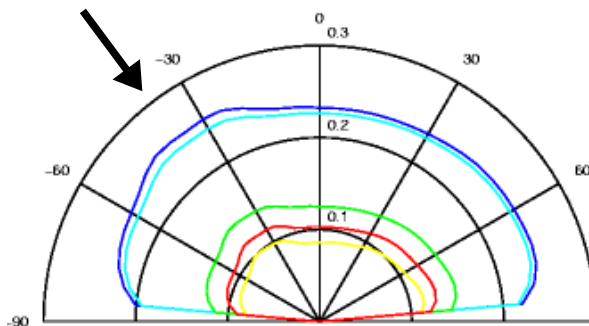
- stříkaný latexový (matnější) lak



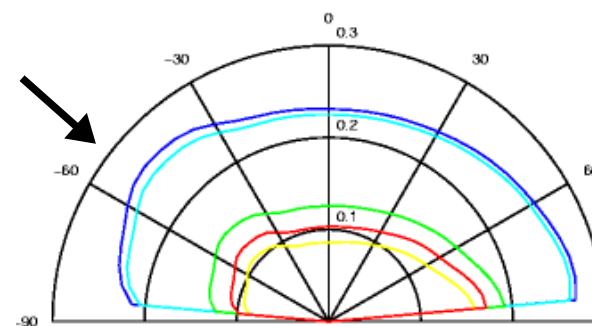
Příklady BSDF III – latexový nátěr



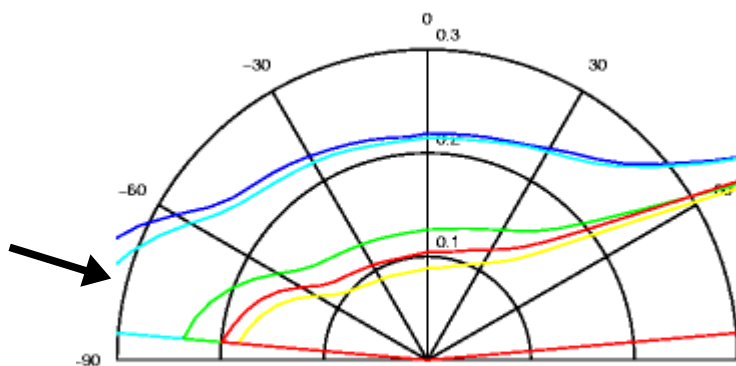
0° (kolmo)



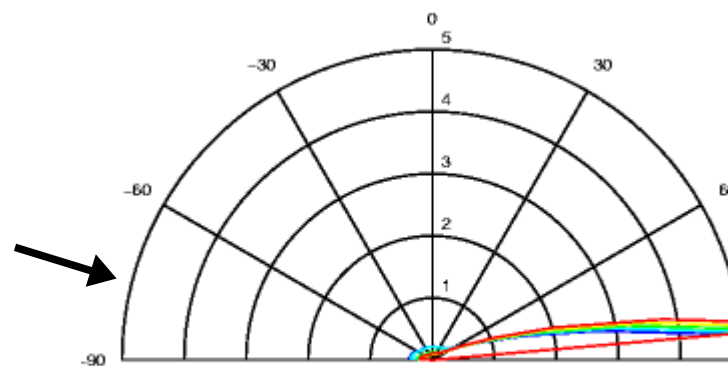
35°



55°








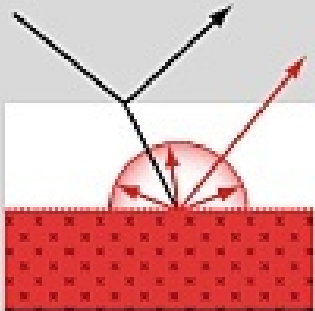
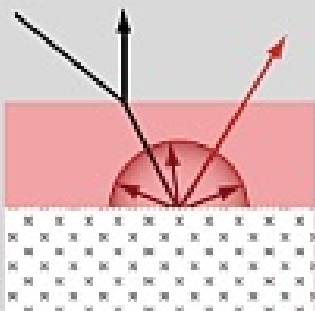
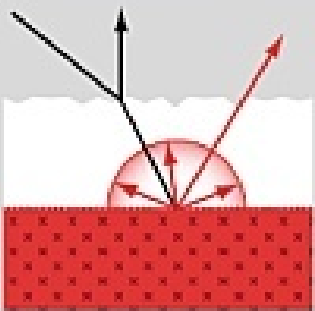
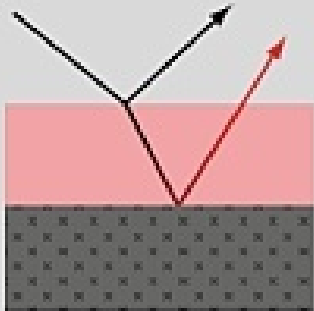
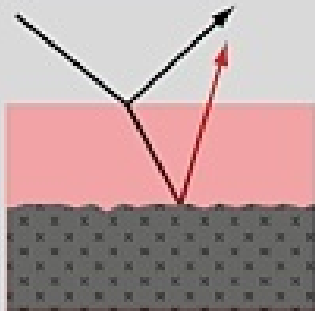







