3D Models: Material

Syllabus

- 3D scene
  - Object
    - Shape
    - **Material**
  - Camera
  - Light
- Rendering
- Image

- Material
  - Material observation
  - Physics (optics) models for CG
  - Material models in CG
    - Scattering models

Big picture


https://www.exp-points.com/asking-the-masters-material-art
SUBSTANCE
DESIGNER

Material is important for object appearance.
Without material...

Shape and material

- Rendering and creating objects in 3D scene to represent a real world objects requires:
  - **Shape modeling**
- **Shape** is needed to:
  - Define object form, size, etc.
  - Place object correctly in the scene with respect to other objects
  - Determine which objects are occluded and areas into which shadow is cast by object

https://polyhaven.com/a/ship_pinnace

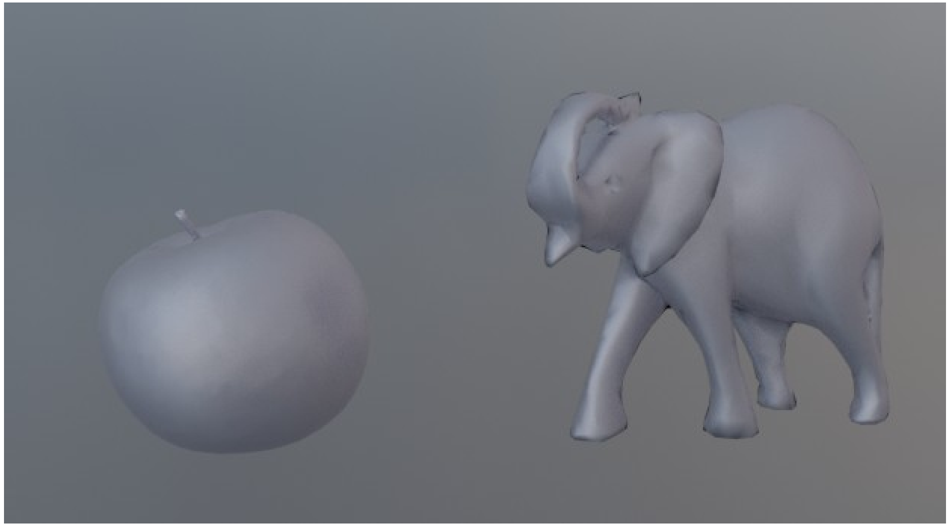
Shape and material

- Rendering and creating objects in 3D scene to represent a real world objects requires:
  - Shape modeling
  - **Visual appearance modeling → material**
- **Material describes light-object interaction: how object will look like**

https://polyhaven.com/a/ship_pinnace

Shape and material

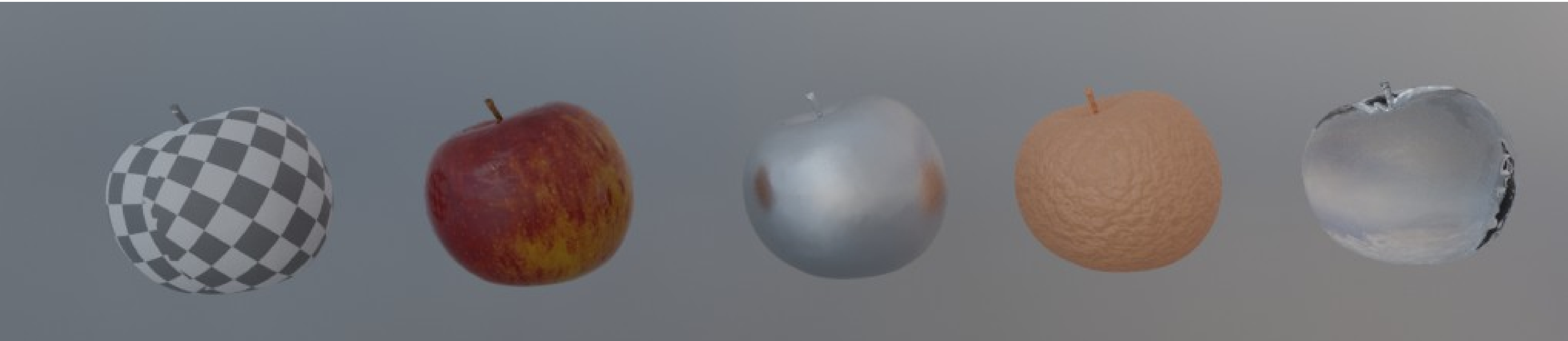
- 3D object: shape and material.
  - Material characteristics are independent of shape and position.
    - Material is modeled separately
  - It is enough to model how specific material generally interacts with light.
    - This model is then used for arbitrary shape.



Aluminum apple and aluminum statue – in both cases aluminum properties are the same. Also, changing position of aluminum apple in space doesn’t change aluminum properties.

"Let the form of an object be what it may, - light, shade and perspective will always make it beautiful" – John Constable

- Importance of material for appearance



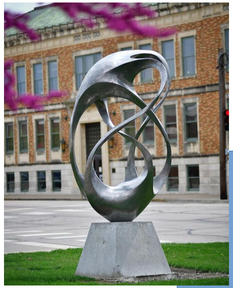
Model and environment texture from PolyHaven: https://polyhaven.com/a/food_apple_01

- Metal
- Plastic
- Glass
- Wood
- Fabric
- Stone
- Clouds
- Water
- Tree bark
- Leafs
- Plaster
- Paper
- Leather
- Sky
- Etc.

https://www.xomacenna.com/blog/a-guide-to-organic-modern-design

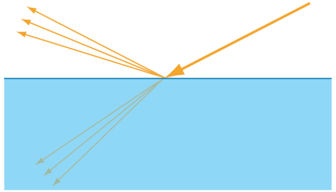
Appearance modeling

- Material modeling in computer graphics relies on:
  - Material observation
  - Physics (optics) models

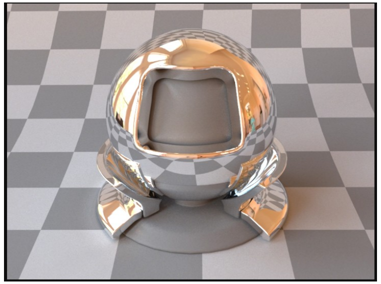


https://www.metalartstatue.com/amsculpture/cast-aluminum-sculpture/

Observation



Optics



Computer graphics model

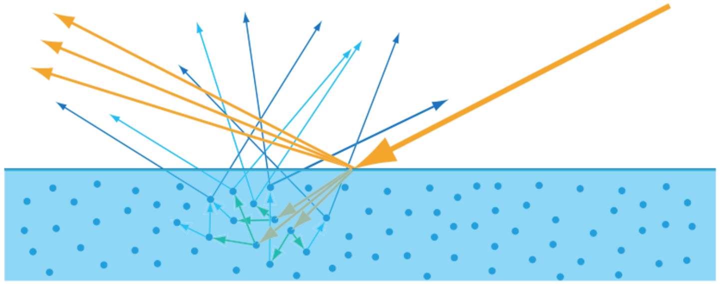
https://mitsuba2.readthedocs.io/en/latest/generated/plugins.html#bsdfs

Modeling and rendering

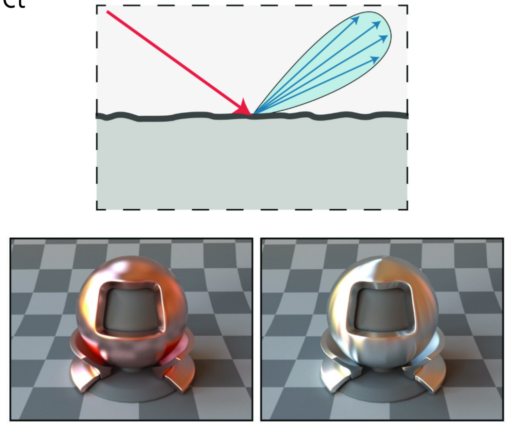
https://pbrt.org/

Materials: real world vs models

- Real world materials are very complex are simplified using different **models:**
  - **Physical models**: best description of real world materials
    - Example: geometric optics
  - **Computer graphics models:** simplification for creation and computational purposes
    - Example: separating objects into shape and material
  - **Artist observation:** based on perception – subjective model of the world
    - Artist draw what they see, not what is physically-correct



Physical model of light scattering



Computer graphics material model

Artist model:
https://shelleyhannafineart.com/painting-silver-objects/

Material observation

Modeling begins with **observation**

Observation

- Observation goals:
  - Understand **what makes each material look different** than other materials
  - Observe **characteristics which are responsible for object appearance**




https://www.metalartstatue.com/amsculpture/outdoor-metal-bird-sculpture/

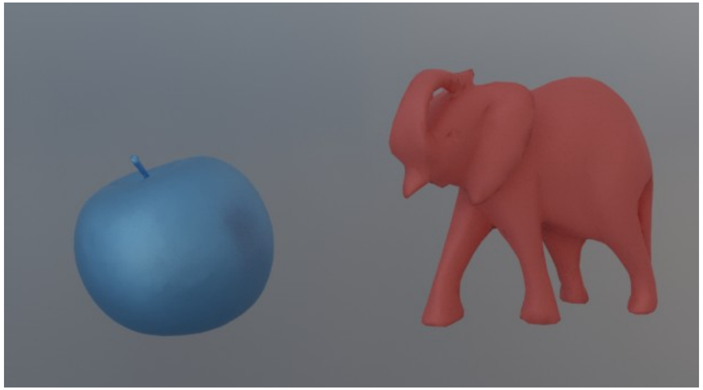
Observation

- Characteristics responsible for object appearance:
  - **Shape**: large scale form or geometry of object.
  - **Material**: for computer graphics modeling purposes: fine-scale geometrical variations and substance properties
  - **Illumination:** size, direction, color, etc.
  - **Sensor/Perception:** point of view, camera properties, etc.

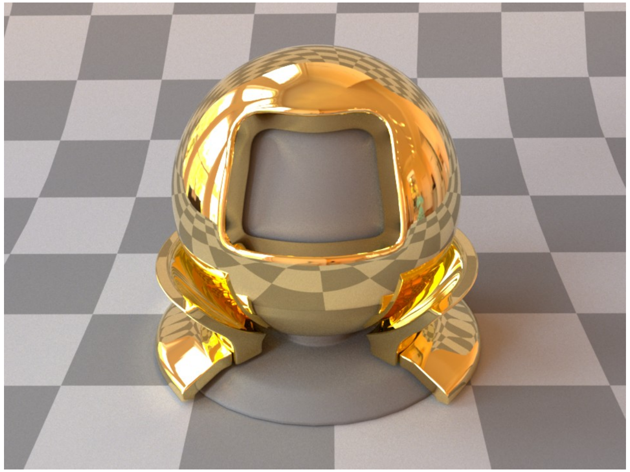
Classifying materials

- Classifying materials enables us to **understand which characteristics are needed to be modeled** in order to obtain required appearance. We can classify any material by following variations:

Spectral: color



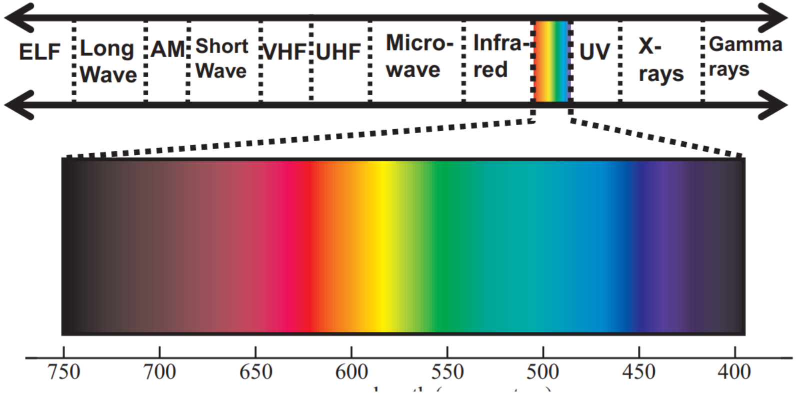
Directional

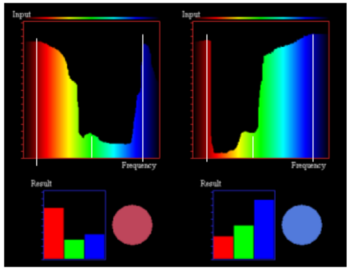


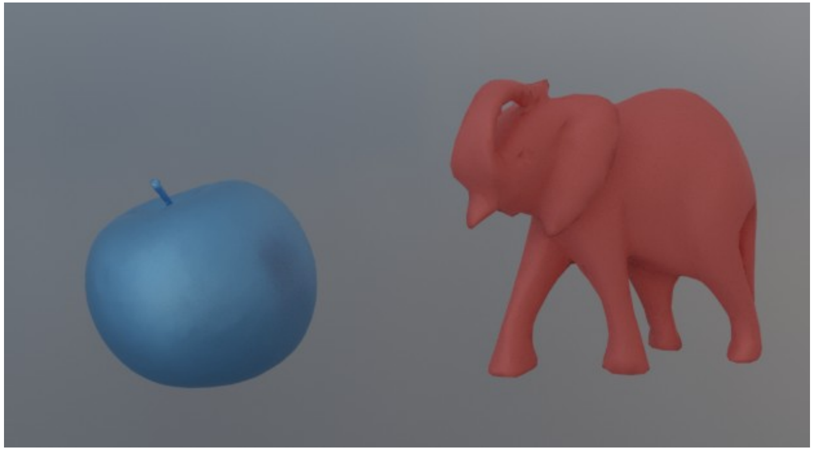
Spatial

Spectral/color material characteristic

- **Light** is a electromagnetic radiation in the wavelength band 380-780nm
- Light scattered from object surface (its material) is described using:
  - **Color** is described with red, green and blue floating points (R, G, B) in [0,1]
  - **Brightness** is floating point value [0, inf]

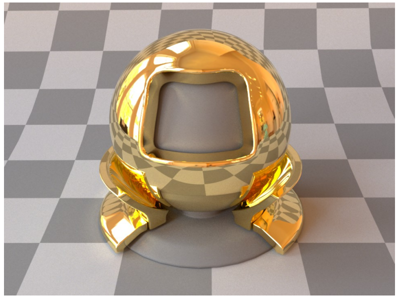




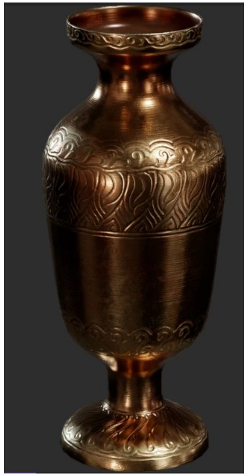


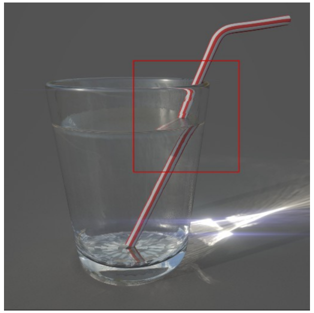
https://blog.selfshadow.com/publications/s2013-shading-course/hoffman/s2013_pbs_physics_math_notes.pdf
https://www.researchgate.net/publication/234814482_Digital_modeling_of_the_appearance_of_materials

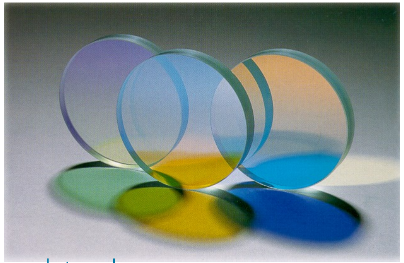
Directional material characteristic



- Result from **directionality of light scattering** by the object
- **Directional effects:** whether we see changes in object as we change view (or light or object position)
  - Those effects are due to structures at a length scale shorter than the visible to the eye → can be observes on completely flat surfaces (what we consider flat by looking at it)
- Directional effects that are attached to object (do not depend on environment):
  - Shiny
  - Hazy
  - Glossy
  - Matte
  - Dull
  - Translucent
  - Transparent





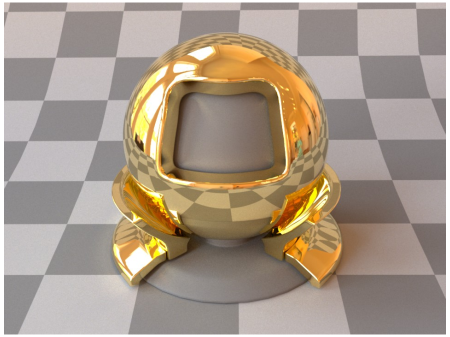


https://substance3d.adobe.com/tutorials/courses/the-pbr-guide-part-1
https://polyhaven.com/

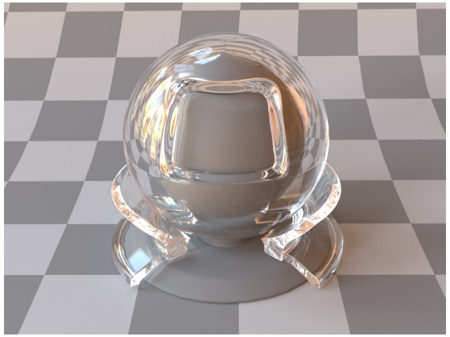
https://en.wikipedia.org/wiki/Transparency_and_translucency

Directional material characteristic

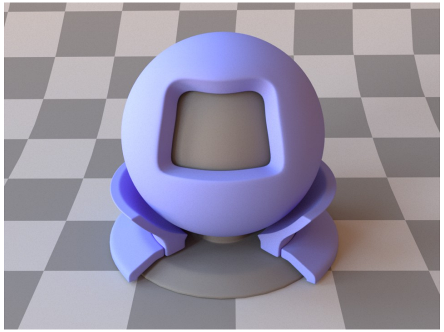
- In computer graphics, higher level description of scattering is used to describe directional – **surface scattering** - characteristics:
  - Reflective
  - Transmissive
  - Specular (mirror)
  - Glossy
  - Lambertian (diffuse)
  - Refractive
  - Retroreflective



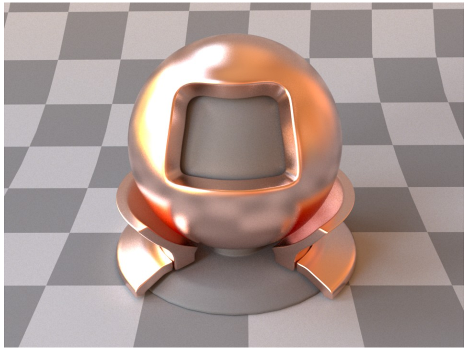
Reflective, specular (conductor, gold)



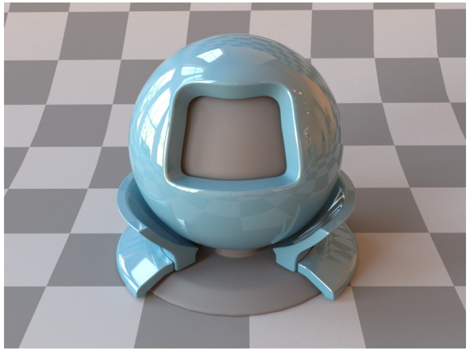
Transmissive, specular (dielectric, glass)



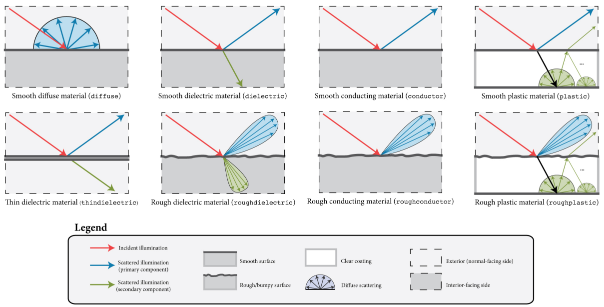
Reflective, Lambertian/diffuse (dielectric, plastic)



Reflective, glossy

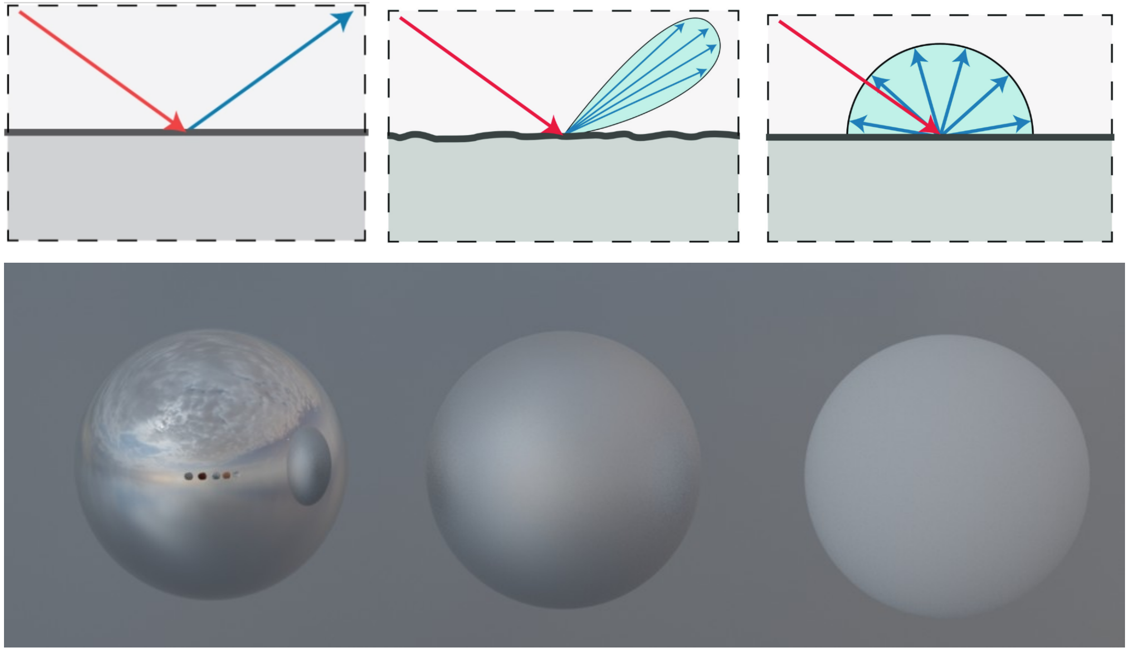


Refractive, glossy (dielectric shiny plastic)



