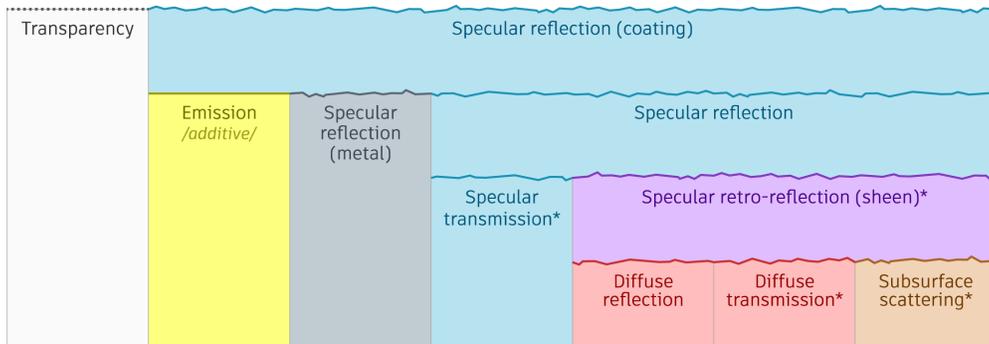


# Standard Surface



The *standard\_surface* shader is a physically-based shader capable of producing many types of materials. It includes a diffuse layer, a specular layer with complex Fresnel for metals, specular transmission for glass, subsurface scattering for skin, thin scattering for water and ice, a secondary specular coat, and light emission.

This shader models a material consisting of ten components that are layered and mixed hierarchically as illustrated in the diagram below. The properties of individual components can vary across the surface.



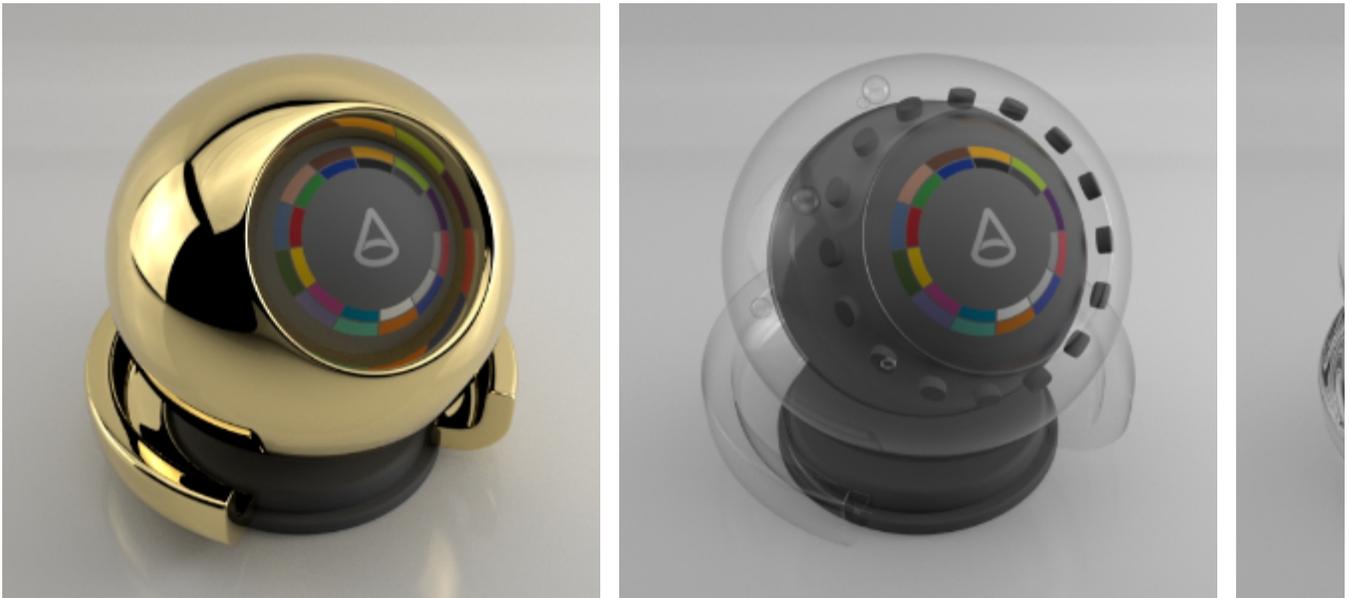
Schematic illustration of the idealized material model that *standard\_surface* emulates. Horizontal stacking of components represents statistical mixture and vertical stacking represents layering. Components marked with an asterisk are optional.

