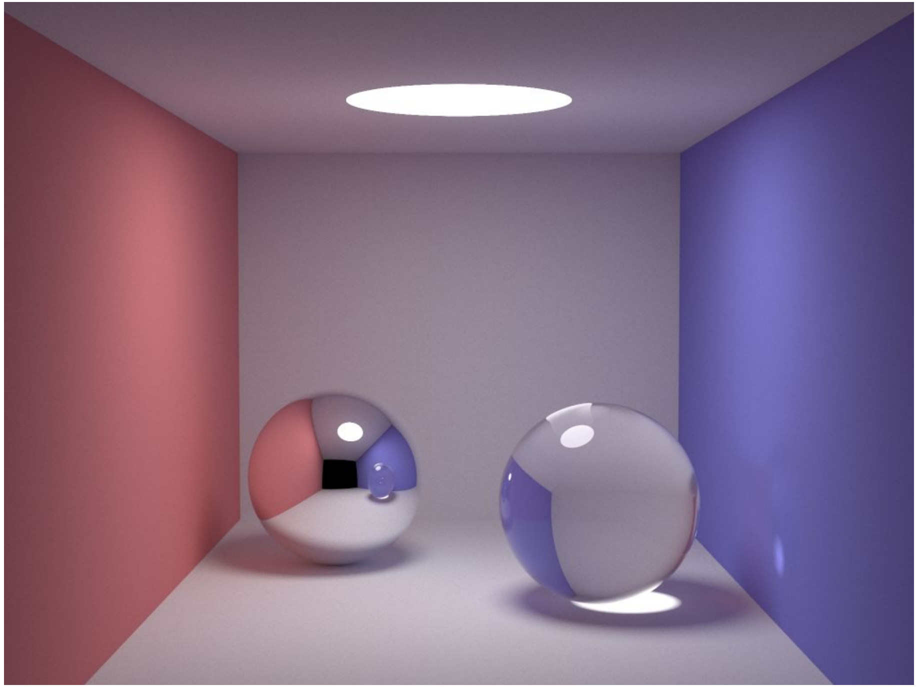
Dielectrics
and
Constructive Solid Geometry