75°



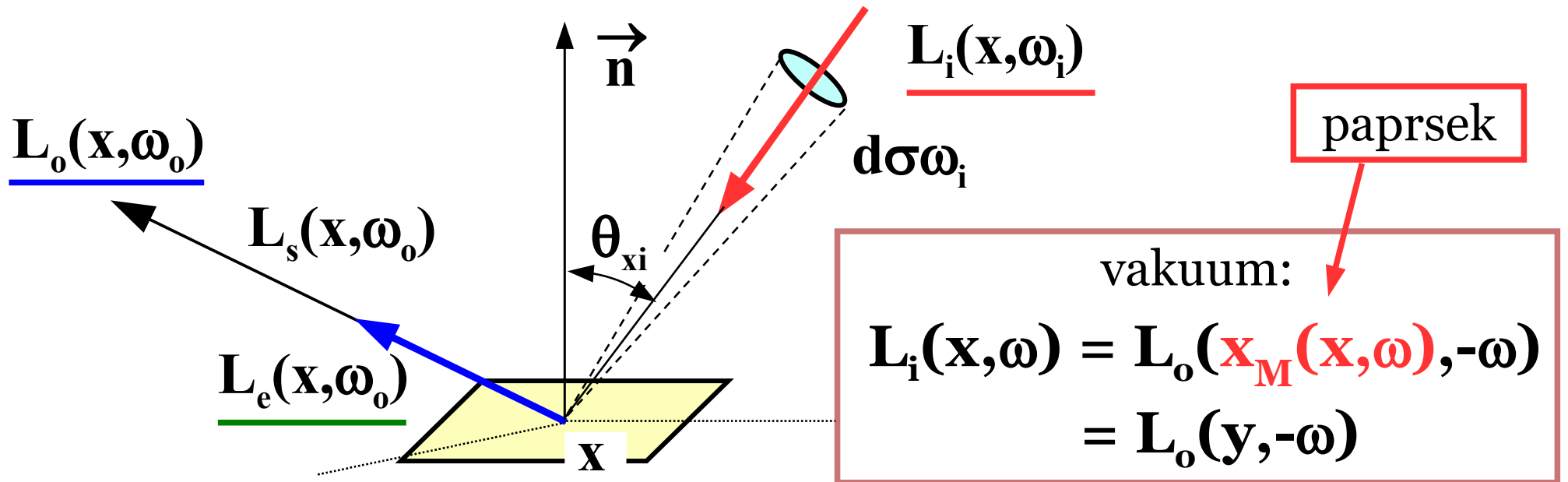
75° (zmenšeno)

Příklady BSDF IV – vrstvy



a)	b)	c)	d)	e)
Glossy Paint	Tinted Glazing	Frosted Paint	Metal Foil	Metallic Paint
				
				
Interfaces:	 Torrance-Sparrow	 Smooth	Materials:	 Coloured Solid
	 Torrance-Sparrow	 Smooth		 Metal
				 Colourle

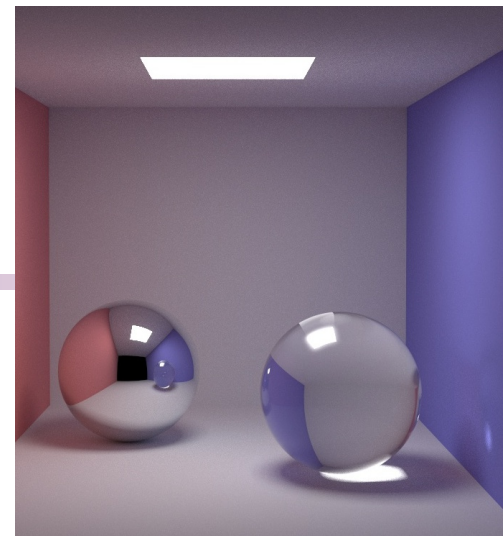
Lokální zobrazovací rovnice



vlastní vyzařování x

$$\underline{L_o(x, \omega_o)} = \underline{L_e(x, \omega_o)} + \int \underline{L_o(y, -\omega_i)} \cdot f_s(x, \omega_i \rightarrow \omega_o) \cdot d\sigma_x^\perp(\omega_i)$$