Surface scattering: scattering function

- Surface directional effects are modeled using **scattering function.** Most important light scattering models are:
  - **Specular** (mirror direction)
  - **Glossy** (preferred direction)
  - **Diffuse** (all directions)

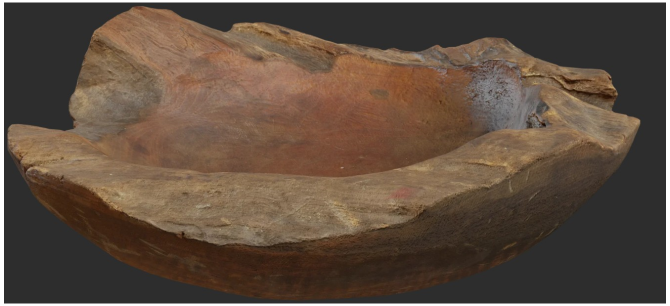


Material characteristics: texture

- Visual variations on the object surface, much smaller scale than size of the object but larger than the wavelength of light - **spatial variations**:
  - Directional or spectral (color)
  - Small scale geometric variations: bumps and pores
- Surfaces with spatial variation are observed as non-uniform: **texture or pattern** can be observed

Color variation

Directional variation

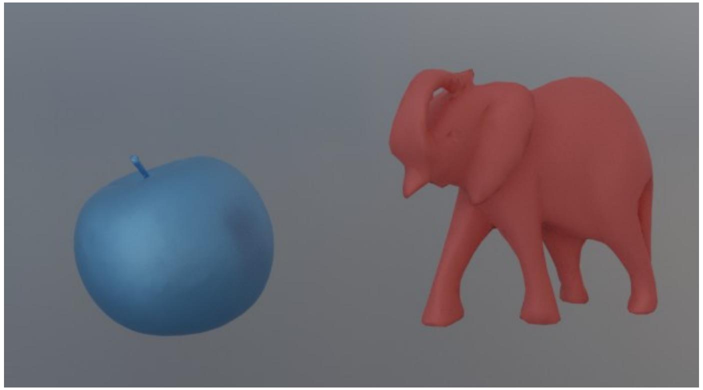


Small scale geometrical variation

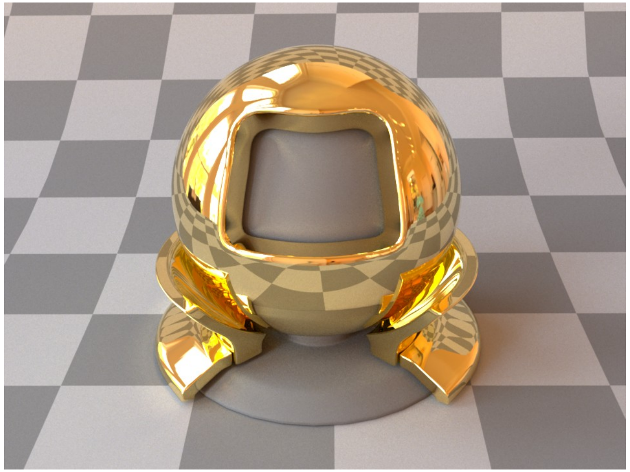
Classifying materials

- Classifying materials enables us to **understand which characteristics are needed to be modeled** in order to obtain required appearance. We can classify any material by following variations:

Spectral: color



Directional



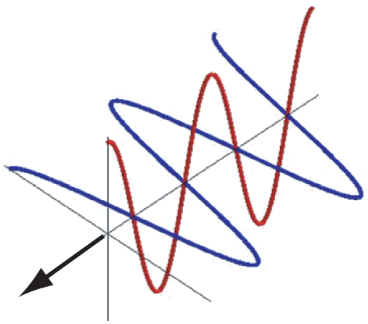
Spatial

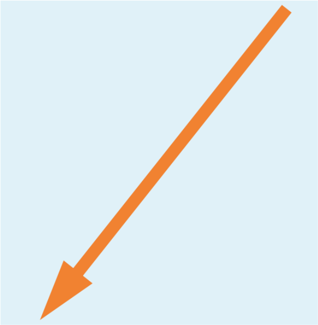
Physical models for computer graphics

Modeling is founded on **optics**

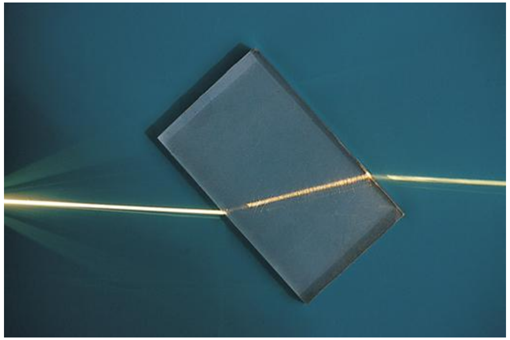
Light-matter interaction

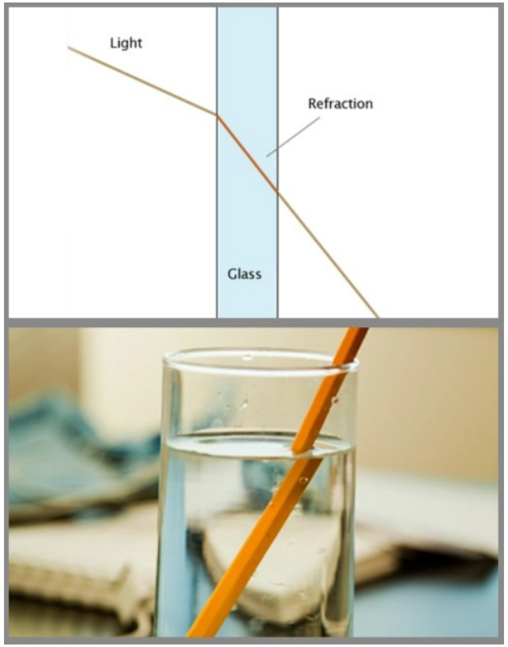
- Light is electromagnetic wave → too complex for CG!
- **Geometrical optics** approximates **light as rays** and effect that material has on light using **index of refraction** (IOR) – a complex number:
  - Real part determines speed of light → **scattering**
  - Imaginary part determines **absorption** of light





Light is an electromagnetic transverse wave which can be represented as a ray.





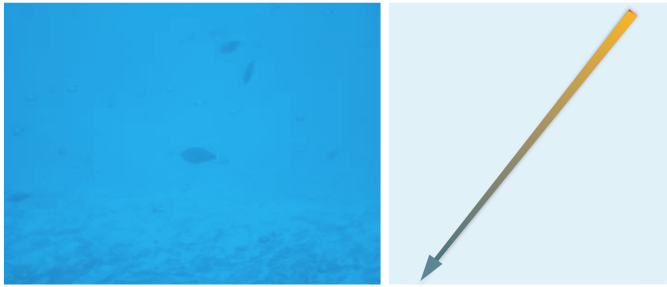
https://manual.keyshot.com/manual/materials/general-parameters/refractive-index/

Light-matter interaction

- Simplest light-matter interaction is light propagating through **homogeneous medium**.
  - **Uniform IOR**: light travels on a straight line. It can only be **absorbed:** direction is same, intensity might be attenuated
  - Examples: transparent materials; glass or water.

Transparent media: straight line with same intensity (scattering only happens when light crosses air-glass-water homogeneous media since they have different (but constant) IORs.

Clear absorbent media: straight line with loss of intensity – selective absorption (color is changed)

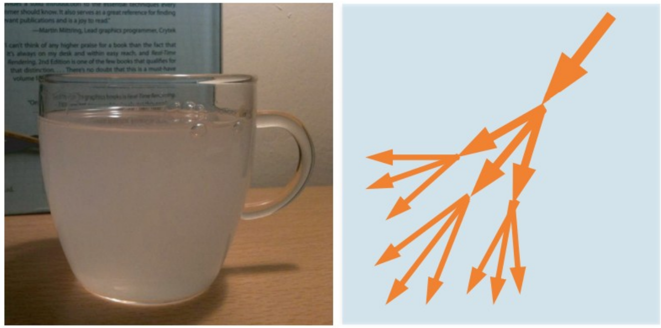


Slight absorption becomes significant with distance

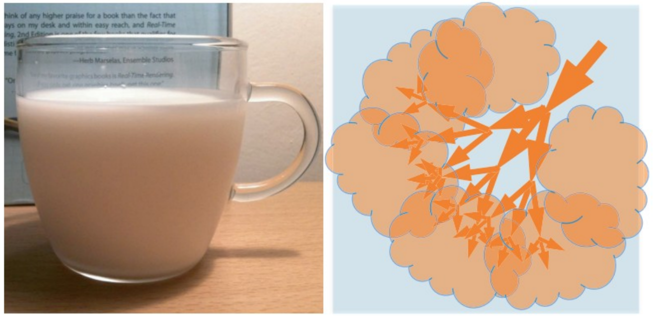
https://blog.selfshadow.com/publications/s2013-shading-course/hoffman/s2013_pbs_physics_math_notes.pdf

Light-matter interaction

- In **Heterogeneous medium** IOR varies which causes **light scattering**
  - Direction of light is changed but not the intensity
  - Light can scatter in all directions, mostly non-uniformly: forward or back scattering (in or reverse of incoming direction)



Microscopic particles cause varying of IOR and light to scatter continuously in all possible directions.



Translucent or **opaque materials** → light is scattered so much that we can not see (clearly) through the object

Longer distances cause more scattering (e.g., clean air)

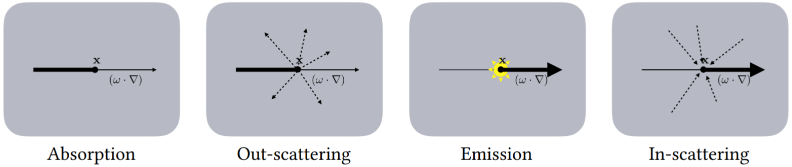
https://blog.selfshadow.com/publications/s2013-shading-course/hoffman/s2013_pbs_physics_math_notes.pdf

Light-matter interaction: scattering and absorption

- Light traveling through medium, based on **index of refraction** will:
  - Absorb
  - Scatter
- Appearance depends on both scattering and absorption

https://blog.selfshadow.com/publications/s2013-shading-course/hoffman/s2013_pbs_physics_math_notes.pdf

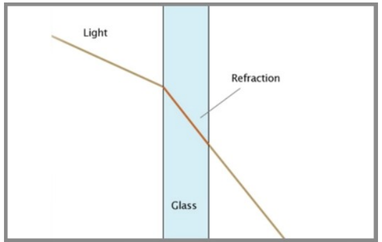
Light-matter interaction



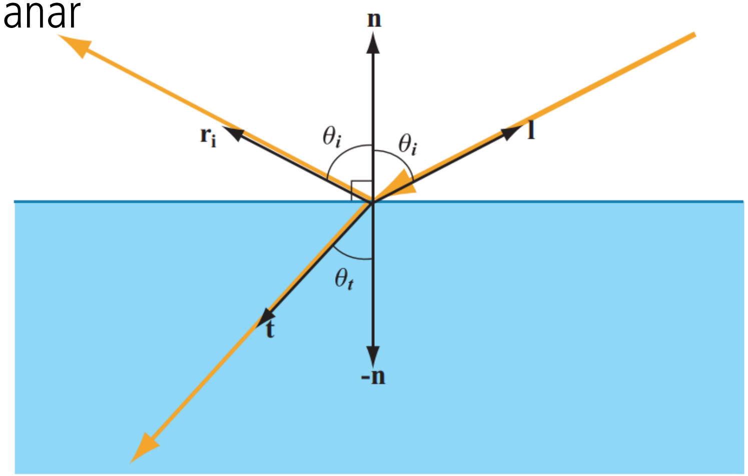
- Light scattering inside medium in CG is called **volumetric scattering**
- **Participating media**
  - Volume between objects

https://cs.dartmouth.edu/wjarosz/publications/jarosz08radiance.html
https://studios.disneyresearch.com/2012/08/05/virtual-ray-lights-for-rendering-scenes-with-participating-media/
https://graphics.pixar.com/library/ProductionVolumeRendering/paper.pdf

Light-matter interaction: between media



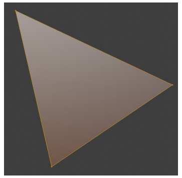
- Light **traveling between media of different index or refraction -** Maxwell's equations – too computationally heavy!
- Simplification and assumptions:
  - Geometrical optics: rays light representation, IOR
  - Interface between (volumes) media is perfectly (optically) flat, planar boundary* → **object surface**
- Solution: **Fresnel equations and Snell's law**
  - Describe light **reflection** and **refraction** direction and amount

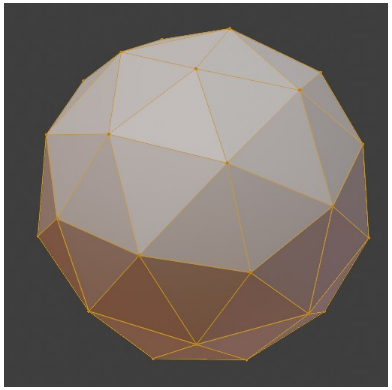


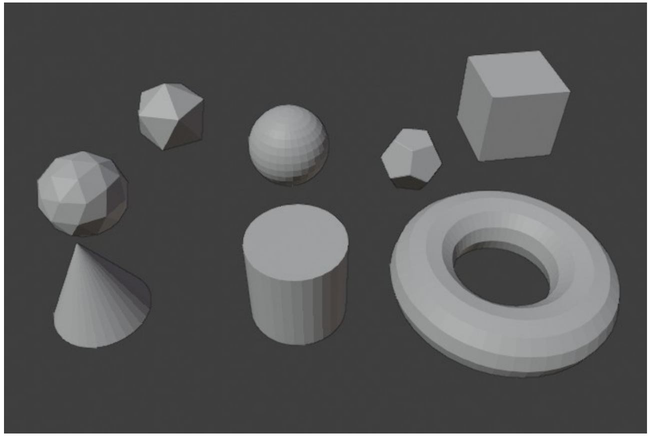
* Surface should be infinetly large, but in comparison with wavelenght of light, surface real objects can be considered as such.

Surface

- Surface is thin interface between two media
- In CG it is described with shape representations, e.g., polygonal mesh

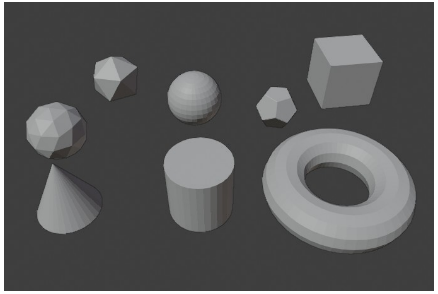


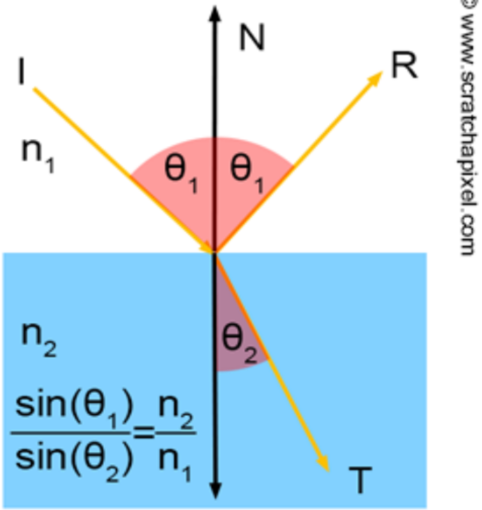




Light-matter interaction: flat surface

- Surface: 2D interface separating volumes with different IOR
- Behavior of light falling on surface depends on:
  - Geometry → surface orientation (normal)
  - Substance → IOR
- Light falling on planar surface can:
  - Reflect
    - Direction: law of reflection
    - Amount: Fresnel equations
  - Refract (transmit)
    - Direction: Snell's law
    - Amount: Fresnel equations





Snell's law

$$\frac{\sin\theta_1}{\sin\theta_2} = \frac{\eta_2}{\eta_1}.$$

Fresnel equations

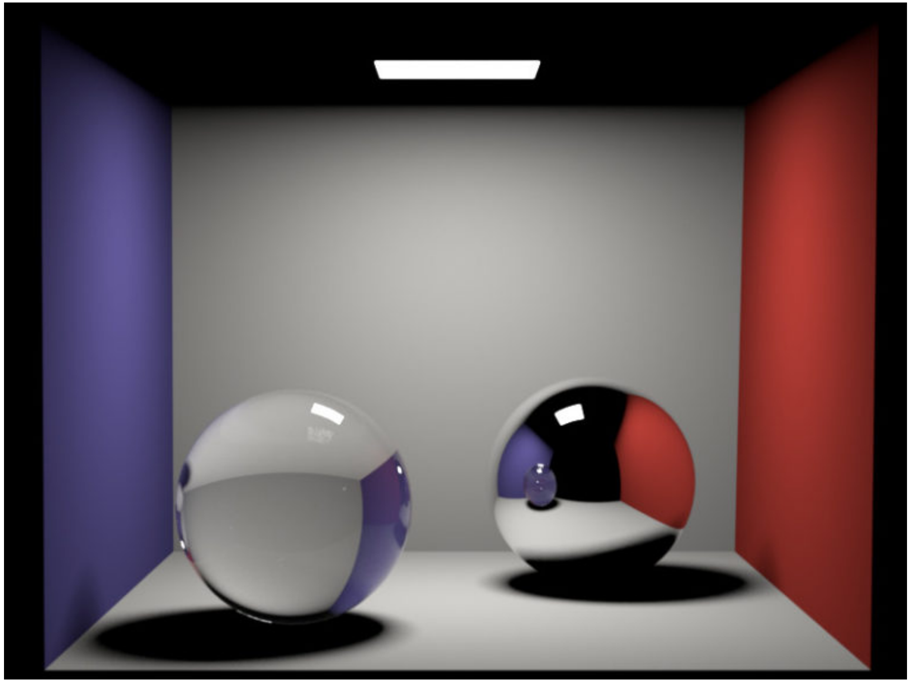
$$F_{R\parallel} = \left(\frac{\eta_2\cos\theta_1 - \eta_1\cos\theta_2}{\eta_2\cos\theta_1 + \eta_1\cos\theta_2}\right)^2,$$

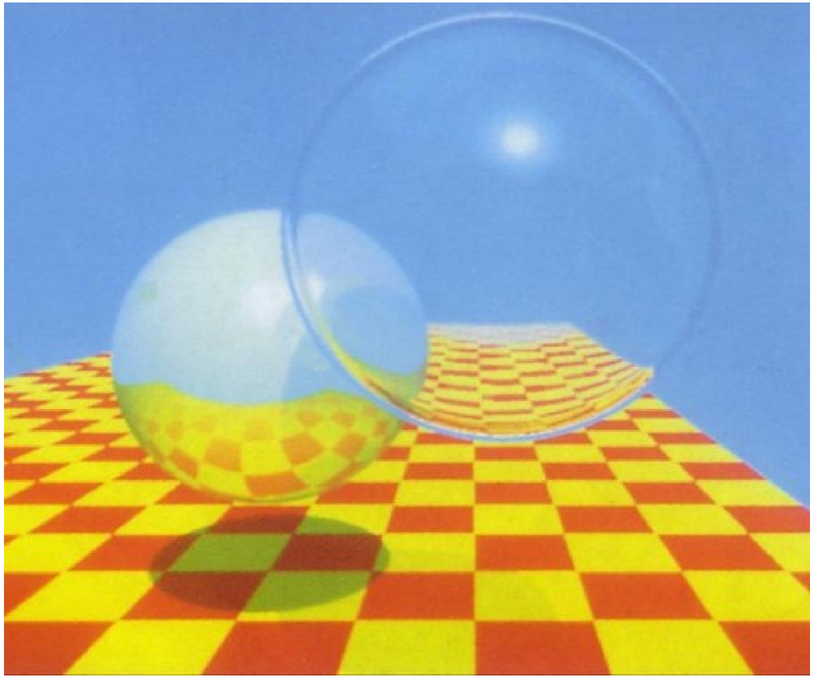
$$F_{R\perp} = \left(\frac{\eta_1\cos\theta_2 - \eta_2\cos\theta_1}{\eta_1\cos\theta_2 + \eta_2\cos\theta_1}\right)^2.$$

$$F_R = \frac{1}{2}(F_{R\parallel} + F_{R\perp}). \qquad F_T = 1 - F_R.$$

Light-matter interaction: flat surface

- Surface reflection and refraction is crucial for realistic synthesis
  - Whitted ray-tracing for realistic image synthesis

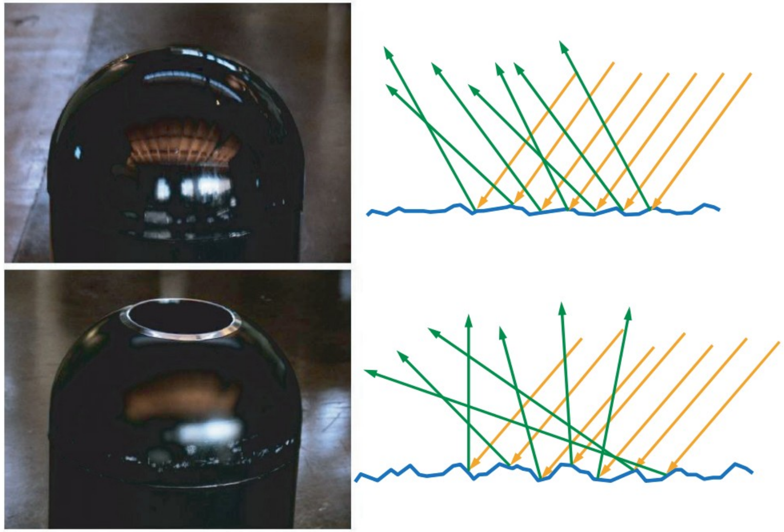




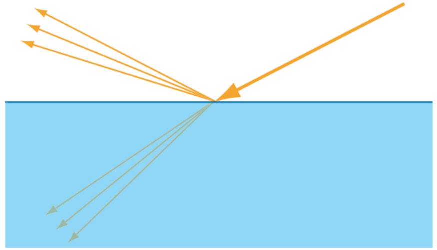
http://www.jallmenroeder.de/2020/11/18/path-tracing-with-nori/

More physically correct surface reflection

- Real surfaces: small geometrical irregularities* not visible by eye but affecting light reflection
- Surface model: large collection of tiny optically flat surfaces – facets. Final appearance is aggregate result of relevant facets.
  - Smaller deviation of those facets cause more **mirror-like surface reflection** (small roughness)
  - Larger deviation of those faces causes more **blurred surface reflection** (glossy, high roughness)



Both surfaces look smooth but surface on bottom is rougher. Roughness difference is on microscopic scale - **microgeometry**.