## Material Types

By default, the parameters are appropriate for materials such as plastic, wood or stone. By setting a few key parameters to 1, different types of materials can be quickly created:

- **Metalness:** gold, silver, iron, car paint.
- **Transmission:** glass, water, honey, soap bubble.
- **Subsurface:** skin, marble, wax, paper, leaves.
- **Thin Walled:** paper, leaves, soap bubble.

Parameter values between 0 and 1 may be used to create more complex materials that are a mix of basic material types.



## Energy Conservation

Standard Surface is energy conserving by default. All its layers are balanced so that the amount of light leaving the surface does not exceed the amount of incoming light. For example, as a surface is made more metallic and the specular layer contribution is increased, the diffuse layer contribution is reduced accordingly to ensure energy conservation.



Diffuse and rough (left) to metallic specular (right).

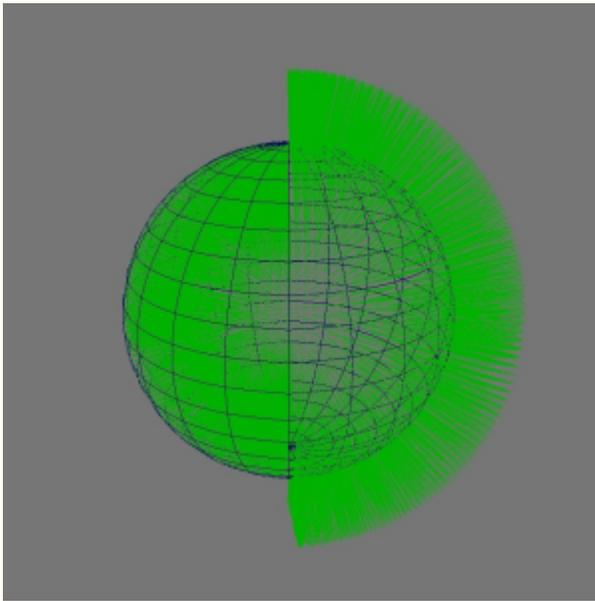


When using layer weights or colors with values higher than 1, energy conservation is broken. It is discouraged to create such materials, as they will not behave predictably under different lighting, and may lead to increased noise and poor rendering performance.