Monte-Carlo výpočet světla

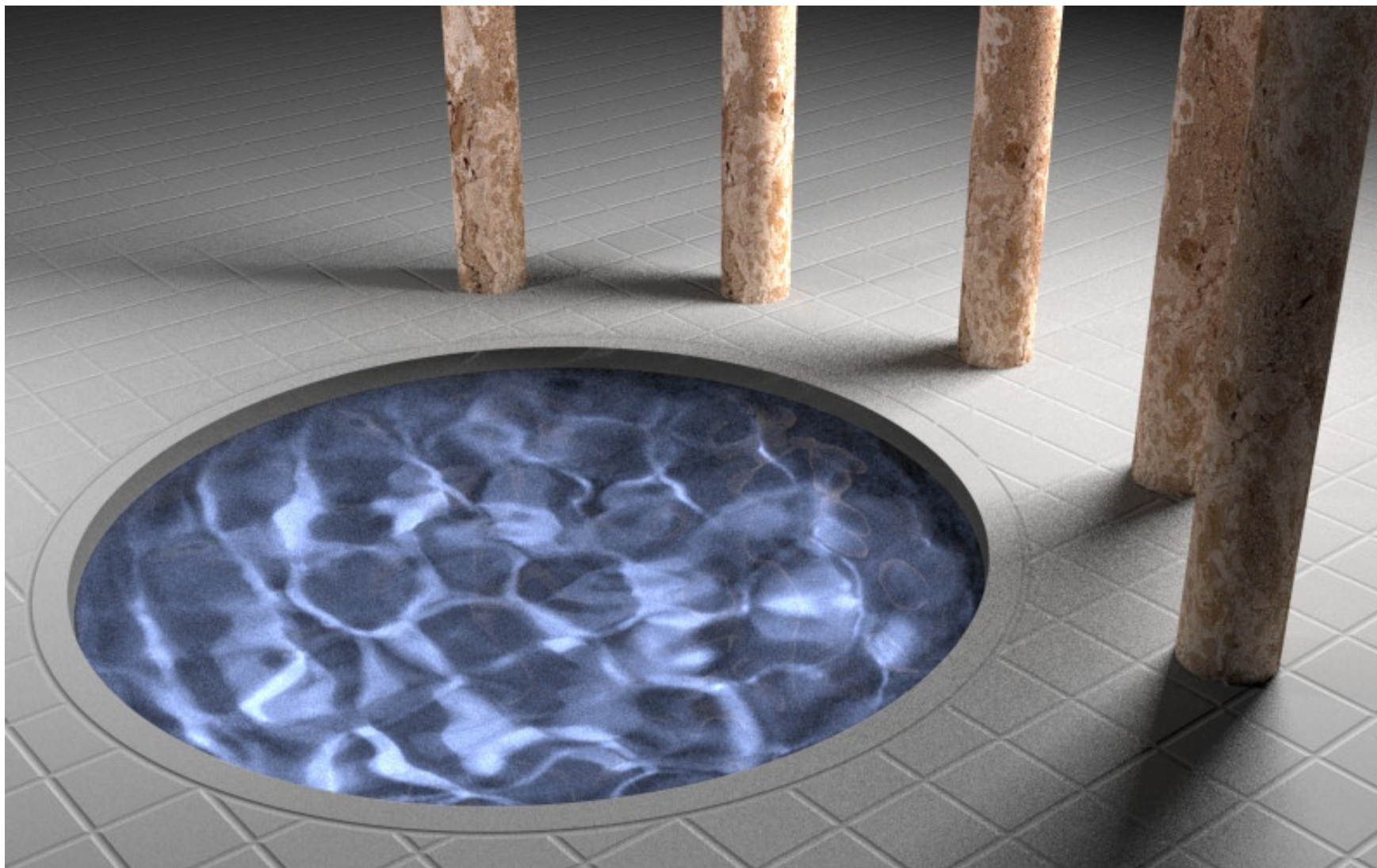


- ◆ **Monte-Carlo kvadratura:** integrály zobrazovacích rovnic jsou **mnoho-rozměrné**
 - ◆ anti-aliasing, hloubka ostrosti, rozmazání pohybem
 - ◆ Monte-Carlo metody nejsou citlivé na vyšší dimenze
- ◆ integrandy mají mnoho **nespojností** různých druhů
- ◆ obyčejně se **nepožaduje velká přesnost**
 - ◆ lidské vidění má velmi omezenou absolutní citlivost
 - ◆ běžně postačí relativní přesnost $1/2 \div 1 \%$