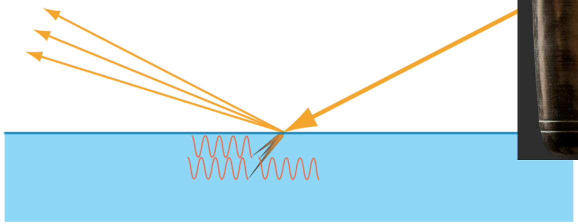
Macroscopically, non-optically flat surface can be treated as flat surfaces reflecting with multiple directions

* not optically flat surfaces: irregularities are larger than wavelength (causing light to reflect differently) and too small to render since this interaction happens under one pixel.

Surface refraction

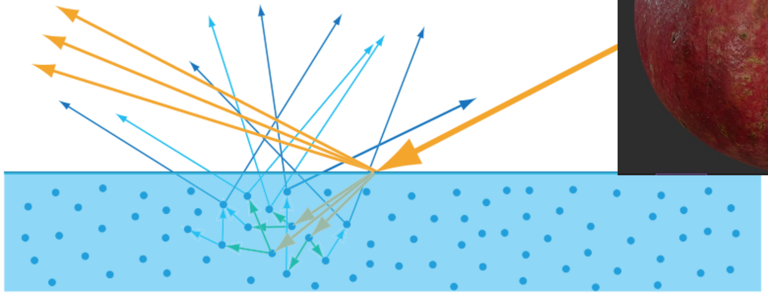
- Besides of reflection on surface, light can also **refract** inside object.
- Amount and direction of refracted light depends on material which we can separate in:

Metals (conductors)



In case of **metals**, most of the light is reflected and rest is immediately absorbed. That is why mirrors are made using metal foundation. Conductors are spectrally selective and thus reflection color may vary

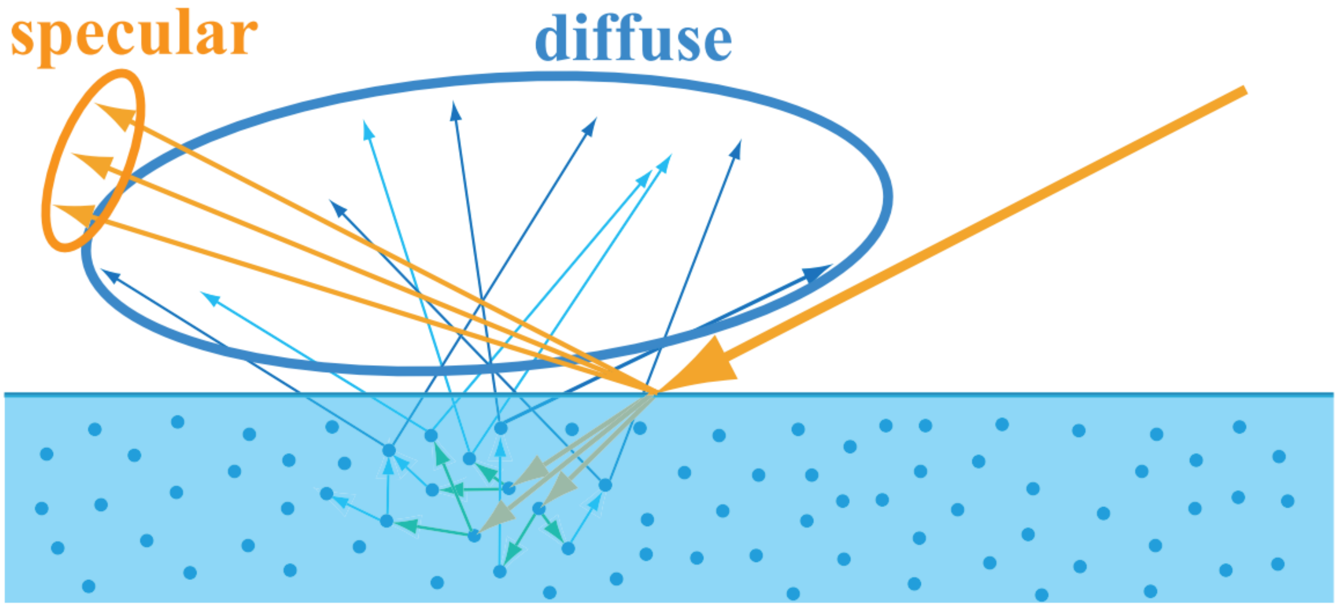
Dielectrics (non-metals)



In case of **dielectrics**, light partially reflects and partially refracts. Refracted light is then absorbed and scattered inside surface (**sub-surface scattering (SSS)**) causing **diffuse** reflection.

Diffuse and specular reflection

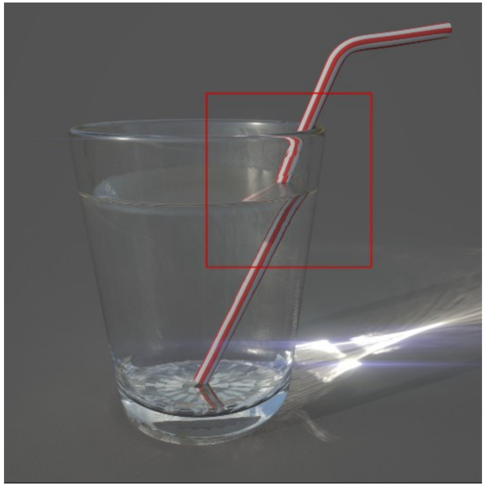
- Two fundamental **light-surface** scattering processes:
  - Diffuse
  - Specular/glossy



Surface refraction

- Refracted light on dielectric surface can be:
  - Absorbed and re-emitted: **opaque** objects
  - Transmitted: **transparent** objects
  - Partially transmitted: **translucent** objects

Opaque



Transparent

Translucent

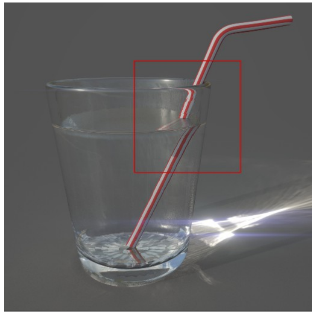
https://www.hippopx.com/en/query?q=translucent

Light-matter interaction: summary

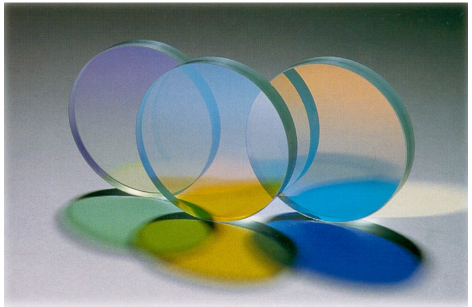
- Light traveling to/between objects can be scattered/absorbed → **volumetric scattering**
- Light falling on object surface – **surface scattering** - can:
  - **Reflect** (metals and dielectrics)
  - **Refract** (only dielectrics)
- Depending on material, refracted light can:
  - **Transmit** (transparent surfaces)
  - **Sub-surface scatter** (opaque and translucent object volume) → **volumetric scattering**

Participating media

Opaque dielectric: reflect + refract → SSS/diffuse



Transparent dielectric: reflect + refract → transmit

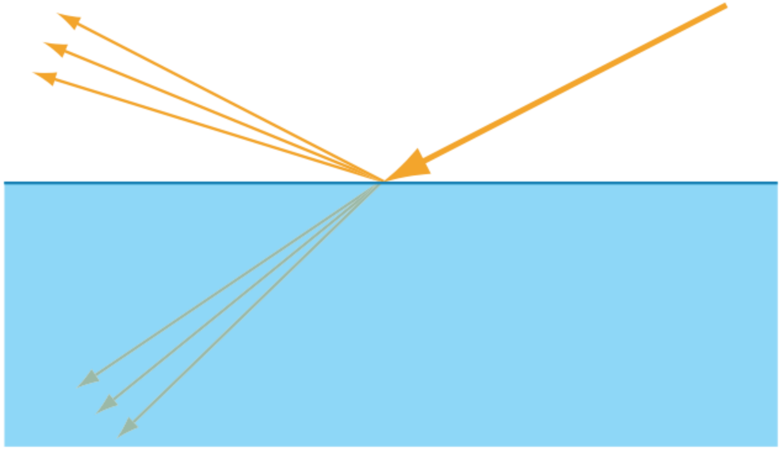


Translucent dielectric: reflect + refract → transmit

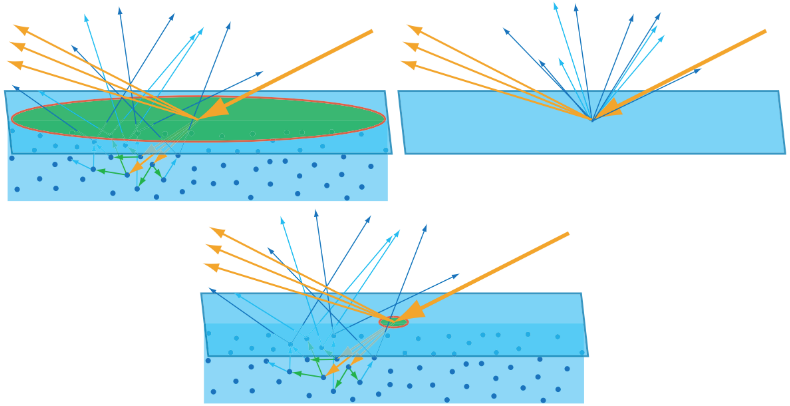
Metal: reflect

Local vs global models

- Local (direct) models: describe what happens when light falls on **one point**
- Volumetric scattering phenomena requires global model
  - Can be approximated with local model (e.g., SSS with diffuse model)



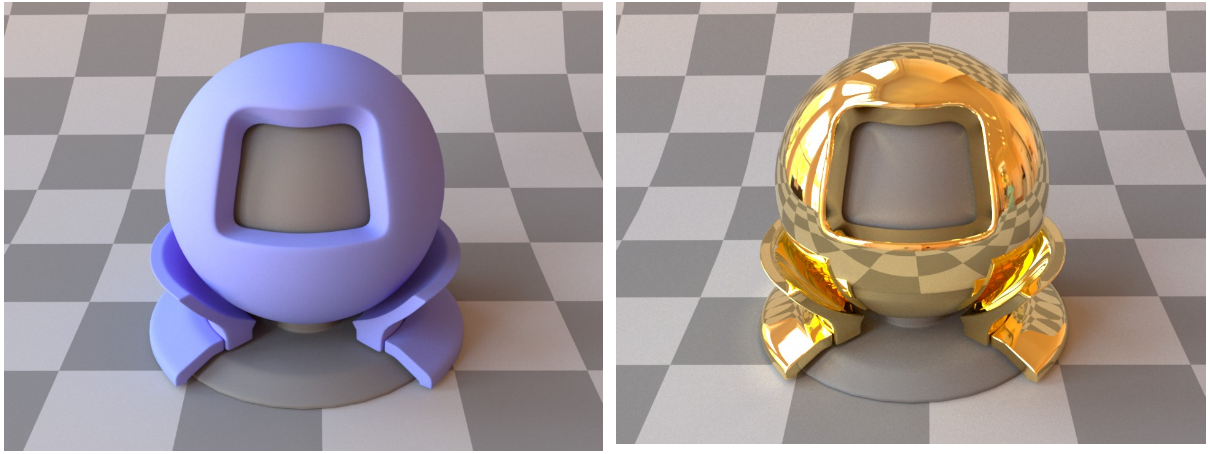
Local models describe what happens at one point/pixel



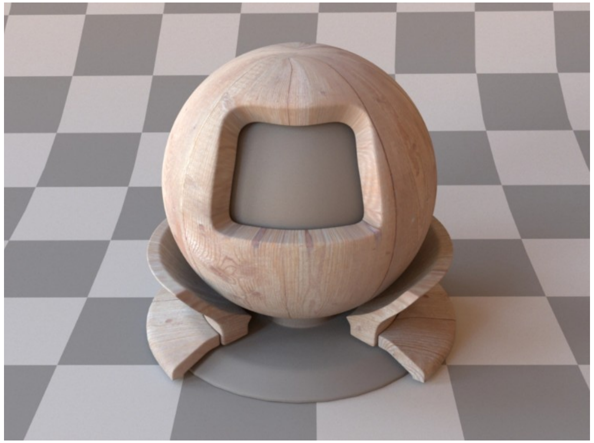
SSS – in this case, light is not exiting from the same point where it has entered, thus we need more than local information for calculating light behavior. Therefore, this is one of more complex effects that require advanced rendering methods or approximation methods.

Modeling material characteristics

- Physical considerations gave foundations to small scale light behavior:
  - Directional effects → scattering
  - Color
- Larger scale variations visible by eye are simulated by varying scattering models or their parameters over surface.
  - Texture



Modeling only scattering function (small scale) can represent different directional and color characteristics but surfaces are smooth.



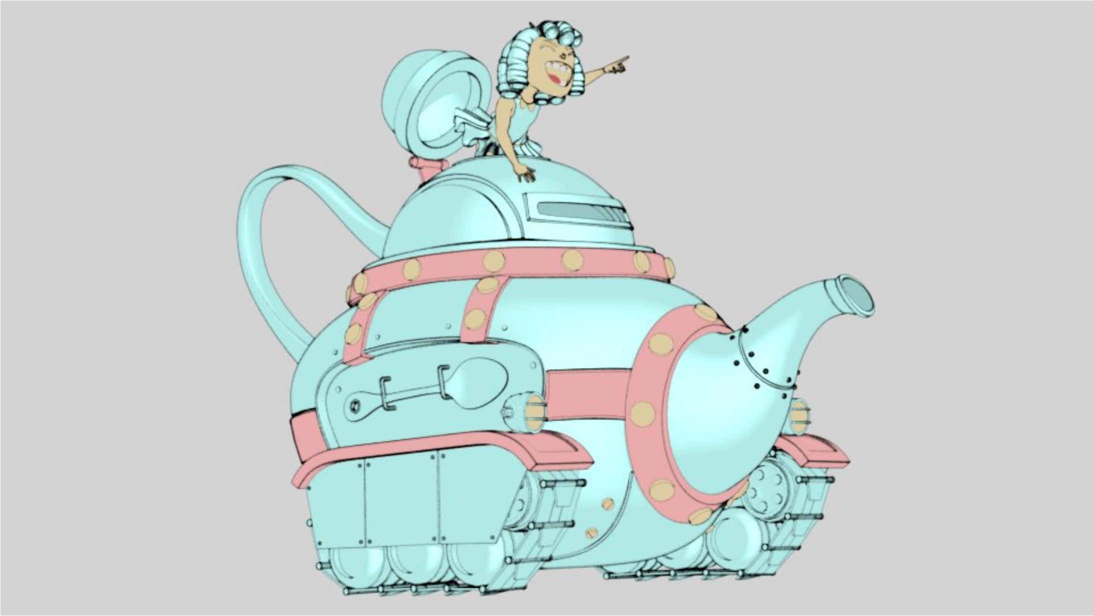
Varying scattering model parameters results in textured objects

Modeling appearance

Diversity of appearance models

- Appearance models, depending on application, can range from **photo-real** to **stylized**.

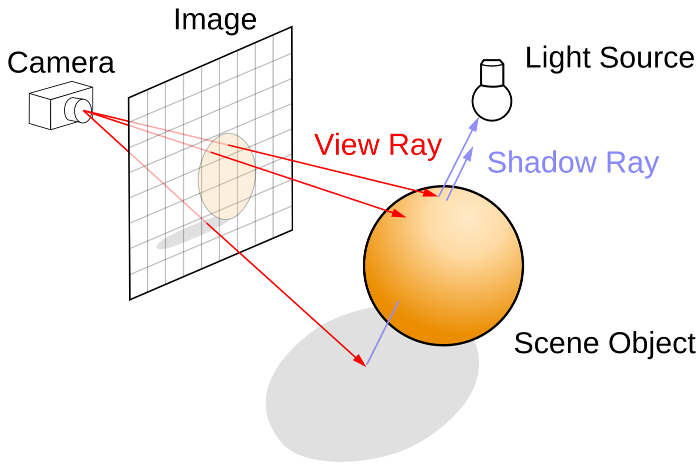
https://www.artstation.com/artwork/rANRe5



"Rolling Teapot" - Model by Brice Laville, concept by Tom Robinson, render by Esteban Tovagliari - RenderMan 'Rolling Teapot' Art Challenge: https://appleseedhq.net/gallery.html#https%3A%2F%2Fappleseedhq.net%2Fimg%2Frenders%2Frolling-teapot.jpg

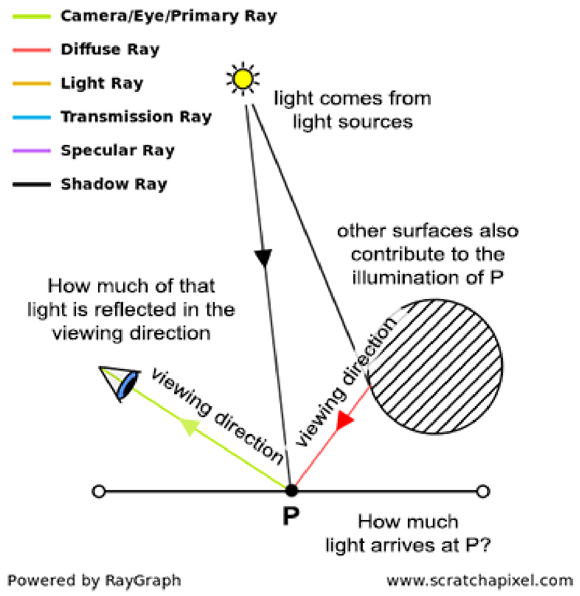
Material and rendering

- Rendering: computing **intensity** and **color** of light entering the camera along set of **viewing rays.**
- Viewing rays are intersecting objects in the scene → intensity and color of viewing ray comes from **closest intersected object surface**
  - Ignore **participating media** for now
- Compute intensity and color of the surface intersected by viewing ray → **shading**



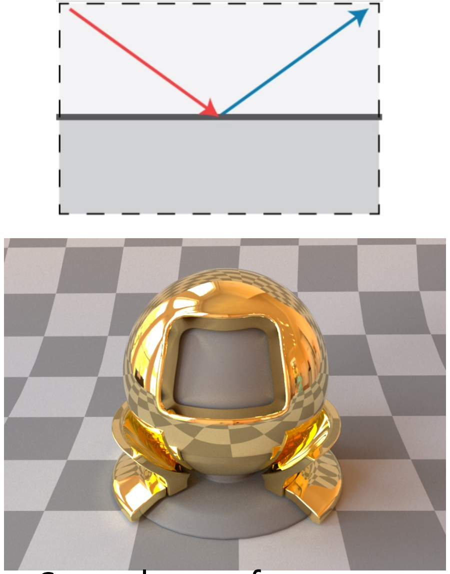
Materials and shading

- **Shading** uses **material** model and couples it with with **viewing** position, object **shape** and **light** information to calculate the object color in intersected point.
  - Light-object interaction in **shading point**
- Final color of surface is calculated by:
  - Summing all **incoming light** that falls onto surface – light transport – also uses material information
  - Calculating color and intensity of light reflected into camera using **material** description - shading

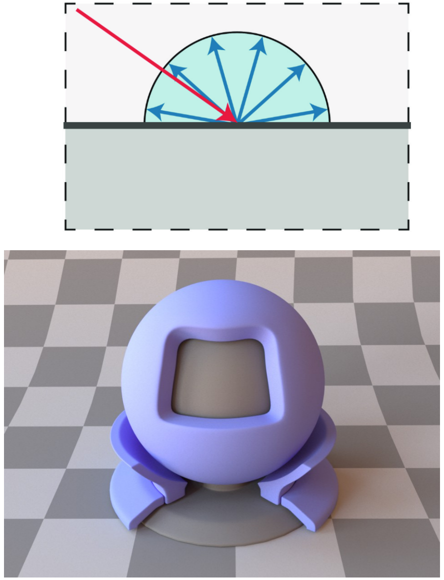


Material modeling

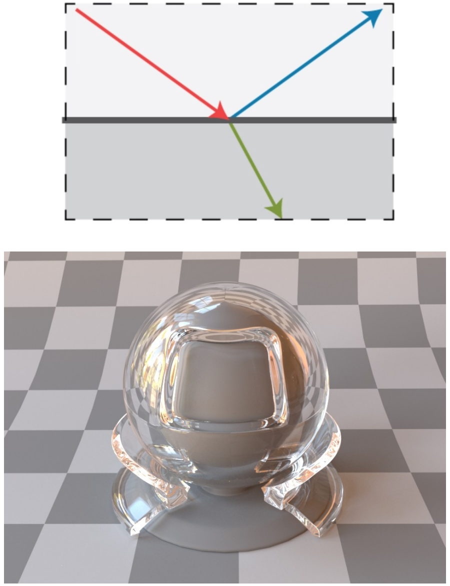
- Model interaction of light in shading point → **scattering model:**



Specular surface



Diffuse surface



Transmissive surface

Material modeling: texture

- Same description of material for each point of the 3D object, results in **homogeneous material**: smooth surface
- Scattering model can be parameterized to vary its properties over the surface.