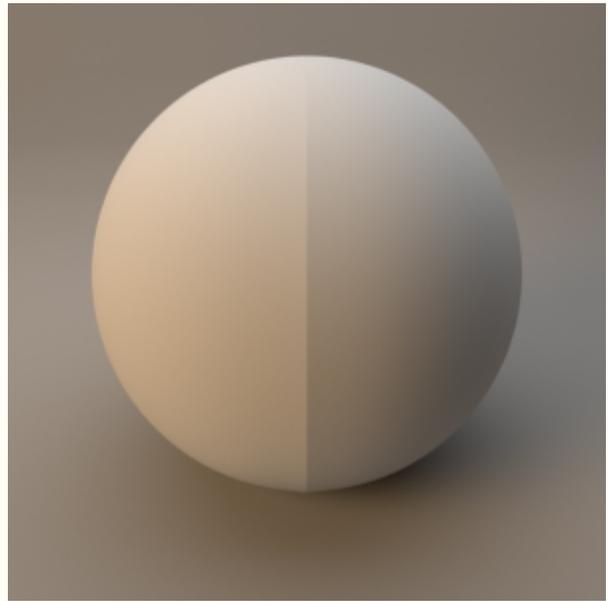


### Surface Normal Direction

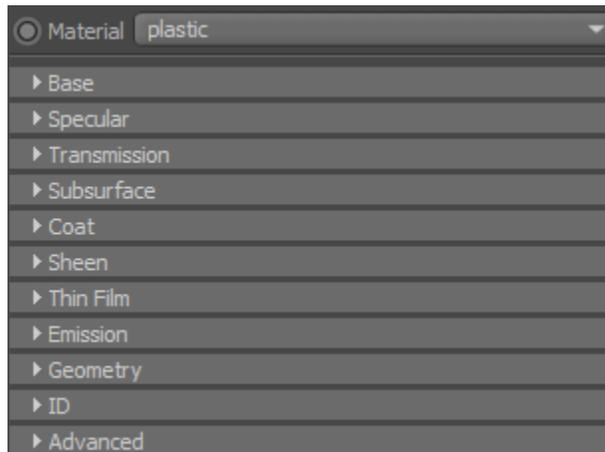
When rendering diffuse surfaces, it is very important that the normals of the geometry face in the right direction. In the example below, you can see the difference between normals that are facing inwards in the wrong direction (left side) versus those that are facing correctly in the outwards direction (right side).



The normals are reversed on the left side of the sphere



The right side of the sphere renders correctly



Due to the large number of controls, the *Standard Surface* shader is split up into several groups. The individual settings for each group are described in more detail in the pages below.

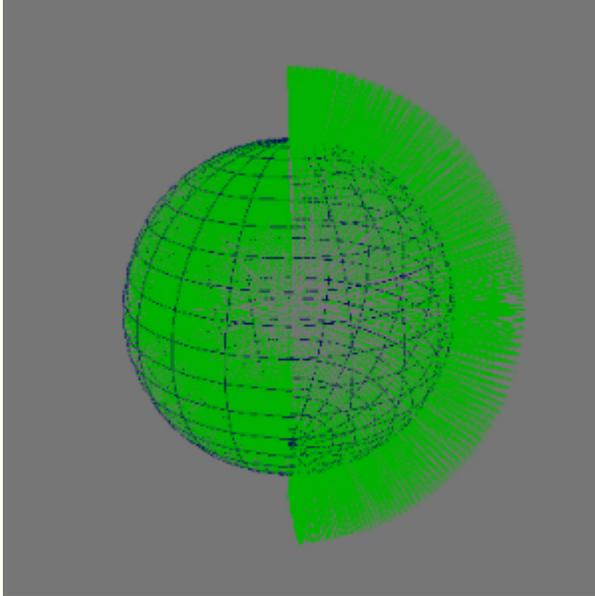
- [Base](#)
- [Specular](#)
- [Transmission](#)
- [Subsurface](#)
- [Coat](#)
- [Sheen](#)
- [Thin Film](#)
- [Emission](#)
- [Geometry](#)
- [Advanced](#)
- [ID AOVs](#)

 The C4DtoA material library can be found [here](#).

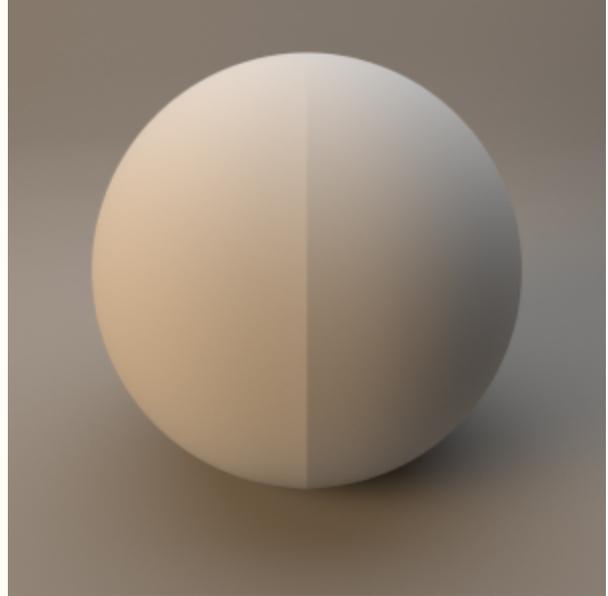
 Further information about physically based rendering in Arnold can be found [here](#).

### **Surface Normal Direction**

When rendering diffuse surfaces it is very important that the normals of the geometry face in the right direction. In the example below you can see the difference between normals that are facing inwards in the wrong direction (left side) versus those that are facing correctly in the outwards direction (right side).



The normals are reversed on the left side of the sphere



The right side of the sphere renders correctly