Presented by: Richard Olney, Ashima Shrivastava

Dielectrics: Transparency and Refraction



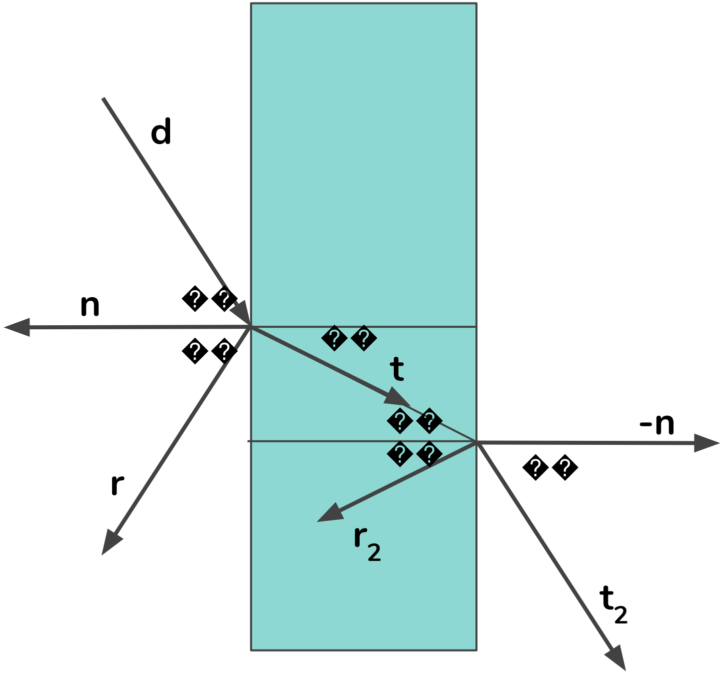
Kevin Beason

Physics: Snell’s Law

$$n \sin \theta = n_t \sin \phi.$$

Refractive indexes n and n_t
where $n_t > n$

Refractive indexes of air is 1.0

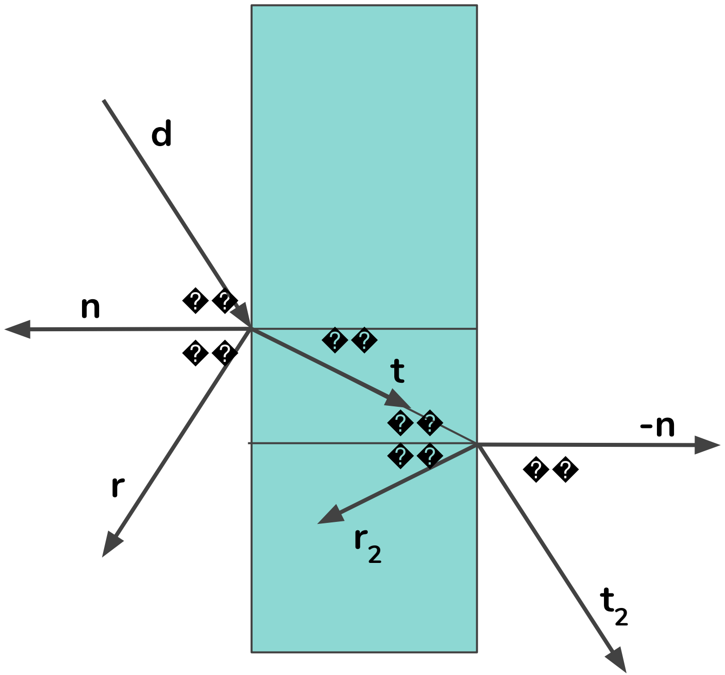


Physics: Some Derivations

$$n \sin \theta = n_t \sin \phi.$$

$$\sin^2 \theta + \cos^2 \theta = 1$$

$$\cos^2 \phi = 1 - (n^2 (1 - \cos^2 \theta))/ n_t^2$$



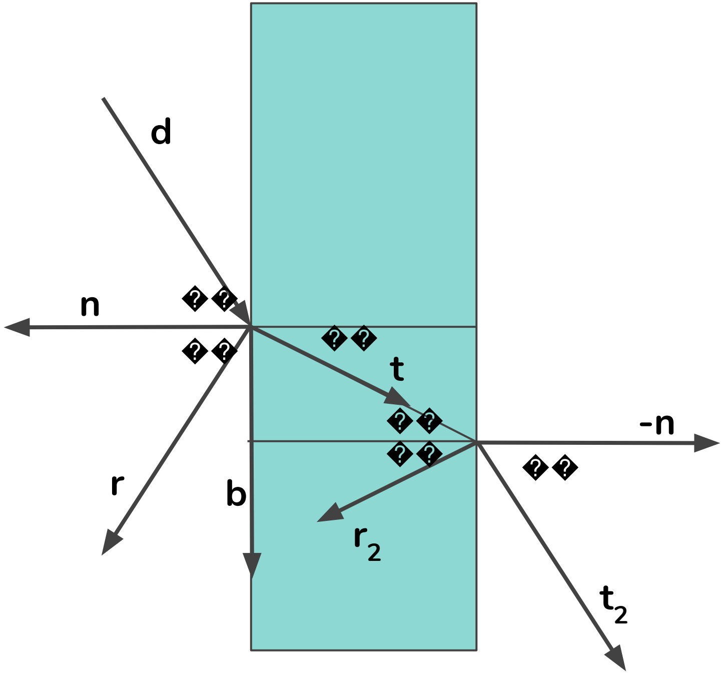
Physics: Orthonormal basis

n and **b** form an orthonormal basis

$$\mathbf{t} = \sin\phi\,\mathbf{b} - \cos\phi\,\mathbf{n}$$

$$\mathbf{d} = \sin\theta\,\mathbf{b} - \cos\theta\,\mathbf{n}$$

$$\mathbf{b} = (\mathbf{d} + \cos\theta\,\mathbf{n}) / \sin\theta$$



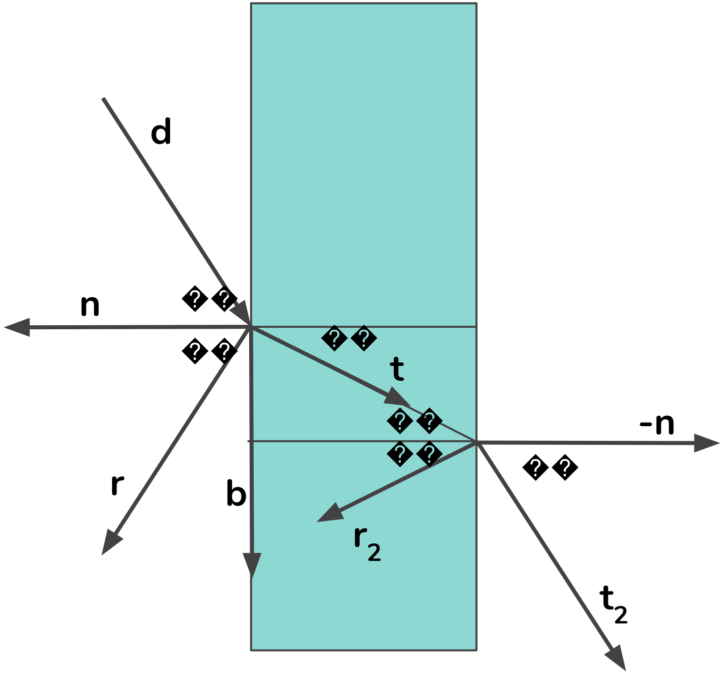
Physics: Salve for t

$$\mathbf{t} = (n^2\,(\mathbf{d} + \cos\theta\,\mathbf{n}))/n_t^{\,2} - \cos\phi\,\mathbf{n}$$