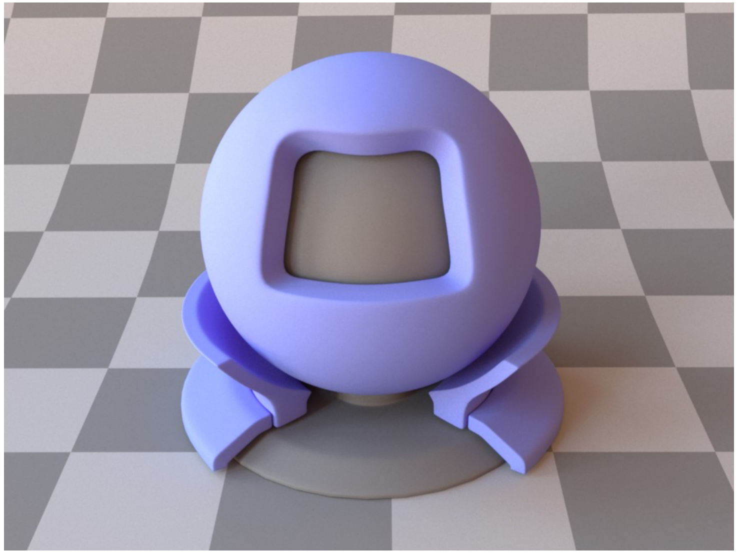
Homogeneous material: scattering model has the same parameters in all points of the surface.

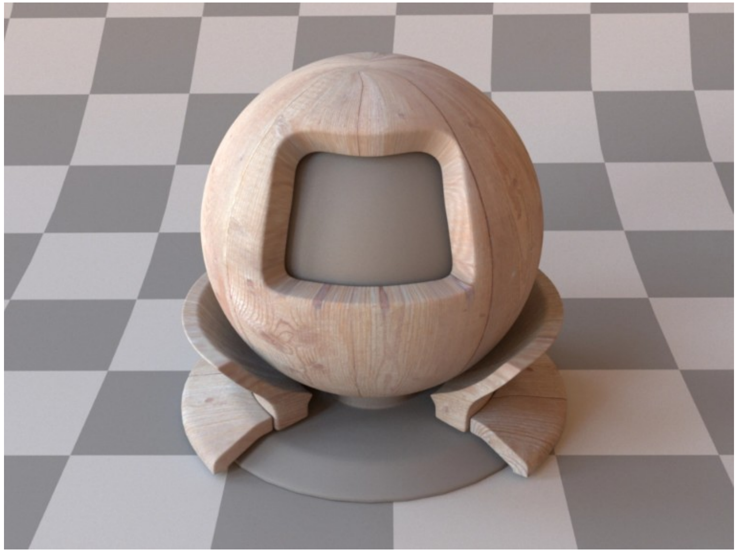
In CG texture is used to vary scattering parameters over surface

Models: https://polyhaven.com/a/throw_pillows_01

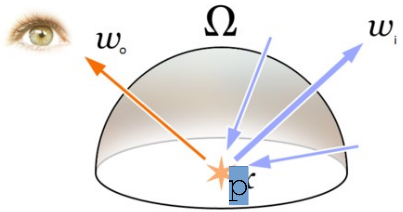
Elements of material model

- Material modeling is separated into:
  - **Scattering** → description of light-matter interaction at a point
  - **Texture** → variation of small-scale geometry and scattering properties across 3D object surface





Shading and light



- Shading "collects" all incoming light and shading point and multiplies it with scattering model.

Collect light over whole hemisphere above shading point, placed and normal in shading point

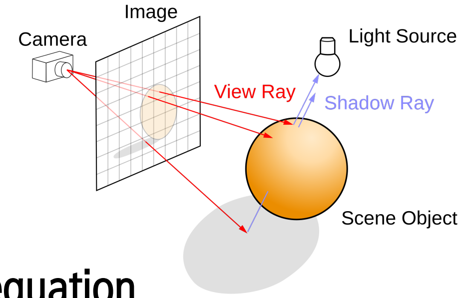
Incoming light at shading point

$$\boxed{L_o(p,v)} = \boxed{\int_{l\in\Omega}} \boxed{f(l,n,v)} \boxed{L(p,l)} \boxed{(n\cdot l)}^{+} \boxed{dl}$$

Color in shading point.

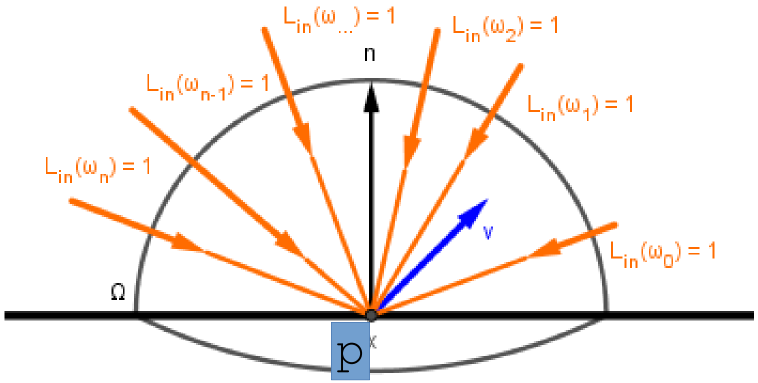
Scattering model

Light attenuation due to surface orientation



Reflectance equation.

Special case of **rendering equation.**



Shading and light

- Simplification: sum light from light sources

$$c_{shaded(p,v)} = \sum_{i=1}^{n} f(l_i, n, v)\, c_{light_i} (n \cdot l_i)^+$$

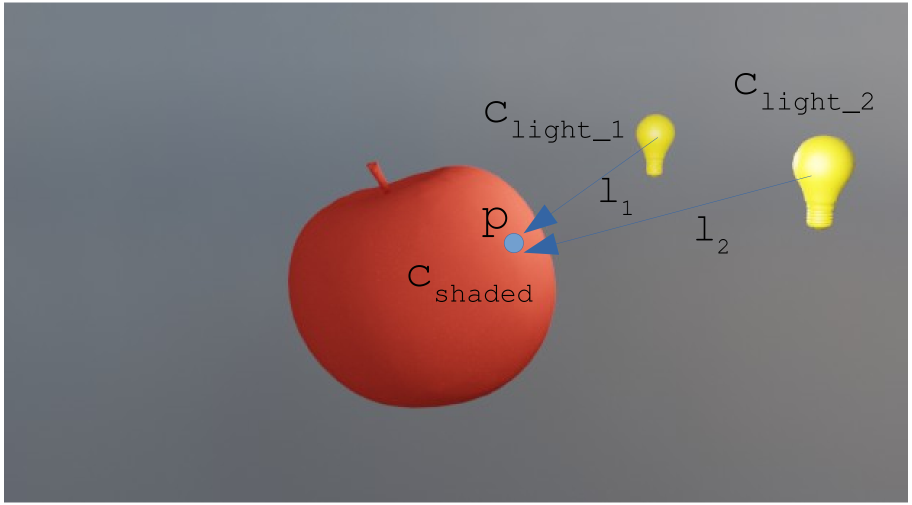
Color in shading point

Sum over all lights in 3D scene

Scattering model

Current light color

Attenuation



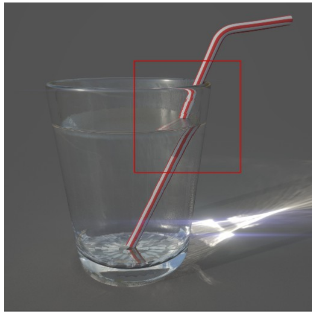
Scattering models

Light-matter interaction: summary

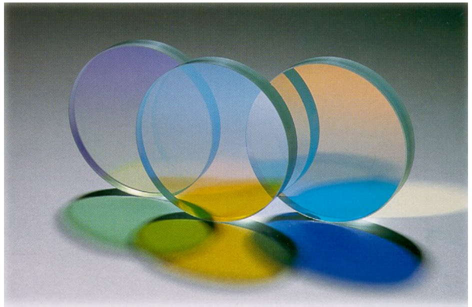
- Light traveling to/between objects can be scattered/absorbed → **volumetric scattering**
- Light falling on object surface – **surface scattering** - can:
  - **Reflect** (metals and dielectrics)
  - **Refract** (only dielectrics)
- Depending on material, refracted light can:
  - **Transmit** (transparent surfaces)
  - **Sub-surface scatter** (opaque and translucent object volume) → **volumetric scattering**

Participating media

Opaque dielectric: reflect + refract → SSS/diffuse



Transparent dielectric: reflect + refract → transmit

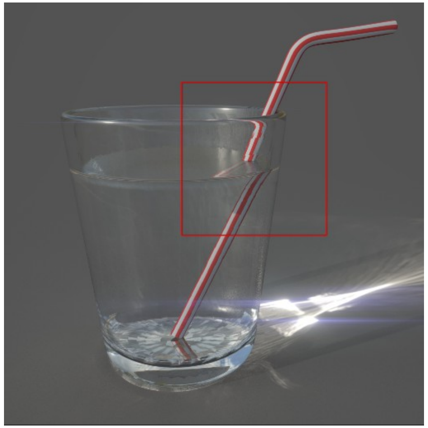


Translucent dielectric: reflect + refract → transmit

Metal: reflect

Surface scattering

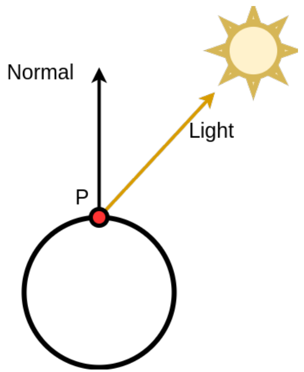
Reflection and transmission

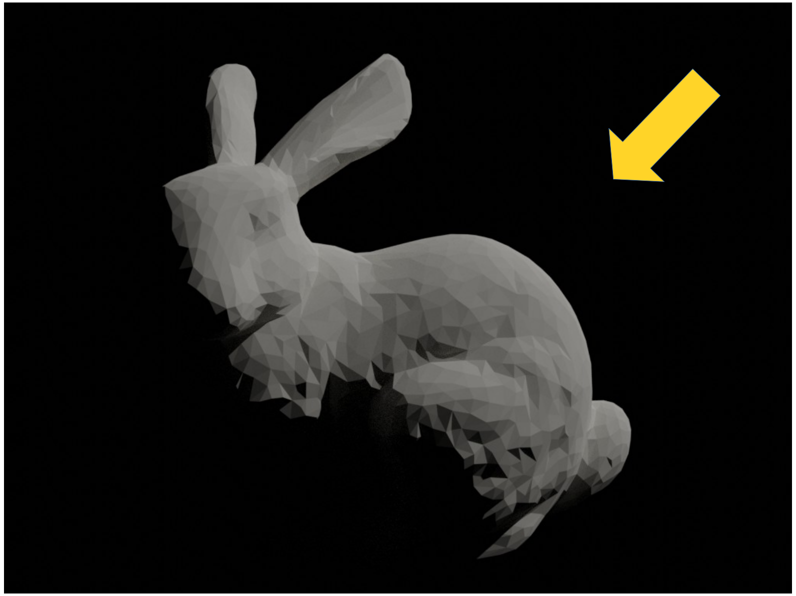


Scattering model

By now we know that amount of light falling on surface depends on shape and light:

- Surface normal in point is crucial for determining how much the surface is oriented towards light

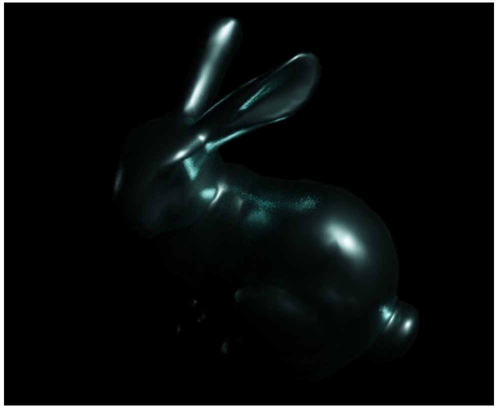




Scattering model

- Observing objects we can see that they have different appearance, although similar in shape
  - This particular look of objects is defined by how light scatters when it falls on surface point.
  - This behavior is defined by **surface scattering function** - surface response to light



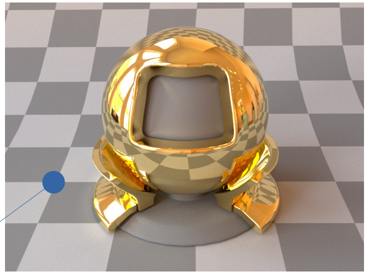
Surface scattering function

- Scattering function can be separated in reflection and transmission
  - Model describing reflection is called “bidirectional **reflectance** distribution function” – **BRDF**.
  - Model describing transmission is called “bidirectional **transmission** distribution function” – **BTDF**.
  - Model describing both reflectance and transmission is called “bidirectional **scattering** distribution function” – **BSDF**.

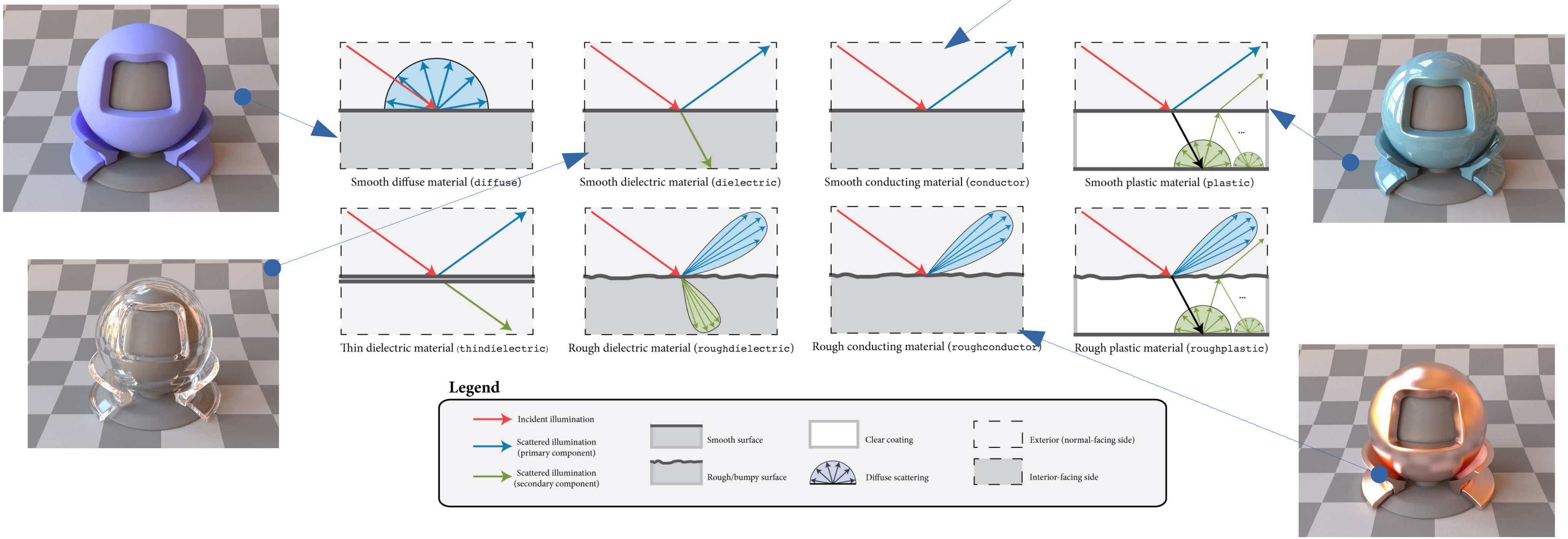
- **Reflective** – all light is scattered above surface
- **Transmissive** – all light is scattered below surface
  - Refractive – special case of transmissive

https://www.pbr-book.org/3ed-2018/Reflection_Models/Specular_Reflection_and_Transmission

Visualizing BRDF/BDTF

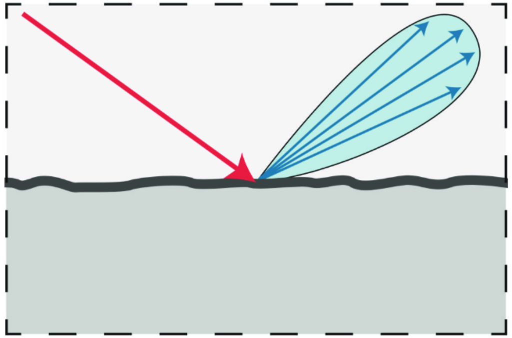


- The way surface reflects light can be visualized using **lobes**
  - Describes unique surface light reflection



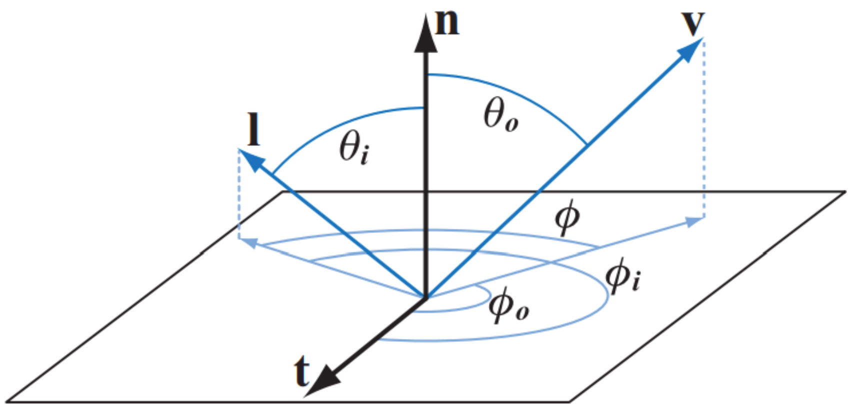
BRDF

- Describes reflectance phenomena: redirection of light hitting surface (shading point) back outward
- **Mathematical approximation of light interaction and microscopic structure of material:**
  - Surface reflection
  - Local sub-surface scattering (e.g., diffuse model)
- BRDF is evaluated at shading point



BRDF notation

- BRDF `f(v,n,l)` describes surface response which depends on:
  - Incoming – unit length vector - light direction (`l`)
  - Outgoing – unit length vector - view direction (`v`)
  - Shading point normal (`n`)
- Incoming (`l`) and outgoing (`v`) directions have 2 degrees of freedom, two angles relative to surface normal:
  - Elevation (θ)
  - Azimuth (Φ)
  - Dimensionality of BRDF: 4



BRDF in shading

$$c_{shaded(p,v)} = \sum_{i=1}^{n} f(l_i, n, v) c_{light_i} (n \cdot l_i)^+$$

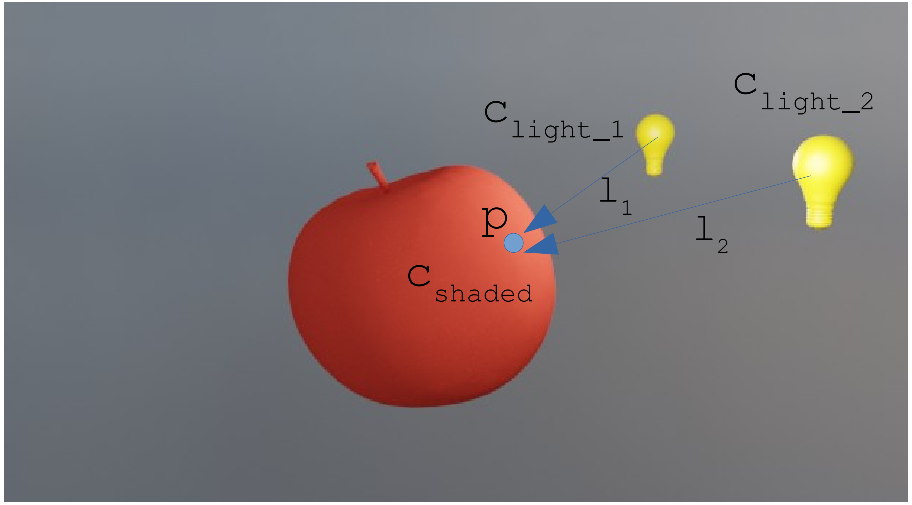
Color in shading point

Sum over all lights in 3D scene

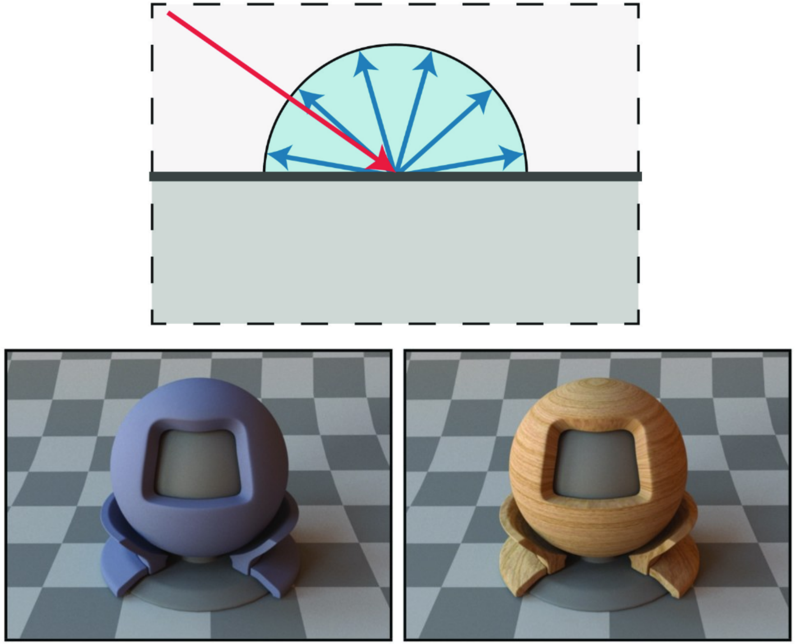
Scattering model → **BRDF**

Current light color

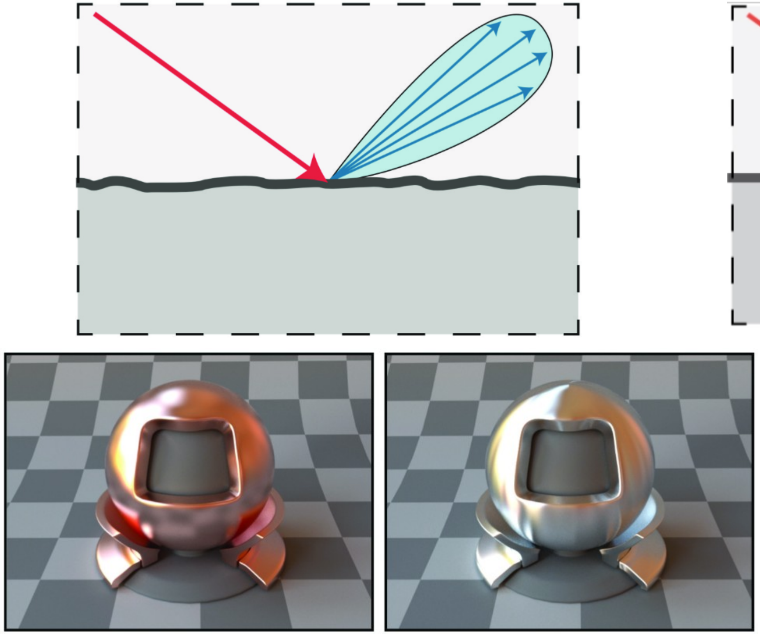
Attenuation



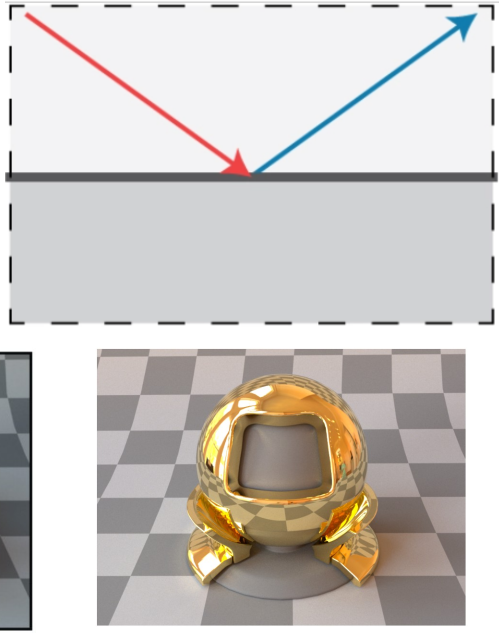
Common BRDFs



Diffuse, Diffuse textured.
Light is scattered in all possible directions. Independent of viewing direction. Equally bright from all directions.