Příklady M-C zobrazování



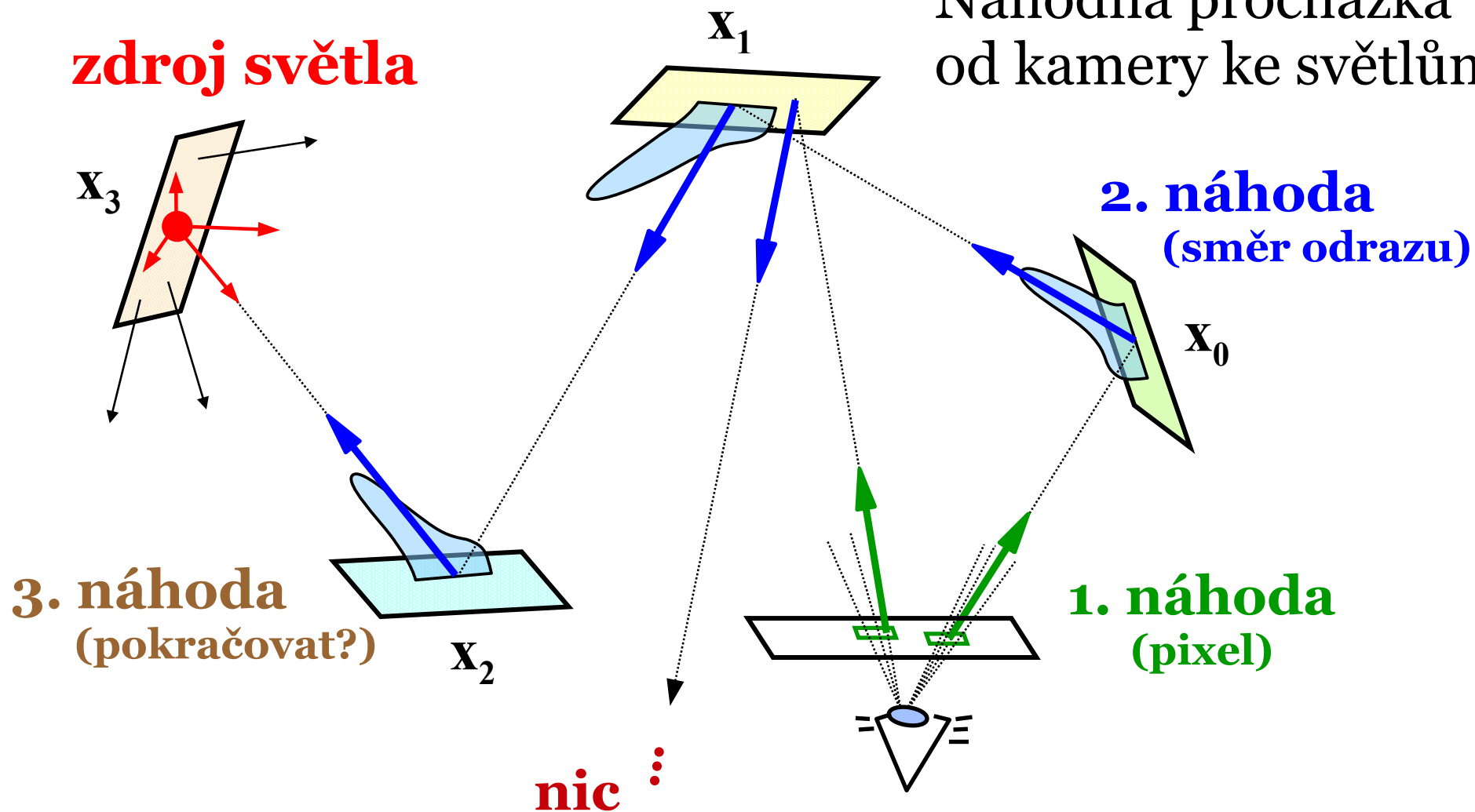
Příklady M-C zobrazování





Path-tracing – princip

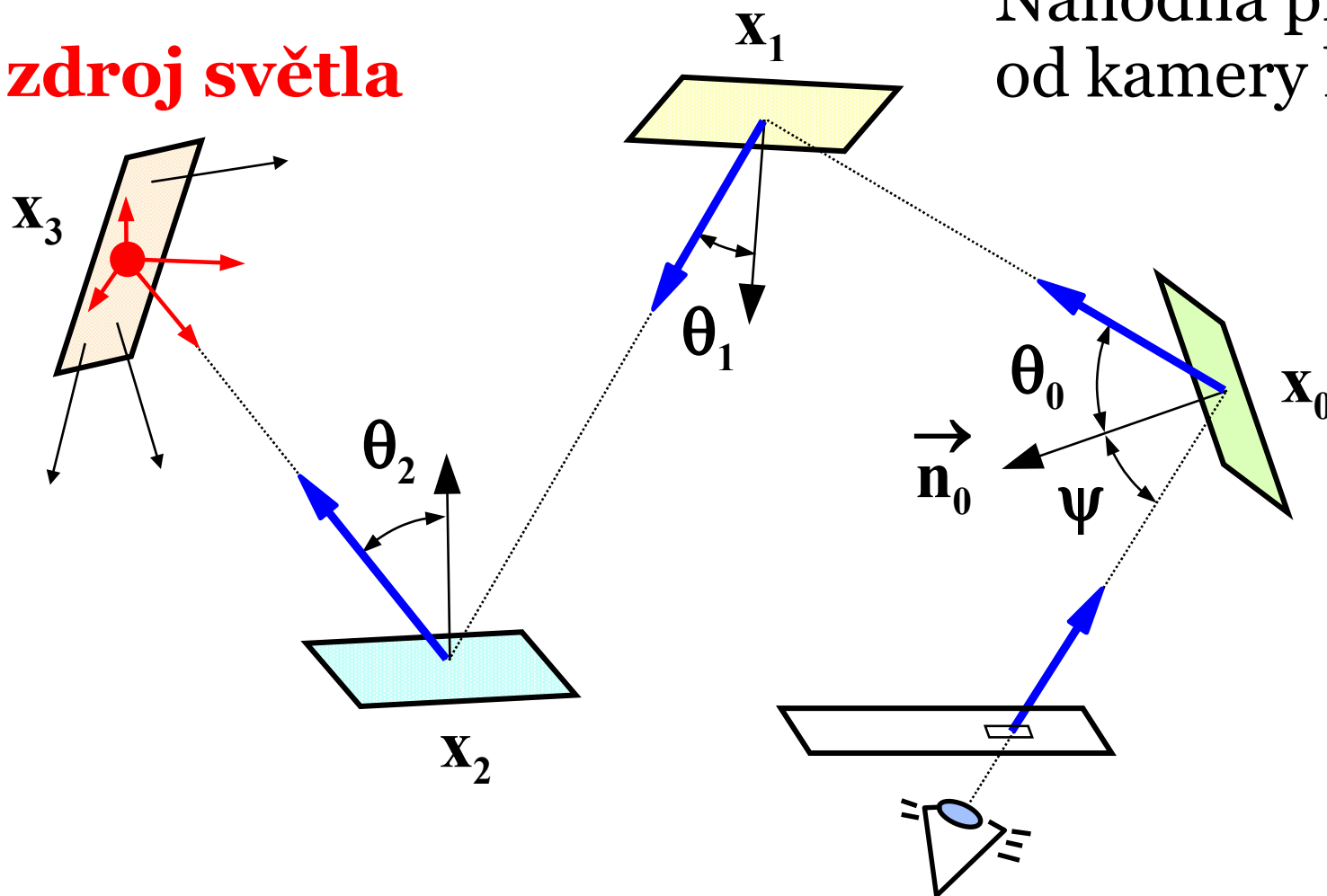