Glossy: Copper, Aluminium.
Scattered light is concentrated around particular direction (lobe). Appears blurred. Dependent of viewing direction.



Specular (gold)
Light is scattered in single direction (mirror-reflection direction). Perfectly sharp. Dependent of viewing direction.

https://mitsuba2.readthedocs.io/en/latest/generated/plugins.html#bsdfs

BRDF reflection

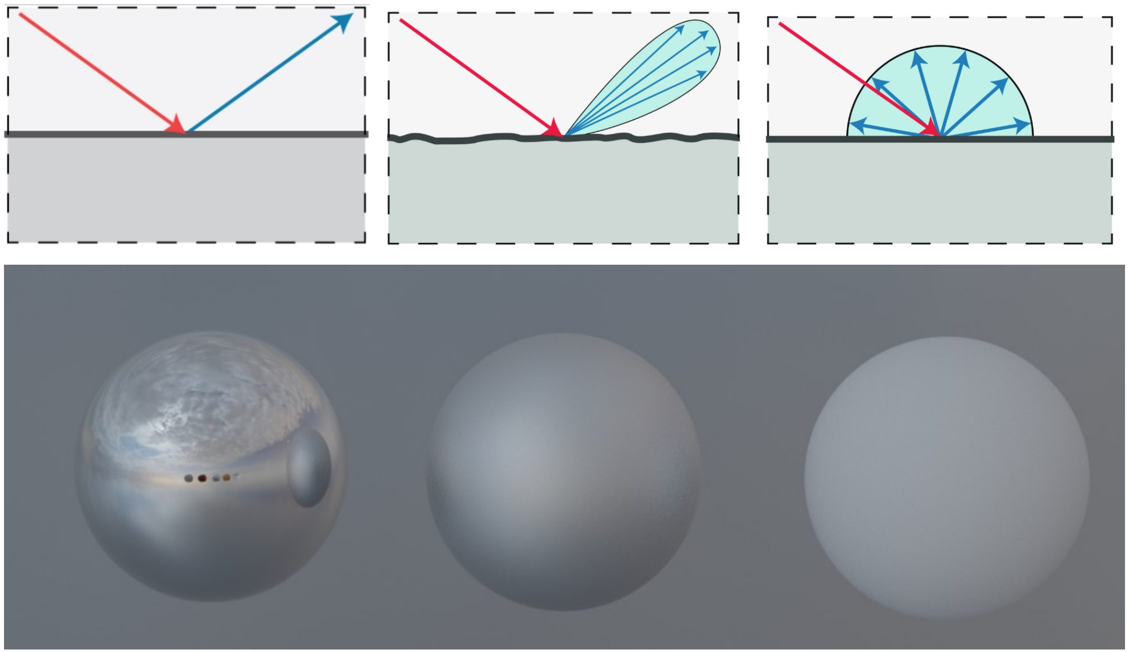
- Real materials have complex lobes
- While modeling, several lobes are combined with different weights and parameters
- Wide range of materials can be described with these three basic reflection types:
  - Specular
  - Diffuse
  - Glossy (rough specular)

Scattering models

- Modeling approaches:
  - **Empirical**
    - Simulate observed scattering phenomena
  - **Data-based**
    - Scattering is measured from real world and stored in tables
  - **Physical-based**
    - Based on physical interaction of light with matter

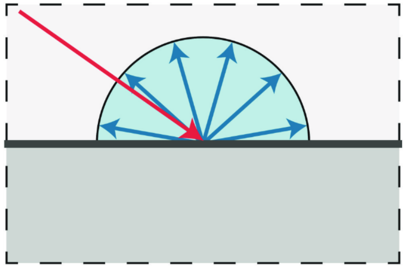
Empirical models

- Models based on **observation** of scattering phenomena rather than physical correctness
- **Phenomenological models**: describe the quantitative properties of real-world surfaces by mimicking them
- Easy to implement and use
- Types:
  - Diffuse → Lambertian
  - Specular → Mirror
  - Phong and Blinn-Phong

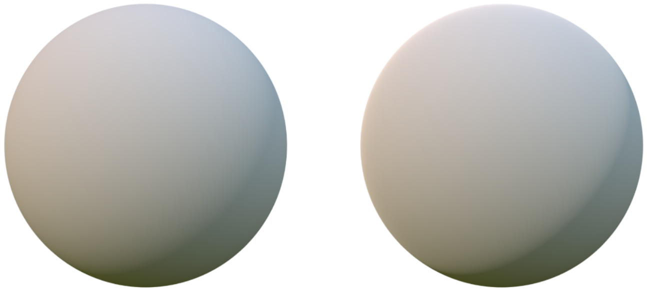


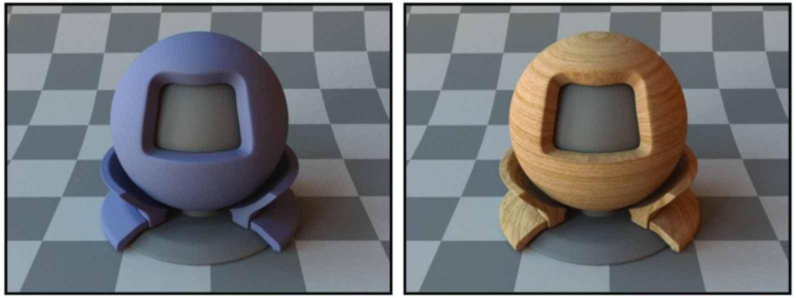
Lambertian (diffuse) model

$$f(v, l) = \frac{albedo}{\pi}$$



- Scattered amount of light in all directions is the same and linearly depend on incoming light
- Simplest BRDF model:
  - Doesn't depend view direction
  - Note that surface orientation will attenuate or increase incoming light making surface darker or brighter (**attenuation factor**)
  - **Constant reflectance value: diffuse color (albedo):** (R, G, B)
- Local model used to approximate sub-surface scattering
- The basis for more complex models





https://google.github.io/filament/Filament.html

Lamberitan in shading

$$c_{shaded(p,v)} = \sum_{i=1}^{n} \frac{albedo}{\pi} c_{light_i} (n \cdot l_i)^+$$

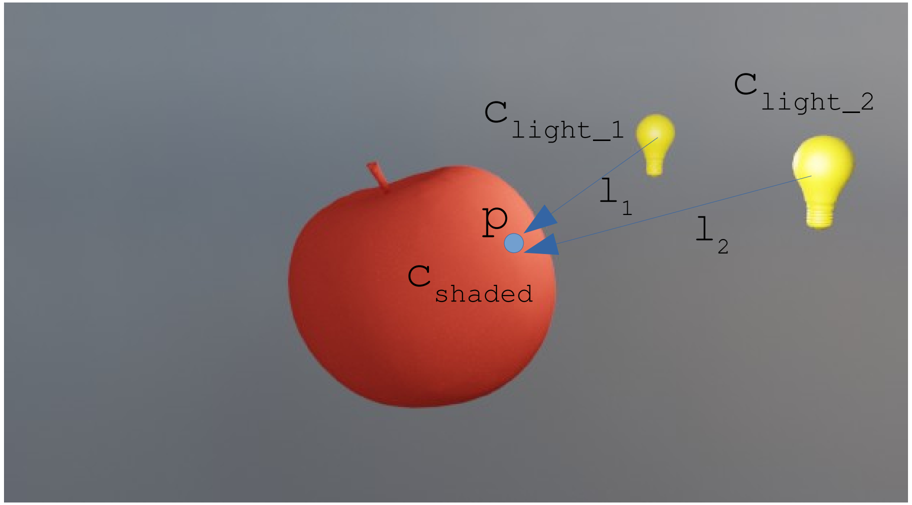
Color in shading point

Sum over all lights in 3D scene

Lambertian scattering model

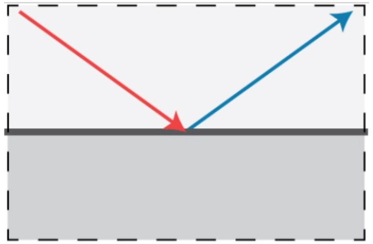
Current light color

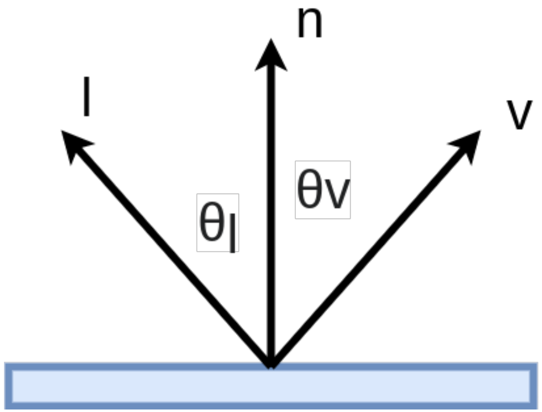
Attenuation

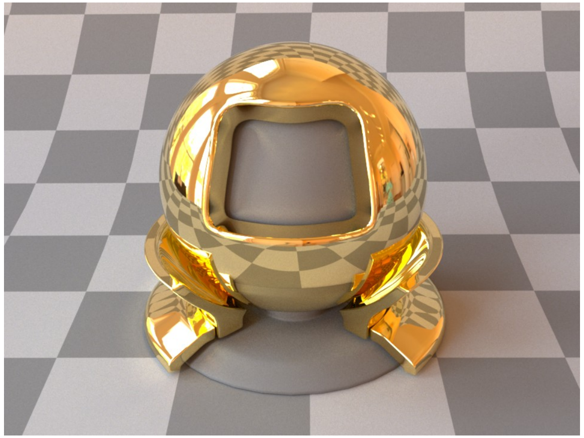


Specular (mirror) model

- Ideal reflection where incoming light is reflected completely in a single outgoing direction – **mirror reflectance direction**
- Depends on view and light directions
- Single parameter: **reflectivity** – a constant (R,G,B)







$$f(v,l) = \begin{cases} reflectivity \text{ if} & v = l - 2(l \cdot n)n \\ 0 & otherwise \end{cases}$$

Specular in shading

- Ideal reflection where incoming light is reflected completely in a single outgoing direction
  - **mirror reflectance direction**

$$f(v,l) = \begin{cases} reflectivity \text{ if } & v = l - 2(l \cdot n)n \\ 0 & otherwise \end{cases}$$

$$c_{shaded(p,v)} = \sum_{i=1}^{n} f(l_i, n, v) c_{light_i} (n \cdot l_i)^+$$

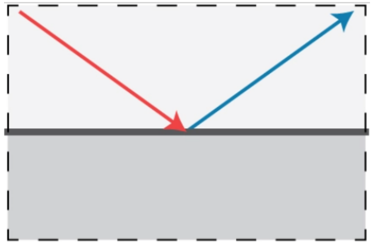
Color in shading point

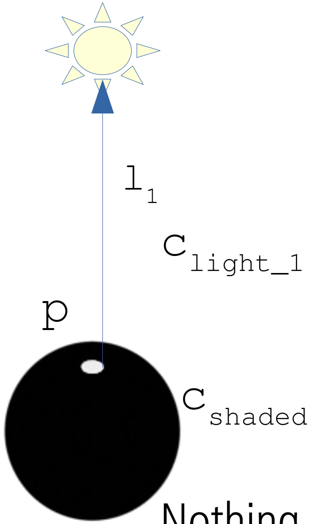
Sum over all lights in 3D scene

Specular scattering model

Current light color

Attenuation





Nothing to reflect!

Specular in shading

- Ideal reflection where incoming light is reflected completely in a single outgoing direction – **mirror reflectance direction**

$$c_{shaded(p,v)} = \sum_{i=1}^{n} f(l_i, n, v) c_{light_i} (n \cdot l_i)^+$$

$$f(v,l) = \begin{cases} reflectivity \text{ if} & v = l - 2(l \cdot n)n \\ 0 & otherwise \end{cases}$$

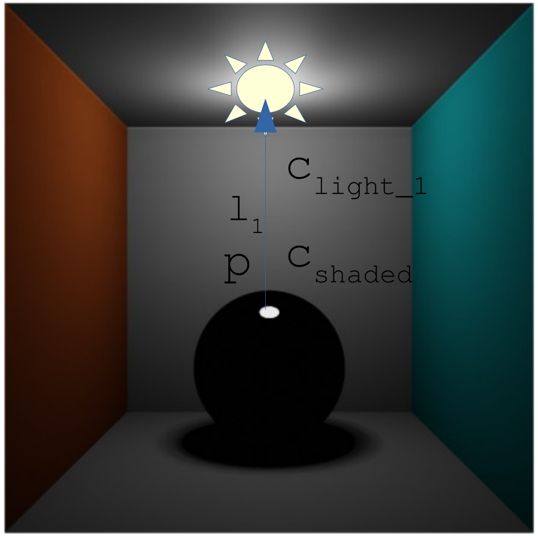
Color in shading point

Sum over all lights in 3D scene

Specular scattering model

Current light color

Attenuation



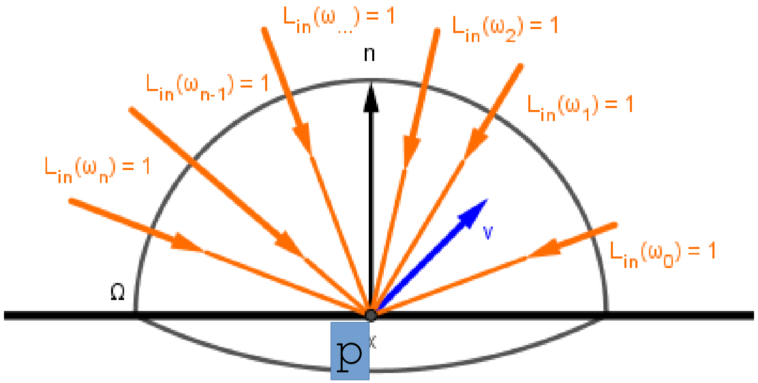
Drawback of **direct illumination**

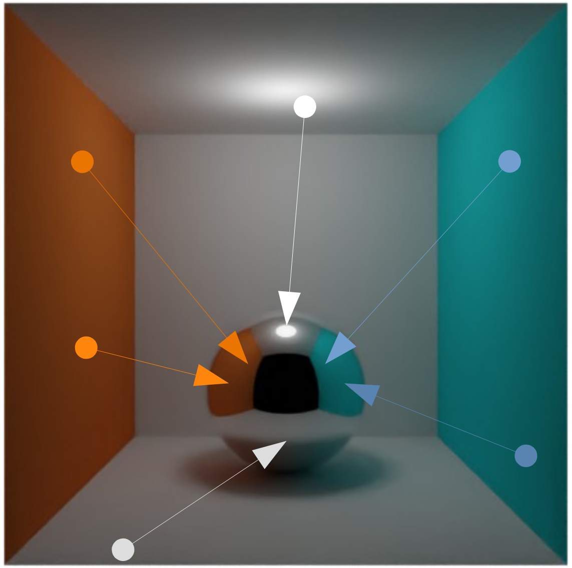
Specular in shading

- Ideal reflection where incoming light is reflected completely in a single outgoing direction – **mirror reflectance direction**
  - **Light has to be gathered from all surfaces in 3D scene → reflectance equation**

$$L_o(p, v) = \int_{l \in \Omega} f(l, n, v) L(p, l)(n \cdot l)^+ dl$$

Global illumination: direct and indirect illumination

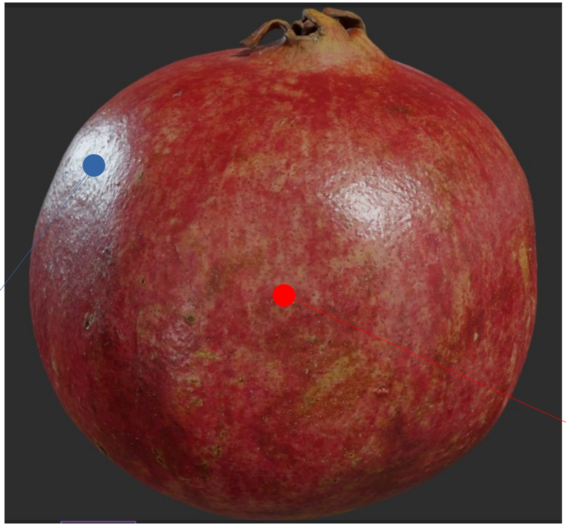




Specular and diffuse model

- Specular reflectivity parameter is color for conductors
  - Conductors have colored specular reflection due to spectral selectivity

Conductors: colored specular highlight

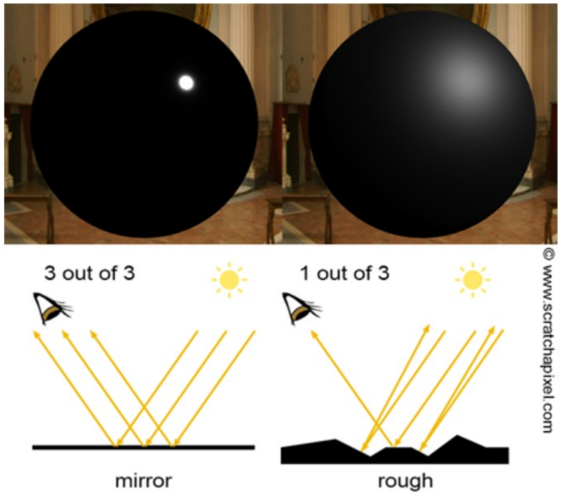


Specular highlight.
Dielectrics: specular highlight the same as light source color

Diffuse

Phong model

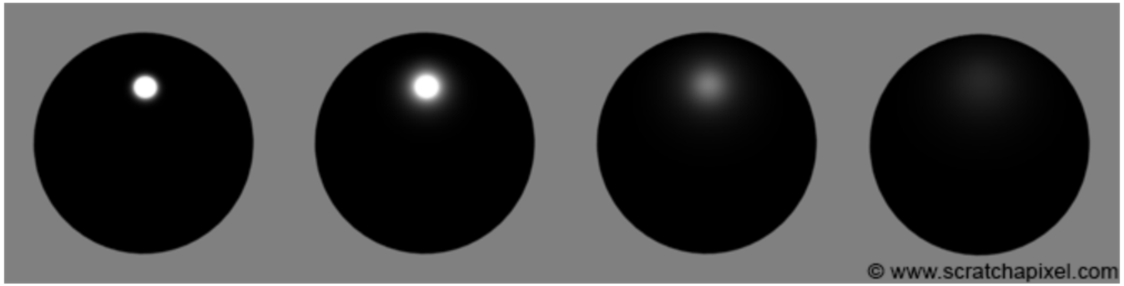
- Idea: wide range of materials can be described with **diffuse** and **specular** component:
  - `surface_color = diffuse() * K`$_d$ `+ specular() * K`$_s$
- Some surfaces exhibit **glossy** highlights: reflection of light source (or another object) on object surface
  - Compared to perfect specular mirror reflection, glossy is blurred and broken up
  - This kind of rough surface acts like “broken mirror” and it is modeled as collection of small mirrors → **micro-facets**

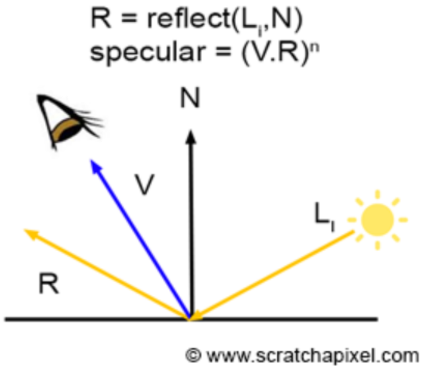


- Only fraction of light rays are reflected in eye direction by rough surface causing dimmer/blurred light reflection
- Brightness of glossy reflection decreases as angle between view and ideal reflection increases – number of microfacets reflecting in eye decreses

Phong model: glossy observations

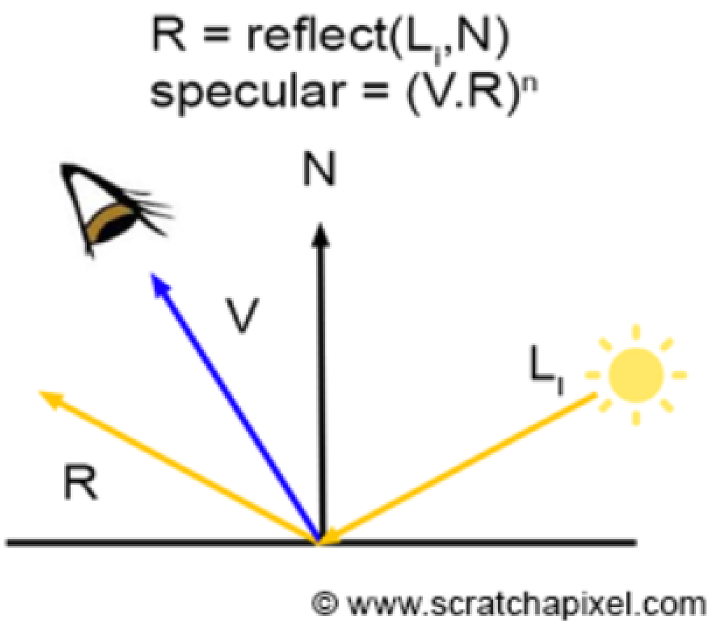
- Brightness of specular reflection decreases as the distance increases between:
  - point on the surface of the object and
  - point where the reflection of the light source would be formed if the surface was a perfect mirror
- Reflection brightness decreases as light source reflection spreads across larger area
- Highlight brightness decreases as distance of object points from original reflected light position increases





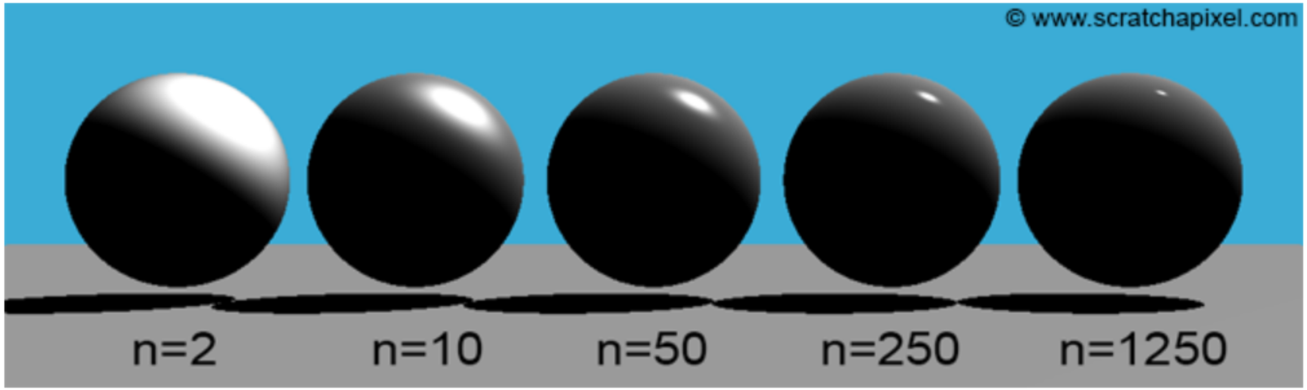
Phong model

- Phong observation: glossy surface can be simulated by:
  - Computing ideal specular reflection `(R)` of incident light ray `(L)`
  - Computing dot product between reflected ray and view direction
  - Raising dot product to power of `n`



$$\text{Specular} \approx (V \cdot R)^n.$$

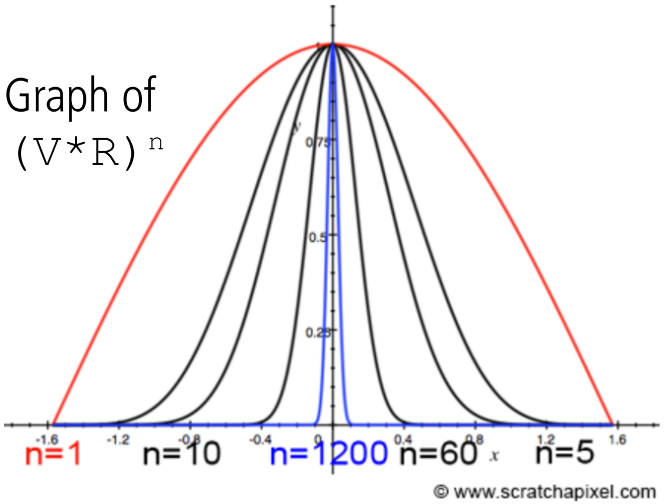
$$R = 2(N \cdot L)N - L.$$



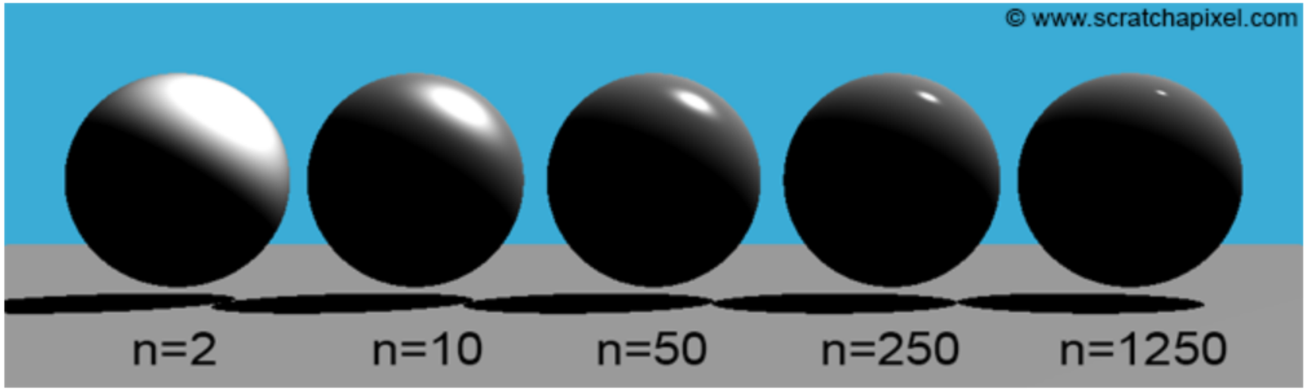
Phong model

- Empirical model: parameters doesn't have physical meaning
  - Parameters are tweaked by artist/user until desired appearance is achieved
- Physically based models build on this concept and provide physically-based parameters

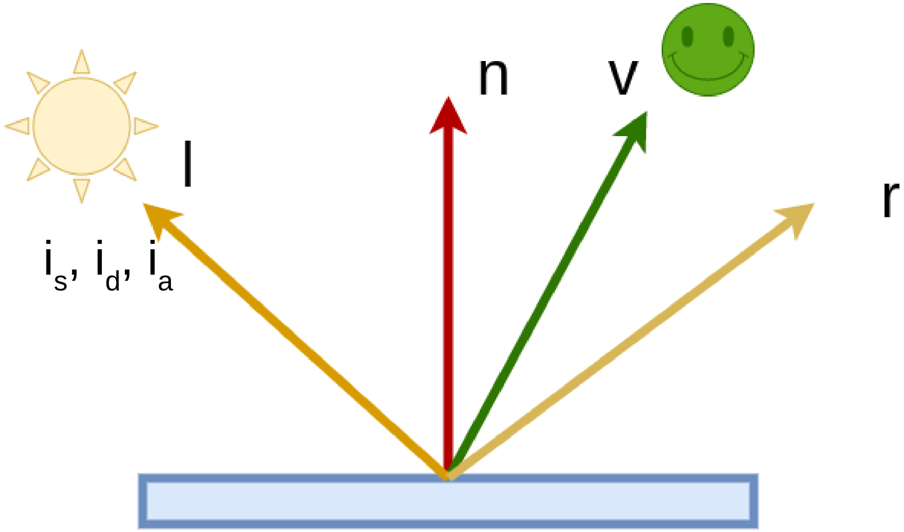
Graph of $(V*R)^n$



$$\text{Specular} \approx (V \cdot R)^n.$$

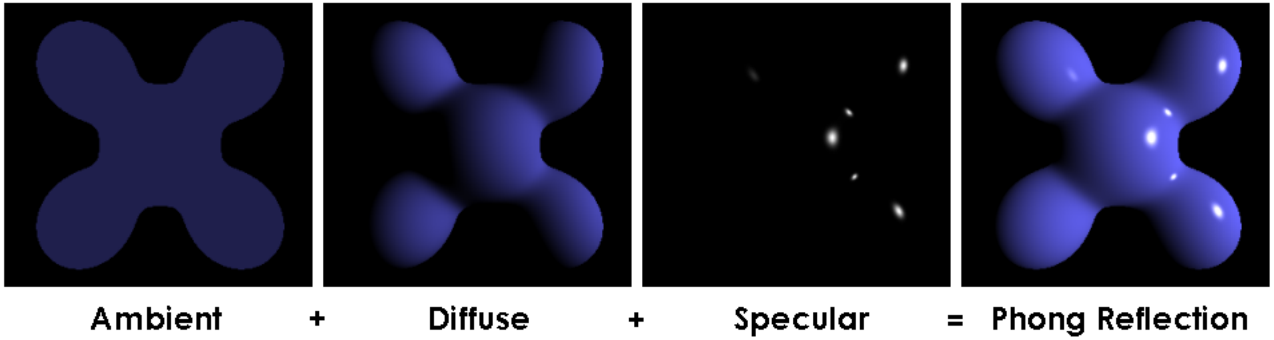


Phong illumination model



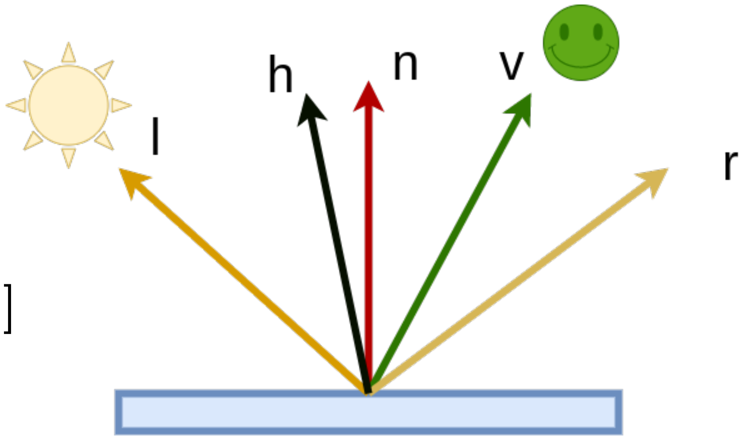
- Phong illumination model describes complete shading
- A phenomenological model using:
  - **Diffuse** reflection: rough surfaces; large highlights
    - Parameter: k_d – ratio of reflection of the diffuse term of incoming light
  - **Specular** reflection: shiny surfaces; small highlights
    - Parameter: k_s – ratio of reflection of the specular term of incoming light
    - Additional parameter: alpha – shininess constant: larger for smooth and mirror-like surfaces and small specular highlights
  - **Ambient** term: small amount of light that comes from around the scene
    - Parameter: k_a – ratio of reflection of the ambient term for all points on the surface

$$L_o = k_a i_a + \sum_j^n (k_d (l_j \cdot n) i_{j,d} + k_s (r_j \cdot v)^\alpha i_{j,s})$$



Blinn-Phong

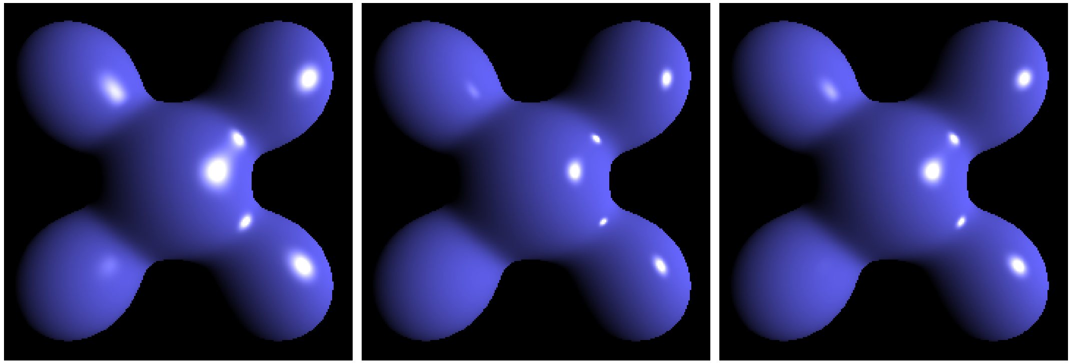
- Modification to Phong model, a BRDF
- Introducing **half vector** between light and view vectors
- Parameters: Lambertian (k_L) and glossy (k_G), range: [0,1]
- Energy conserving if $k_L + k_G <= 1$



$$f(v,l) = \frac{k_L}{\pi} + k_G \frac{8+s}{8\pi} z^2$$

$$z = max(0, h \cdot n)$$

$$h = \frac{v+l}{2}$$



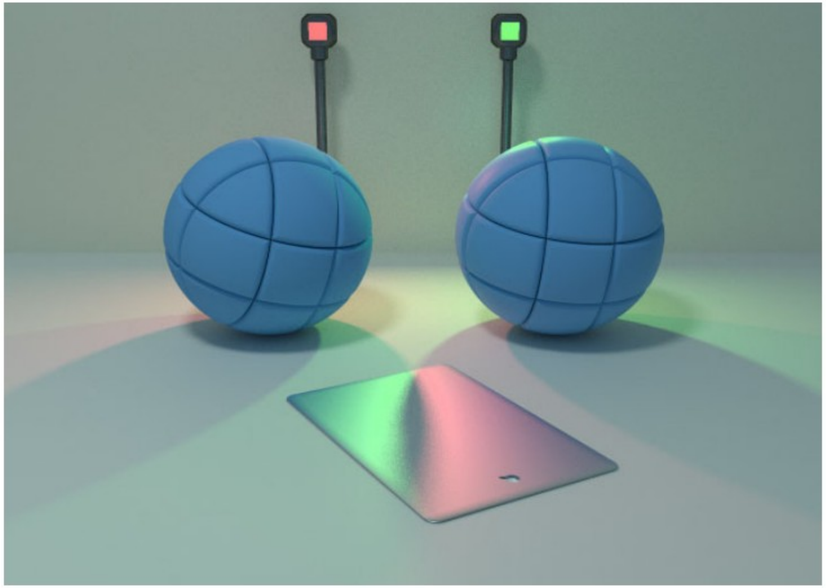
Blinn–Phong **Phong** **Blinn–Phong (higher exponent)**

https://en.wikipedia.org/wiki/Blinn%E2%80%93Phong_reflection_model

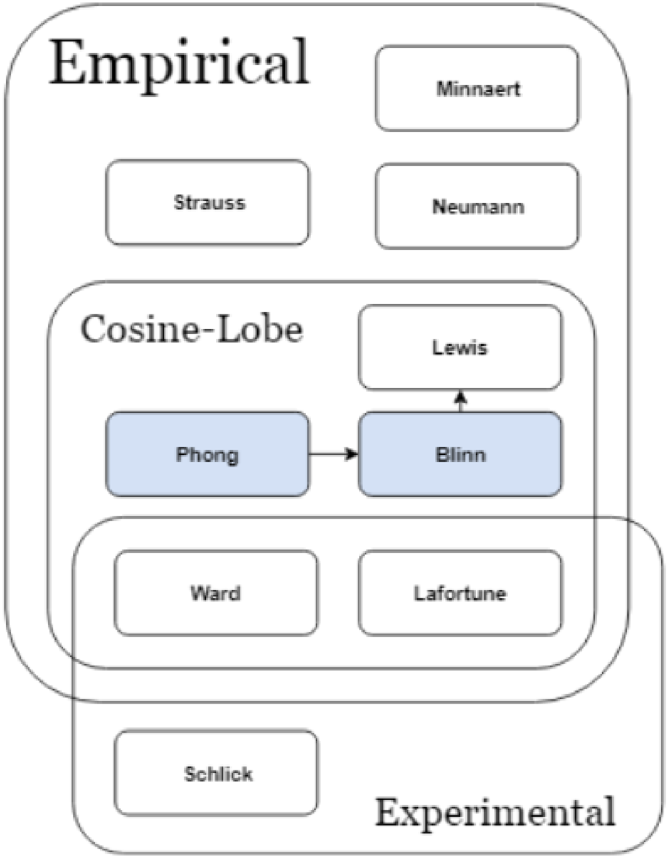
Example in OpenGL:
https://learnopengl.com/Advanced-Lighting/Advanced-Lighting

Other empirical models

- Lafortune model
  - Generalization of Phong's model
  - Richer appearance with multiple lobes



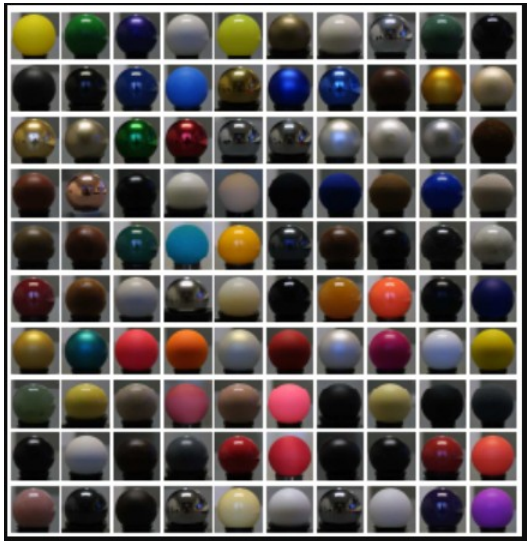
Lafortune model. http://people.csail.mit.edu/addy/research/ngan05_brdf_eval_ppt.pdf



https://cglearn.eu/pub/advanced-computer-graphics/physically-based-shading

Data-based models

- BRDF measurement → **gonioreflectomenter**
  - Measurements are stored in table on which lookup with view and light direction is performed
- BRDF measurement of real-world can be used for:
  - Evaluation of phenomenological and physically based models
  - Modeling of material
- Different real-world materials have been measured:
  - Isotropic BRDFs
  - Anisotropic BRDFs
  - Texture characteristics
  - Sub-surface scattering
- Problems:
  - Costly for rendering
  - Memory

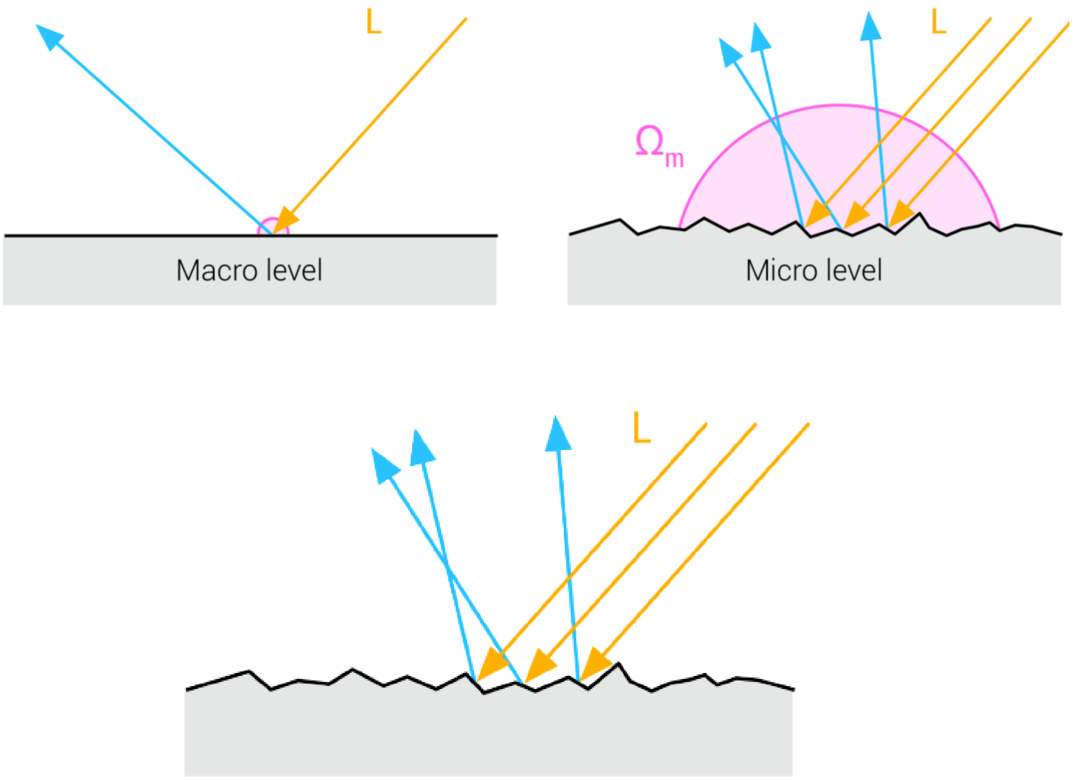


Isotropic materials: https://www.merl.com/brdf/

Anisotropic materials: Brushed aluminium, Yellow satin, Purple satin, Red velvet
http://people.csail.mit.edu/addy/research/brdf/