Náhodná procházka od kamery ke světlům





Path-tracing – postup výpočtu

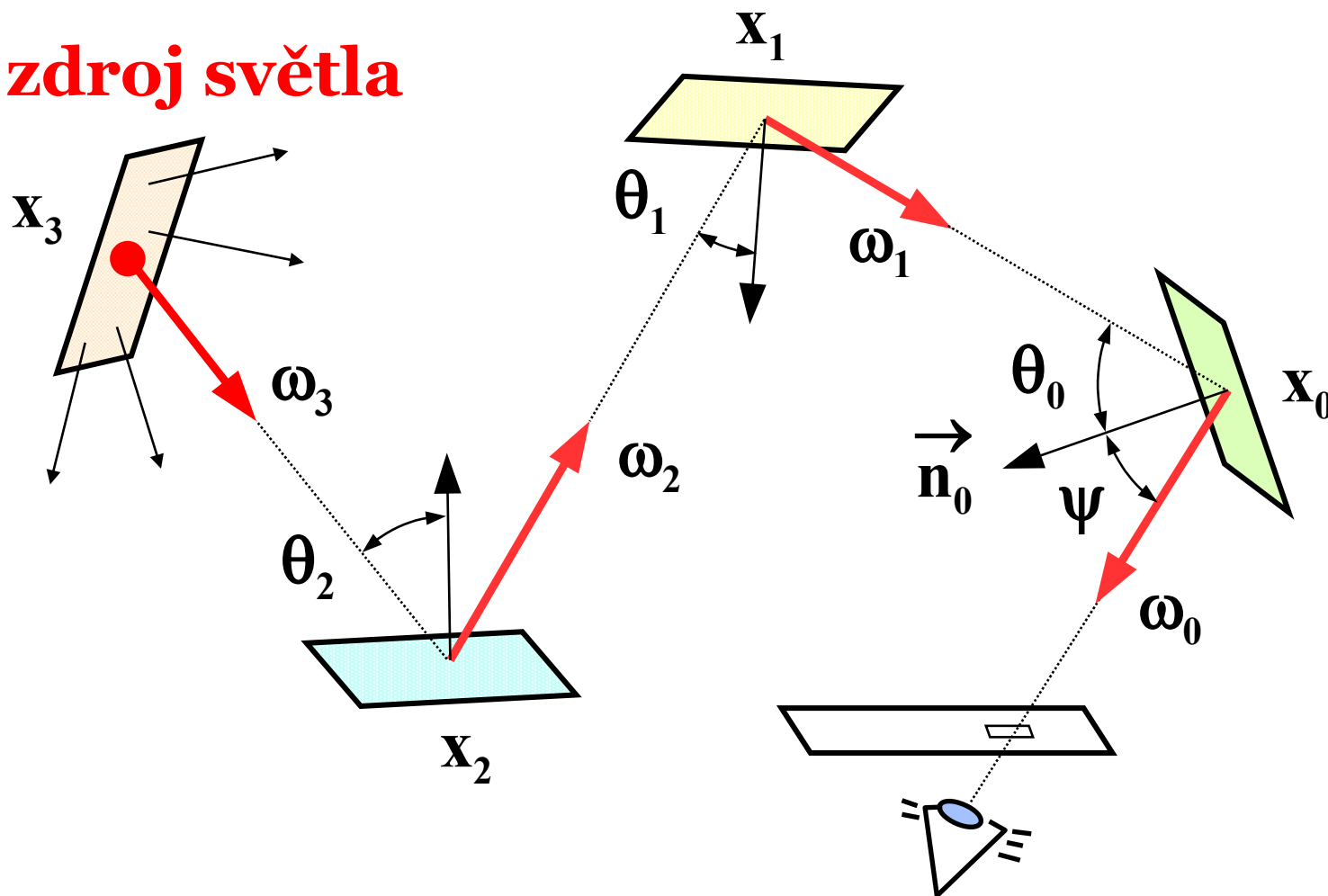
zdroj světla