Physically-based models

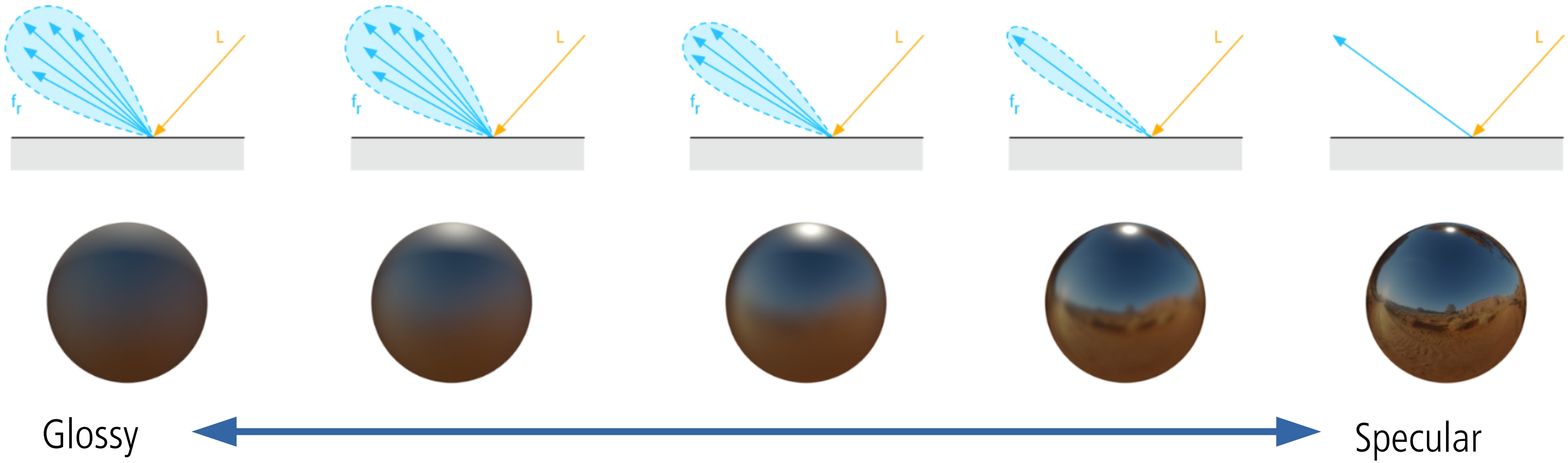
- Real-world surfaces have geometrical detail on multiple and thus very small scales
  - Although we can not see those small-scale details individually, we can see their aggregate response and they define how surface actually appears
- **Small-scale surface irregularities** (smaller than a pixel) can not be modeled explicitly, therefore this is modeled using BRDF
  - **Geometric optics** assumption: these irregularities are much smaller than light wavelength (they have not effect of appearance) or much larger than light wavelength (they cause light redirection). Wave optics describes phenomena in between.



https://google.github.io/filament/Filament.html

Physically-based models

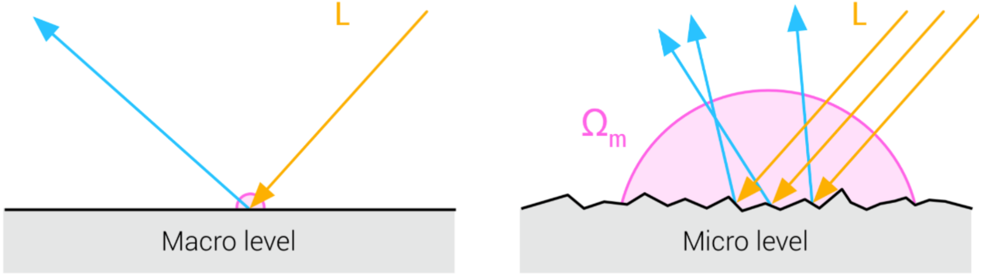
- Small scale geometry is invisible to the eye directly, but its cumulative response is visible.
- Transition from glossy to specular is determined by small scale irregularities

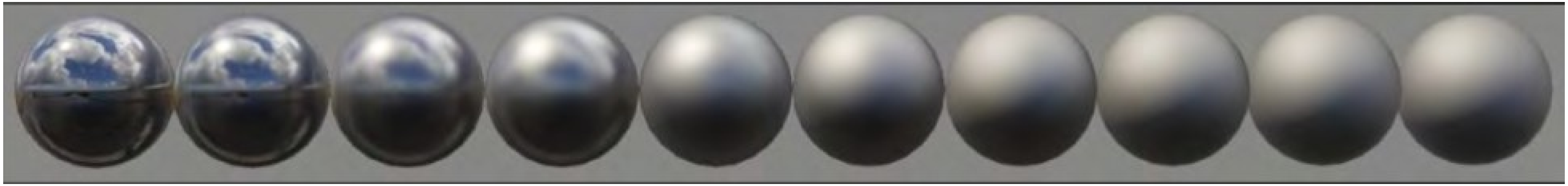


https://google.github.io/filament/Filament.html

Physically-based models

- Small scale irregularities are represented as **facets** with **microsurface normals***.
  - **Micro-facet theory**: statistical distribution of microfacets which have strong peak at macrosurface normal. Spread of this distribution is called **roughness**
  - Higher roughness means more blurred surface since micronormals will be more spread (**glossy-diffuse**). Small roughness gives mirror-like surface (**specular**).



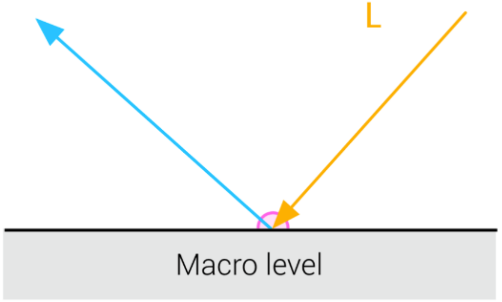


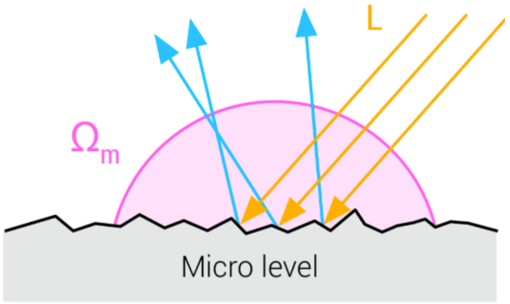
* Note how important normal information in computer graphics is. Different scales of geometry are always described with normal information.

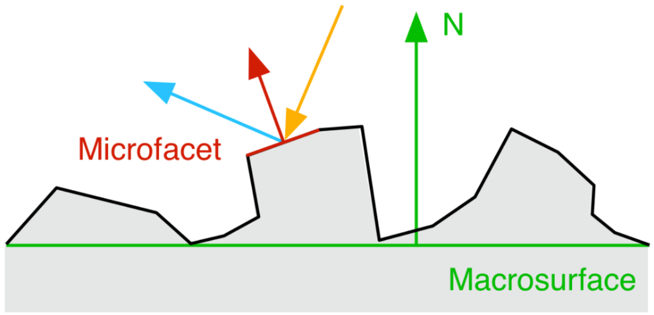
https://docs.quixel.com/mixer/1/en/topic/pbr-physically-based-rendering

Physically-based models

- Assumption of microfacets which are **optically-flat**
  - Diffuse
  - Specular
- Geometric optics can approximate how light interacts with each microfacet using **IOR, Fresnel equations and Snell's law**





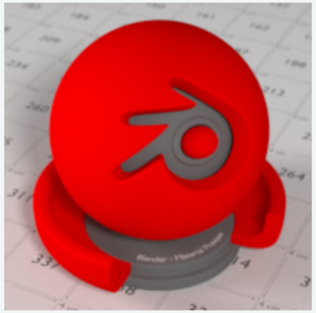


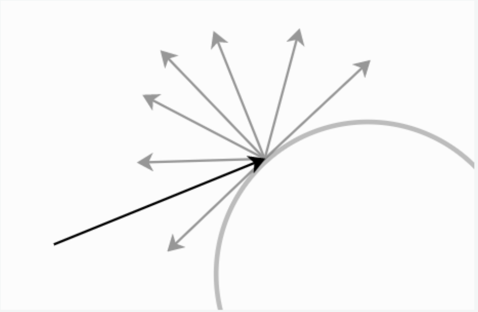
Physically-based models

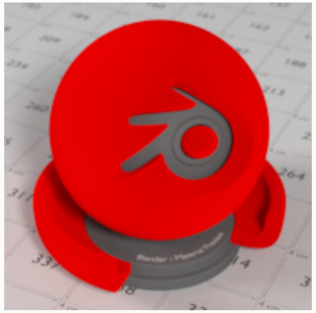
- Microfacet-based BRDFs are state of the art reflectance models using in professional modeling and rendering software.
- Fundamental models are:
  - **Torrance–Sparrow** for glossy surfaces
  - **Oren-Nayar** for diffuse surfaces

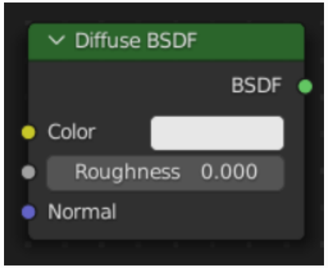
https://www.pbr-book.org/3ed-2018/Reflection_Models/Microfacet_Models

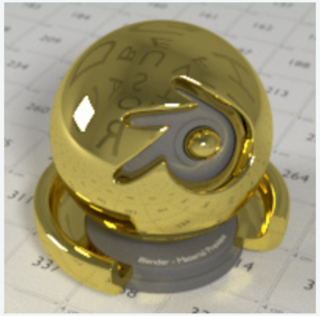
Blender is our friend

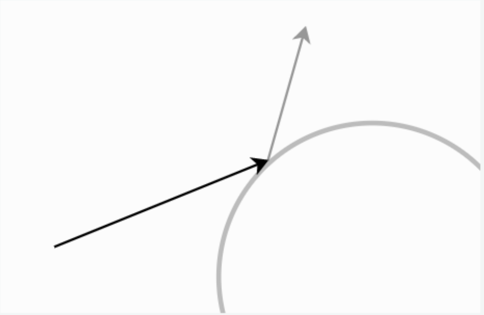

Lambertian reflection.


Diffuse shader behavior.

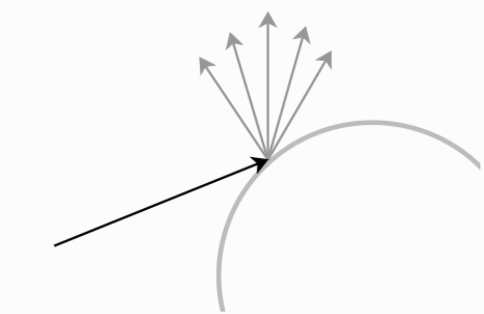

Oren-Nayar reflection.




Sharp Glossy example.


Sharp Glossy behavior.


Rough Glossy example.


Rough Glossy behavior.

- https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/diffuse.html
- https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/glossy.html
- https://docs.blender.org/manual/fr/2.79/render/blender_render/materials/properties/diffuse_shaders.html?highlight=diffuse%20shaders

Physically-based models

- Physically based rendering imposes two laws:
  - **Helmholz reciprocity:** `f(l,v) = f(v,l)` – input and output can be switched and the function value will stay the same
  - **Conservation of energy*:** outgoing energy can not be greater than incoming energy. BRDF which significantly violates this property leads to too bright and thus not realistic surfaces.
    - Note that BRDF can have arbitrary large values in certain directions if the distribution it describes is highly non-uniform. An example are highly reflective surfaces with highlights.

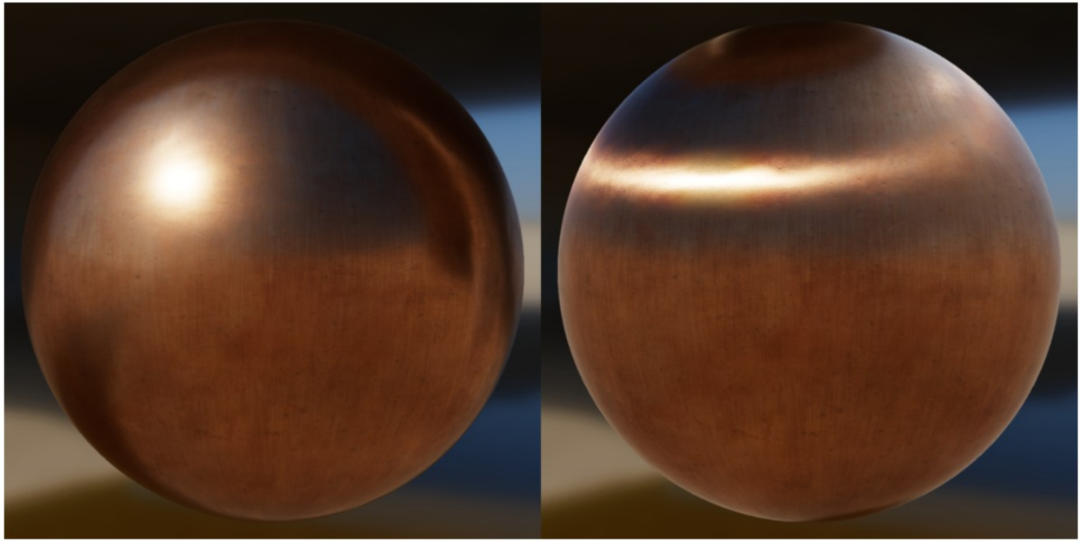
White furnace test: white sphere is illuminated with white light from all directions. If energy conserving is satisfied, the sphere will disappear in white background

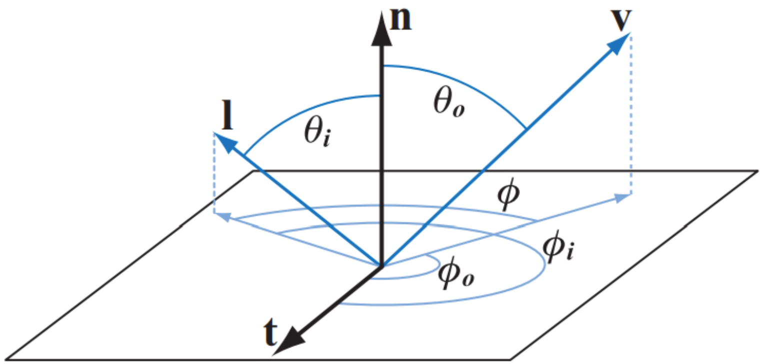
https://boksajak.github.io/files/CrashCourseBRDF.pdf

* Energy conservation is measured with directional-hemispherical reflectance R(l). It measures amount of light coming from given direction that is reflected into any outgoing direction in the hemisphere around normal – it measures energy loss for a given incoming direction. If BRDF is reciprocal then hemispherical-directional reflectance can be calculated as well giving the same value. Term for both reflectances in Directional albedo. The value must be in [0,1] to satisfy energy conservation: 0 is completely absorbed, 1 is completely reflected. Note that this restriction doesn't apply to BRDF since it can have arbitrarily large values in certain direction (e.g., highlight direction).

Isotropic and anisotropic BRDF

- **Isotropic BRDF**: rotating light and view directions around the surface normal does not affect the BRDF.
  - Incoming and outgoing directions have the same relative angles between them: such BRDF can be parameterized with three angles.
- **Anisoptropic BRDF**: reflection behavior changes when light and view vectors are rotated around normal

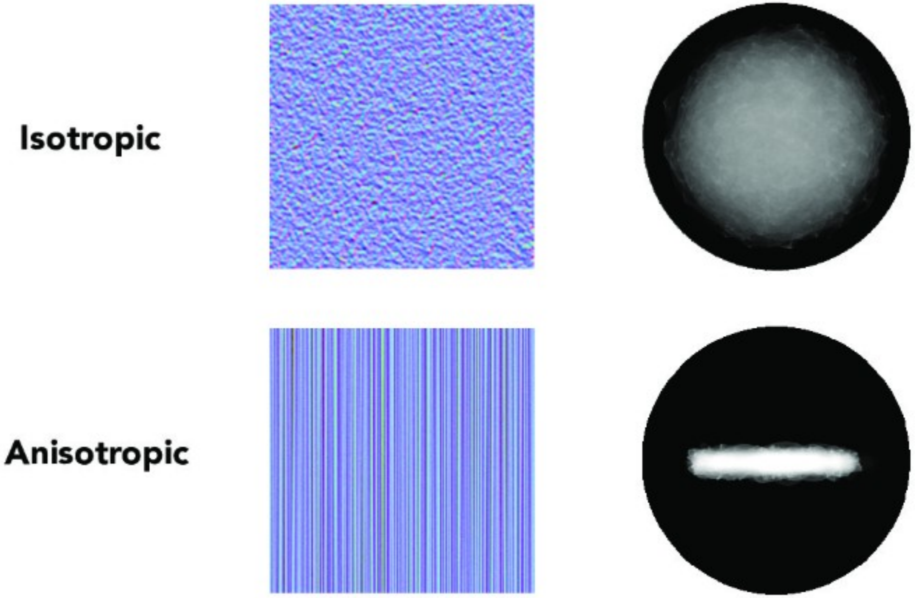




https://google.github.io/filament/Filament.html

Isotropic and anisotropic BRDF

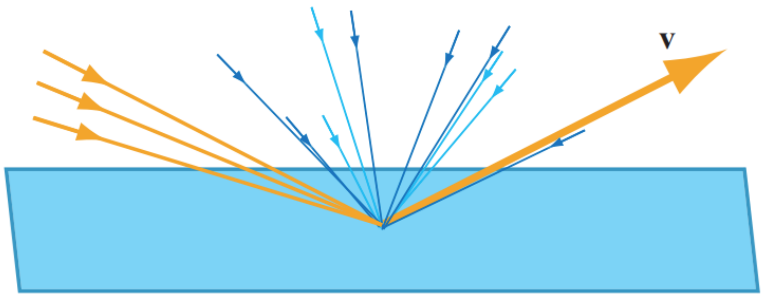
- **Anisoptropic behavior** is present due to underlying surface sturcture which is directional



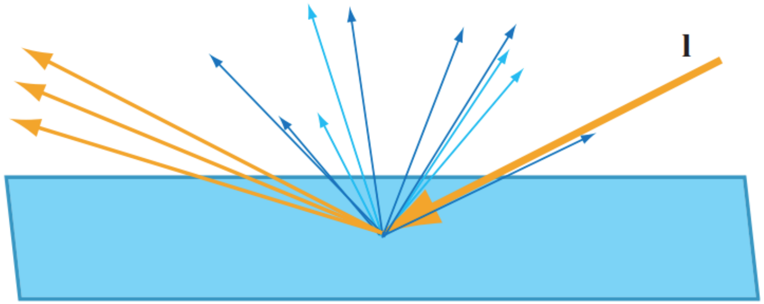
https://cs184.eecs.berkeley.edu/sp22/lecture/14-29/material-modeling

Note: bidirectionality of BRDF

- "**bidirectional**" in BRDF means that given given incoming and outgoing direction, we can compute **amount of reflected light** in outgoing direction.
- Further, bidirectionality can be used for:
  - Given outgoing (view) direction, it specifies the relative contributions of incoming light
  - Given incoming light direction, it specifies distribution of outgoing light



Contributions of incoming light given view direction



Distribution of outgoing light directions given incoming light direction

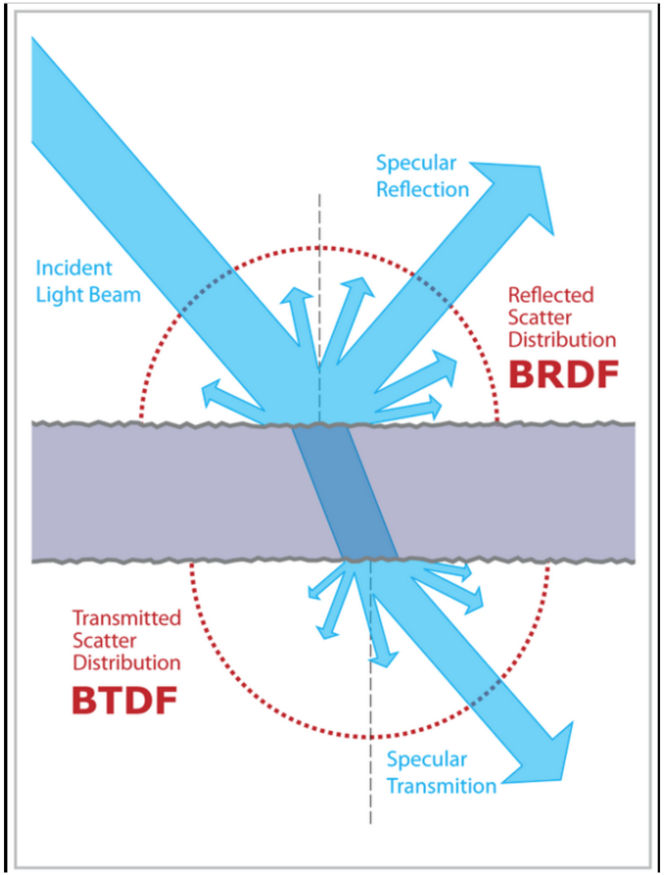
https://blog.selfshadow.com/publications/s2013-shading-course/hoffman/s2013_pbs_physics_math_notes.pdf

BDTF

- Scattering function can be separated in reflection and transmission
  - Model describing reflection is called “bidirectional **reflectance** distribution function” – **BRDF**.
  - Model describing transmission is called “bidirectional **transmission** distribution function” – **BTDF**.