Path-tracing – cesta „fotonu“

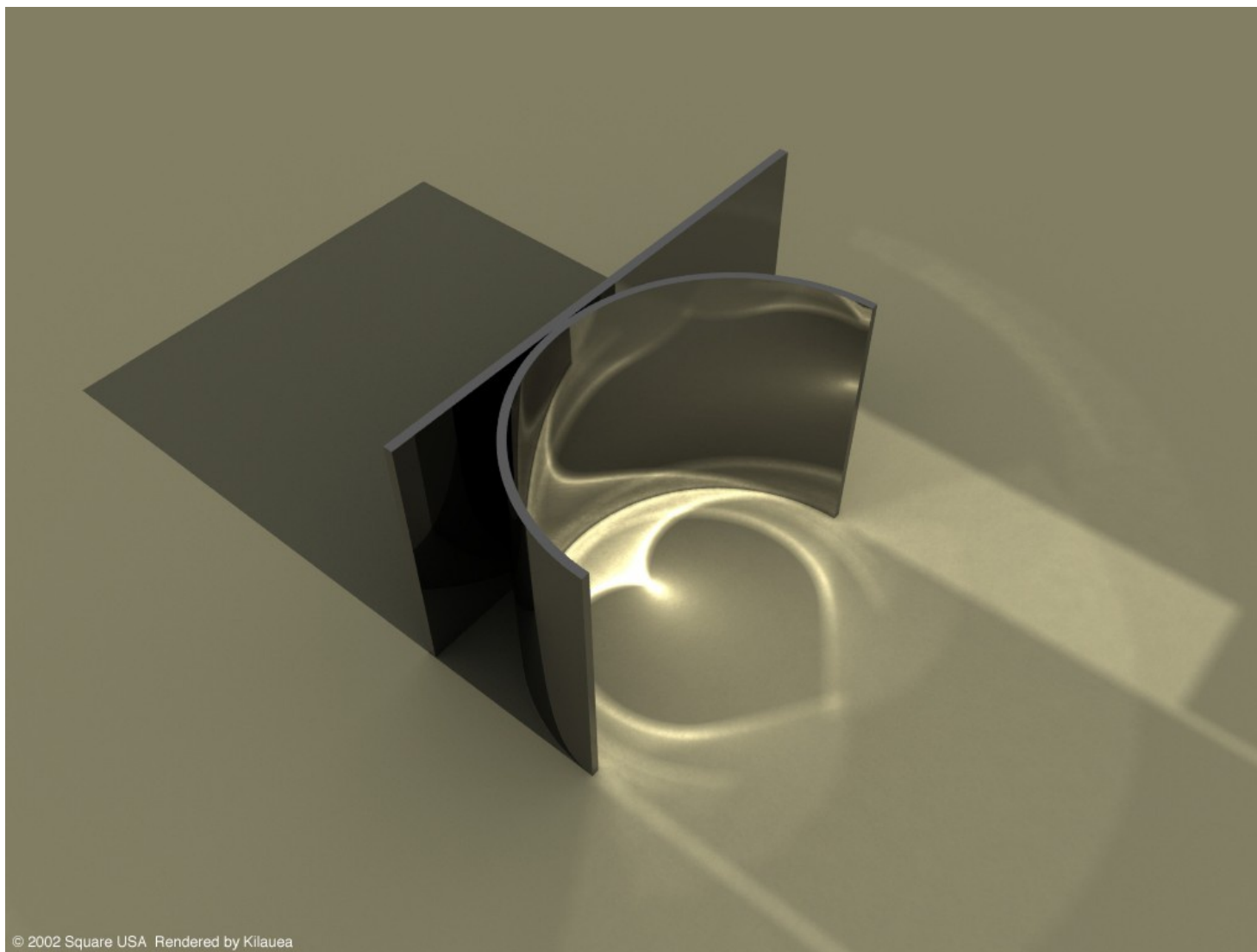
zdroj světla



Light-tracing - příklad



Photon-tracing - příklad (kaustika)