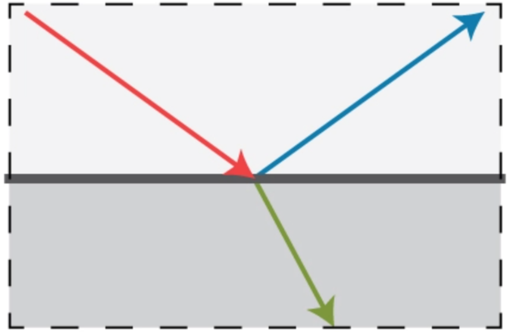
Reflection

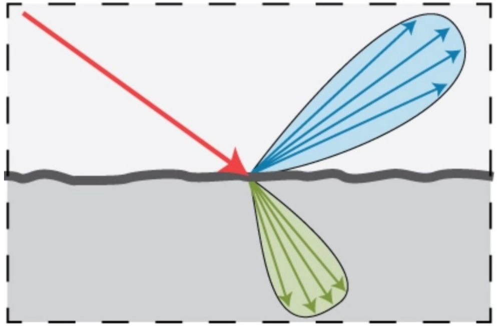
Transmission

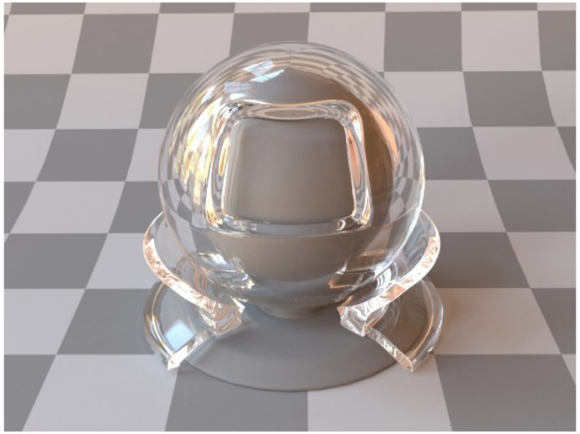


https://en.wikipedia.org/wiki/Bidirectional_scattering_distribution_function

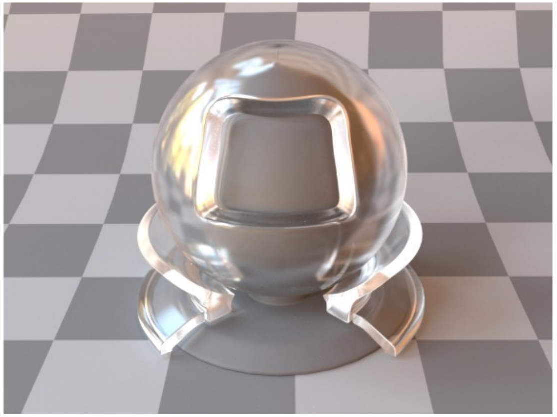
Common BDTF types







Specular transmission
Smooth dielectic



Glossy transmission
Rough dielectic

https://mitsuba2.readthedocs.io/en/latest/generated/plugins.html#bsdfs

BTDF: specular transmission

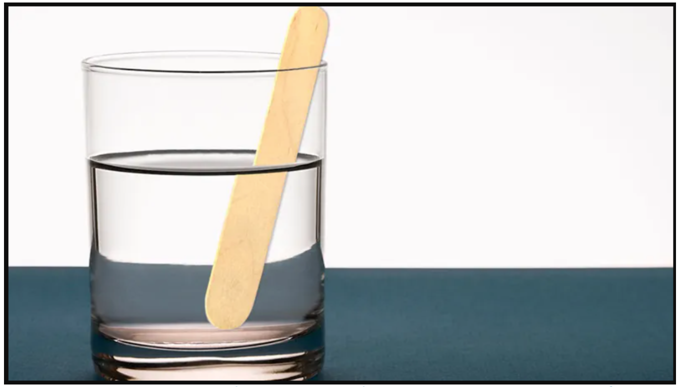
- Light passing from one transparent to another transparent medium with different IOR
  - Refraction – light changes direction – illusion of disproportional/broken object
- New - transmission - direction depends on:
  - IOR η
  - Incident direction
- Snell's law: refracted angle

$$\frac{\sin\theta_1}{\sin\theta_2} = \frac{\eta_2}{\eta_1}.$$

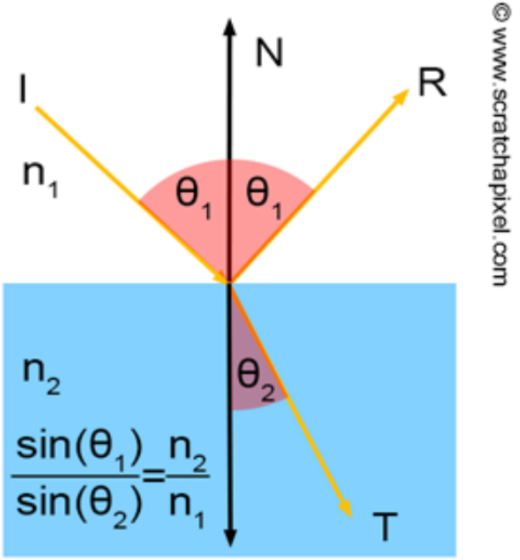
- Refraction/transmission direction:

$$T = \eta I + (\eta c_1 - c_2)N.$$

$$\eta = \frac{\eta_1}{\eta_2},$$
$$c_1 = \cos(\theta_1) = N \cdot I,$$
$$c_2 = \sqrt{1 - \left(\frac{n_1}{n_2}\right)^2 \sin^2(\theta_1)} \rightarrow \sqrt{1 - \left(\frac{n_1}{n_2}\right)^2 (1 - \cos^2(\theta_1))}$$

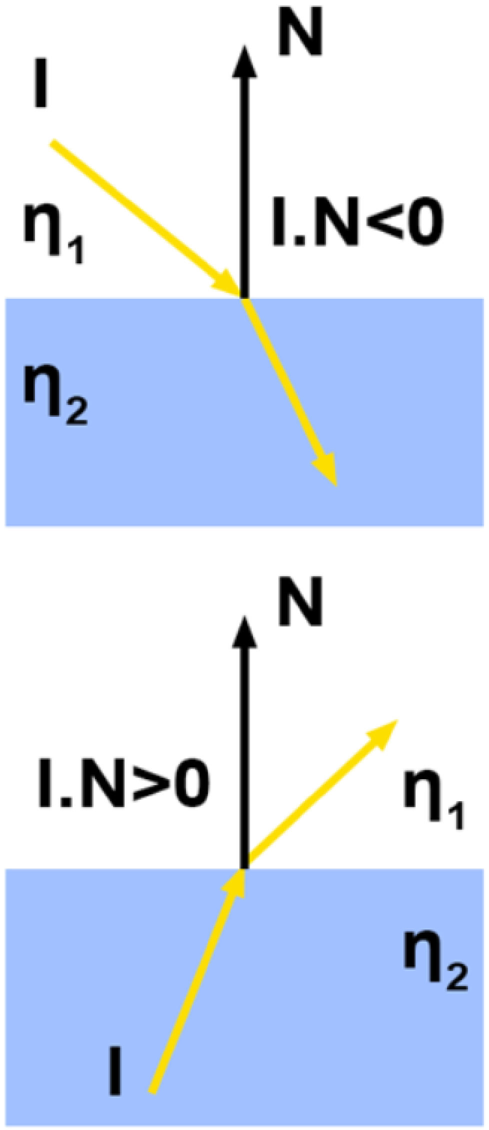


https://www.britannica.com/science/refraction



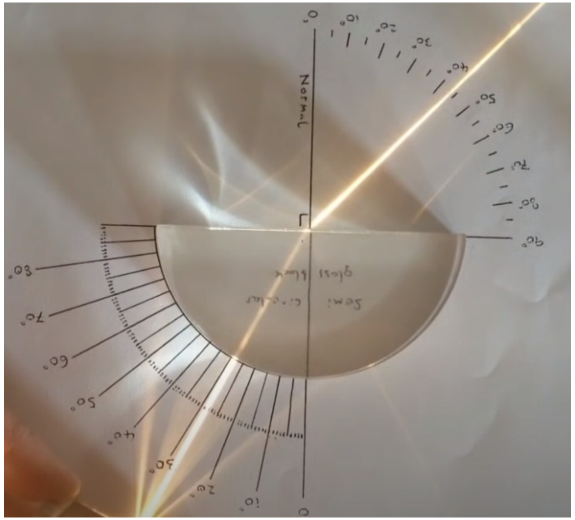
BTDF: specular transmission

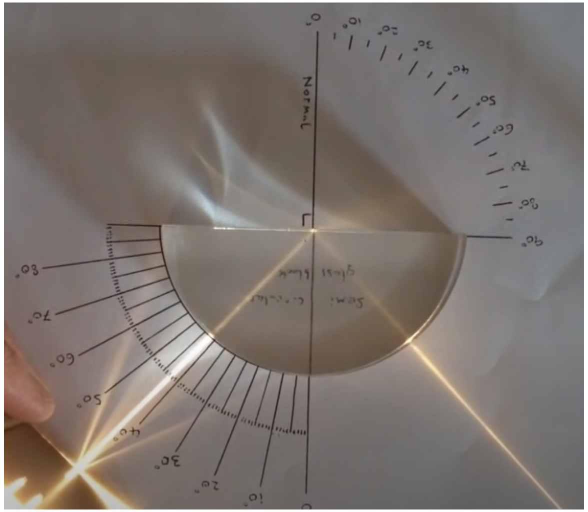
- 3D objects represent surfaces encapsulating media with different IOR
- Space around 3D objects is often considered vacuum → lowest IOR
- Light can travel:
  - From lower to higher IOR medium, e.g., light hitting surface from outside
  - From higher to lower IOR medium, e.g., light leaves volume of water
- If light travels from inside object (from higher to lower IOR) then normal direction must be flipped and then used for computing normal direction
  - Check the sign of dot product between incident ray direction and normal



BTDF: specular transmission

- When angle of incidence is larger than critical angle, then 100% of the light is reflected
  - Happens when light ray passes from higher to lower IOR (e.g., glass-water)
  - **Total internal reflection**
  - Check: term under square root is negative $c_2 = \sqrt{1 - \left(\frac{n_1}{n_2}\right)^2 (1 - \cos^2(\theta_1))}$





https://www.youtube.com/watch?v=NAaHPRsveJk&ab_channel=QuantumBoffin

BRDF and BTDF

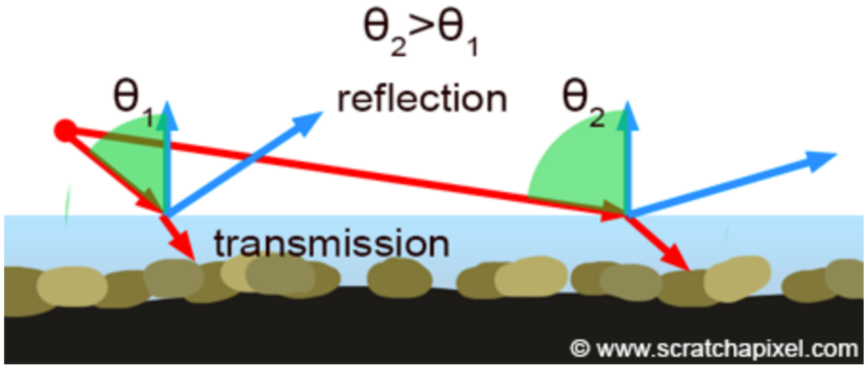
https://shanesimmsart.wordpress.com/2018/03/29/fresnel-reflection/

- How much light is reflected and refracted?
  - **Fresnel equations**
- Amount of reflected/refracted light depends on:
  - Media IOR
  - Angle of incidence → amount of transmitted light increases when angle of incidence decreases
- Amount of reflected/transmitted light (in case when total internal reflection doesn't occurs)

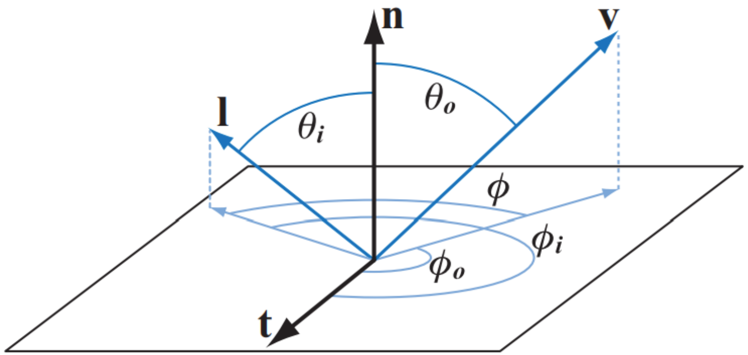
$$F_{R\parallel} = \left(\frac{\eta_2 \cos\theta_1 - \eta_1 \cos\theta_2}{\eta_2 \cos\theta_1 + \eta_1 \cos\theta_2} \right)^2,$$

$$F_{R\perp} = \left(\frac{\eta_1 \cos\theta_2 - \eta_2 \cos\theta_1}{\eta_1 \cos\theta_2 + \eta_2 \cos\theta_1} \right)^2.$$

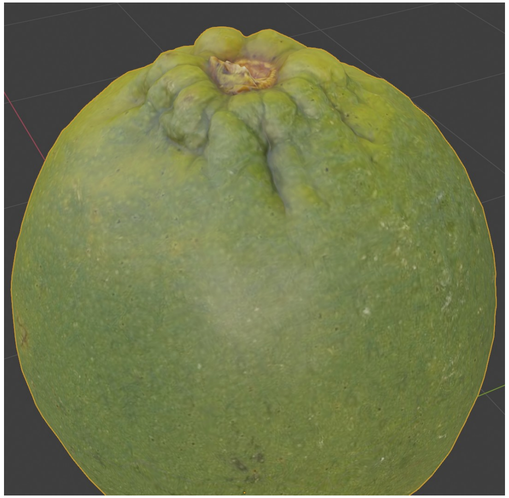
$$F_R = \frac{1}{2}(F_{R\parallel} + F_{R\perp}). \qquad F_T = 1 - F_R.$$

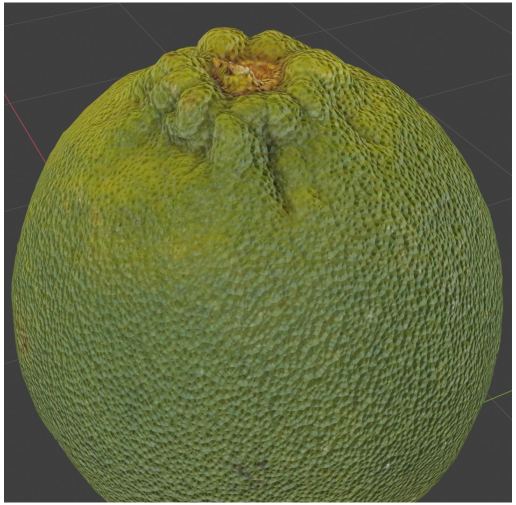


Normal and BRDF (BTDF)



- BRDF evaluation depends on surface normal
  - Normal defines so called **basis**
- Note that **perturbing the normal,** would tilt the basis and reflected light would be different!
- This enables modeling small **scale geometrical details**
- Variation of surface normal is task for **texturing**.





Scattering models parameters

- Scattering models are parameterized: color, roughness, etc.
- **Uniform parameters** result in overly smooth and perfect surface – not realistic.
- If BRDF depends on position on which is evaluated then it is called spatially varying BRDF
  - Variation of parameters over surface is done using **texturing**

Scattering models: practical tip

- In graphics, **various scattering models have been developed** (and still are!) to represent surface even more correctly or efficiently.
- Choice of the model depends on application and desired appearance.
- Practical tips:
  - **When modeling a material in DCC Tool**, you will be often offered with multiple implementations of basic (or more advanced) models that you further combine to achieve desired material description. In this case, it is good that you are familiar with how they work and their parameters because a lot of time is actually spent on "tweaking" parameters to achieve desired appearance. Understanding parameters of scattering models help very much with upcoming topic: texturing. This is huge and important topic.
  - If you are more interested in **developing your own scattering models** to achieve different appearance (not necessary photo-realistic, rather non-photorealistic which will be discussed later) then understanding of existing scattering models is great foundation to build on: you will see that advancements of scattering models just added more complexity to basic ones.

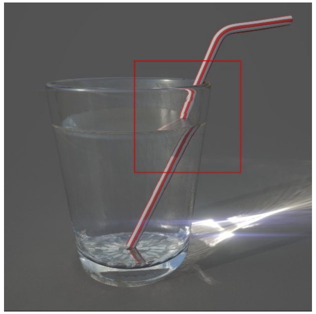
Exploring BSDFs

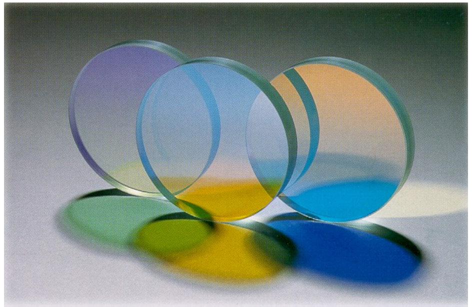
- Various BSDFs are available in modeling tools for material creation. Core scattering models are deeply integrated into renderer source code and user is provided with an interface to those for combining and creation of complex material.
  - Cycles/EEVEE (Blender): https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/index.html
  - Appleseed: https://appleseed.readthedocs.io/projects/appleseed-maya/en/master/shaders/shaders.html#materials
- Similarly as for object shape (e.g., mesh), materials are meant to be transferable between applications. Note that it is up to applications renderer which scattering models are supported. Therefore, it is often a case that material defined in one modeling tool can not be easily fully and exactly transferred to another application.
  - This requires matching supported scattering functions between applications. Example is Blender to Unity
- In order enable easier communication and transfer, standardized BSDF is created and supported by diferent applications.
  - Principled BSDF: Blender to Godot: https://docs.godotengine.org/en/3.0/tutorials/3d/spatial_material.html
- Tendency is towards integrations of material modeling tool standards into game engines for easier transfer
  - Example: https://substance3d.adobe.com/plugins/substance-in-unreal-engine/

Volumetric scattering

Light-matter interaction: summary

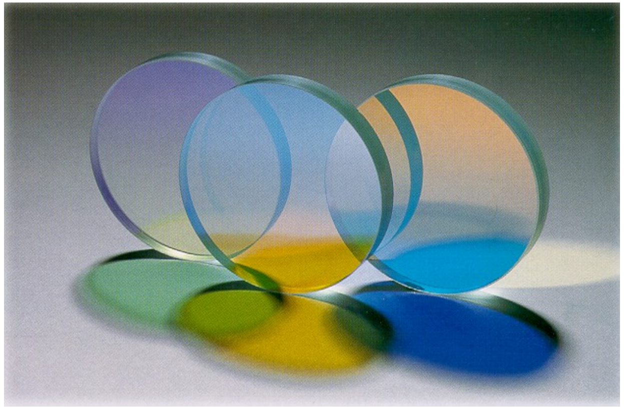
- Light traveling to/between objects can be scattered/absorbed → **volumetric scattering**
- Light falling on object surface – **surface scattering** - can:
  - Reflect (metals and dielectrics)
  - Refract (only dielectrics)
- Depending on material, refracted light can:
  - Transmit (transparent surfaces)
  - Sub-surface scatter (opaque and translucent object volume) → **volumetric scattering**





Volumetric scattering

- Light traveling to object can be attenuated by **participating media**
- Light which is not reflected from surface is scattering inside surface: **sub-surface scattering**.



Participating media

- Sky, atmosphere, clouds, fog, smoke, etc.

Sub-surface scattering

- Translucent objects: wax, skin, leafs, etc.

Volumetric scattering

- Complex topic which can not be described with direct (local) models
  - When it comes to shading, advanced light transport must be utilized to capture volumetric effects
- Often, for real-time applications, volumetric scattering can be simulated to look correct, without physically-based computation

Light-matter interaction

- Ratio of light that gets through medium over certain distance is given by **Beer-Lambert Law**
- Simple volumetric fog effect can be simulated using Beer-Lambert Law:

$F = e^{-(d * z)}$, d – medium density, z – medium depth

To remember

- Material observation
  - Directional effects
  - Color
  - Spatial variation
- Optics for material
  - Scattering and absorption
  - Surface scattering
- Material model
  - BRDF
  - Scattering parameters

To remember

- Material observation
  - Directional effects
  - Color
  - Spatial variation
- Optics for material
  - Scattering and absorption
  - Surface scattering
- Material model
  - BRDF
  - Scattering parameters

More into topic

- Note that BSDFs describe interaction of light with surface (opaque or transparent)
- Other objects have specific appearance due to sub-surface scattering. Example for such material is wax. Such material is not transparent, but it is important what happens under surface since some light scatters outside and influences appearance – such material is called **translucent** and requires **BSSRDF***.
  - For simpler applications these materials can be approximated with diffuse BRDF (note that diffuse scattering is actually a result of sub-surface scattering and re-emitting as well as surface roughness)
  - Different approaches take phenomenological approach were they model what we observe in reality. They result in realistic appearance but are not physically correct: https://www.ea.com/frostbite/news/approximating-translucency-for-a-fast-cheap-and-convincing-subsurface-scattering-look
  - Finally, phyiscally based approaches model actual scattering of light inside of surface and its absorption, reflection and re-emission.
- Next to translucent surfaces, there are many phenomena for which is important to model light scattering inside a volume. These are called **volumetric rendering** approaches and they rely both on scattering function and light transport.
  - Volumentric rendering is highly researched and developed field: http://advances.realtimerendering.com/s2015/The%20Real-time%20Volumetric%20Cloudscapes%20of%20Horizon%20-%20Zero%20Dawn%20-%20ARTR.pdf
- Often cloth is important material to render. **BRDF for cloth** (e.g., sheen BRDF) are developed and investigated.
- We discussed light scattering on geometric optics. **Wave optics** is active research area.
- https://www.realtimerendering.com/#visapp

* https://www.pbr-book.org/3ed-2018/Volume_Scattering/The_BSSRDF

Literature

- https://github.com/lorentzo/IntroductionToComputerGraphics/wiki