Bidirectional PT – příklady



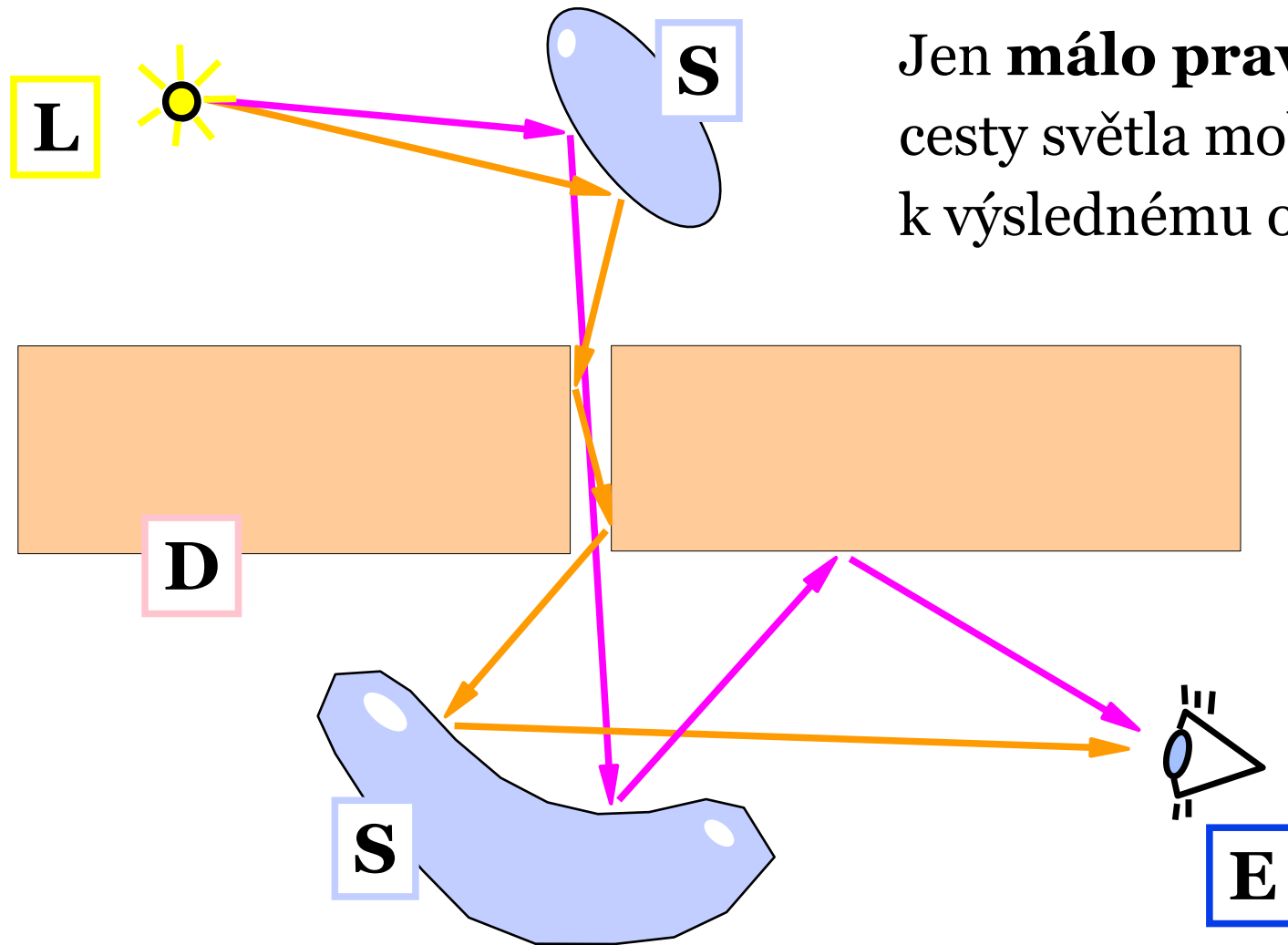
Škvíra ve dveřích (Metropolis)



“Veach door”



Problematická situace (Metropolis)



Jen **málo** pravděpodobné cesty světla mohou přispět k výslednému obrázku...

Path tracing (1h)

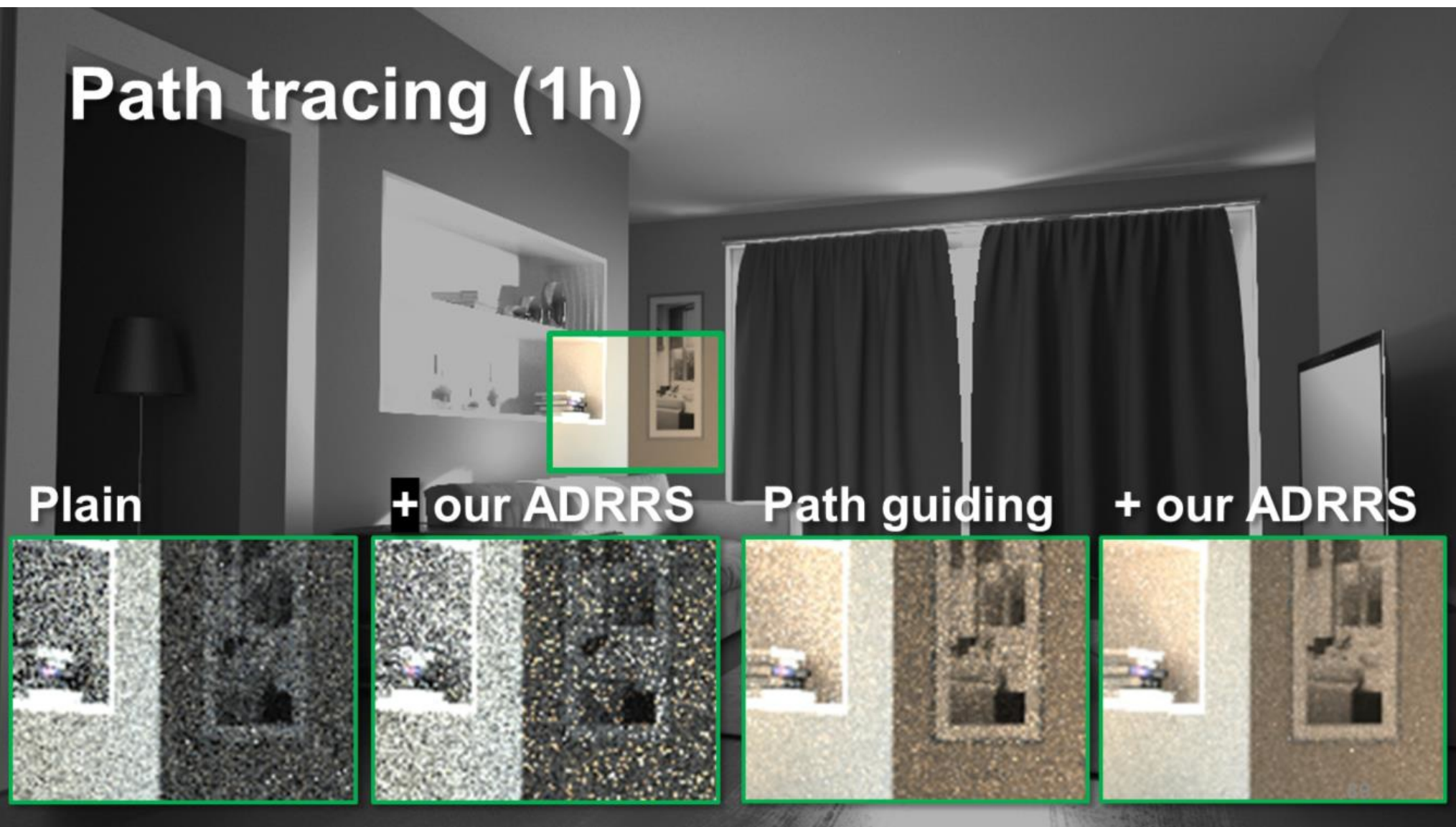
Our ADRRS + path guiding (1h)
in path tracing



Adjoint-Driven Russian Roulette and Splitting in Light Transport Simulation

- ◆ Jiří Vorba, Jaroslav Křivánek, SIGGRAPH 2016

Path tracing (1h)



<http://cgg.mff.cuni.cz/~jirka/papers/2016/adrrs/>

Děkuji

Computer
Graphics
Charles
University



Kontakty:

Jaroslav Křivánek

Josef Pelikán

Alexander Wilkie



<http://cgg.mff.cuni.cz/>

Literatura – knihy



- Andrew Glassner: ***Principles of Digital Image Synthesis***, Morgan Kaufmann, 1995
- Henrik Wann Jensen: ***Realistic Image Synthesis Using Photon Mapping***, A K Peters, 2001
- Matt Pharr, Greg Humphreys: ***Physically Based Rendering, 2nd Edition: From Theory To Implementation***, Morgan Kaufmann, 2010
- Philip Dutre, Kavita Bala, Philippe Baekert: ***Advanced Global Illumination***, A K Peters, 2006

Literatura



- Eric Veach: ***Robust Monte Carlo Methods for Light Transport Simulation***, PhD Thesis, 1997
- Henrik Wann Jensen et al.: ***A Practical Guide to Global Illumination using Photon Mapping***, SIGGRAPH 2002 Course
- Jiří Vorba, Jaroslav Křivánek: ***Adjoint-Driven Russian Roulette and Splitting***, SIGGRAPH 2016