$$= (n^2(\mathbf{d}+\mathbf{n}(\mathbf{d}\square\mathbf{n})))/n_t^{\,2}$$

$$- \mathbf{n}\sqrt{}(1-(n^2(1-(\mathbf{d}\square\mathbf{n})^2))/n_t^{\,2})$$

Total Internal Reflection if the discriminant is < 0



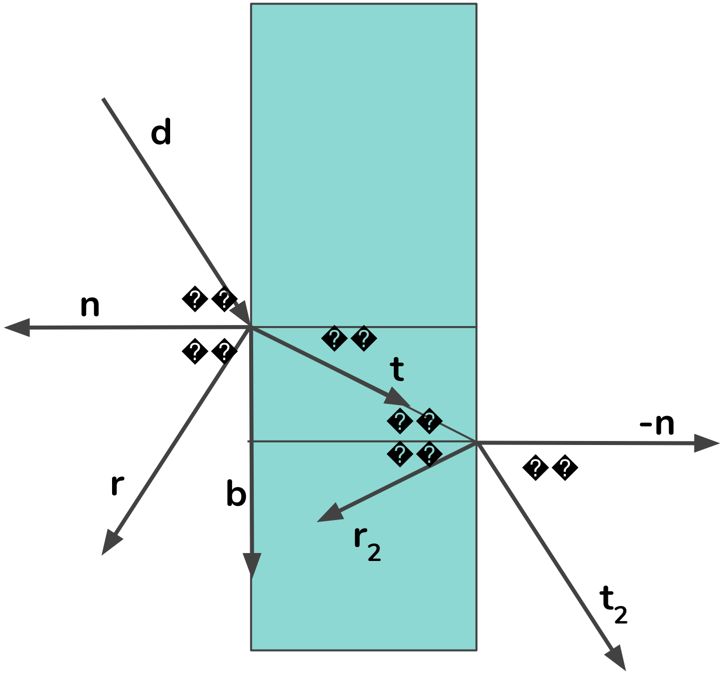
Schlick approximation

$$R(\theta) = R_0 + (1 - R_0)\,(1 - \cos\theta)^5$$

$$R(\theta) = R_0 + (1 - R_0)\,(1 - \mathbf{d}\square\mathbf{n})^5$$

$$R_0 = ((n_t - 1)/(n_t + 1))^2$$

θ angle in air, and n = 1.0



Light Attenuation

$$I = k \exp(-Cx)$$

$$I(0) = I_0$$

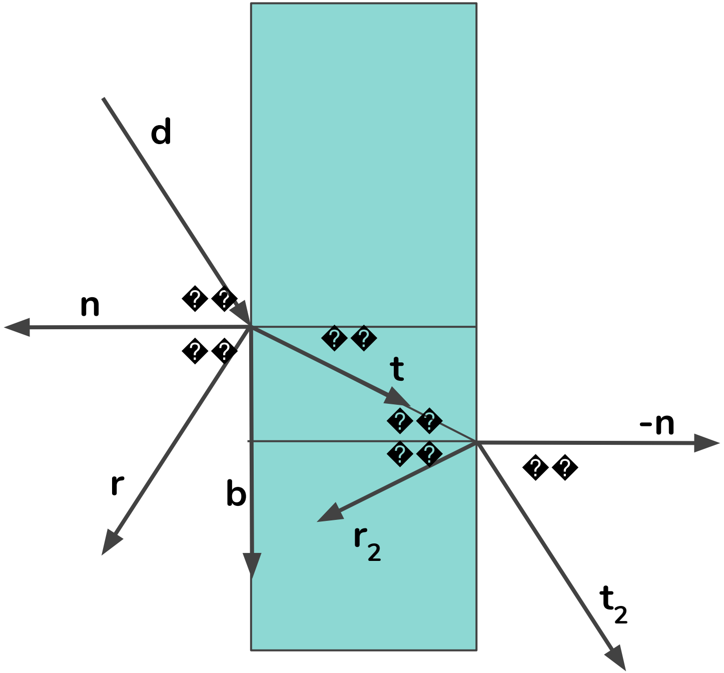
$$I(x) = I_0 \exp(-Cx)$$

$$I(1) = aI_0$$

$$I_0 a = I_0 \exp(-C) \implies -C = \ln(a)$$

$$I(s) = I(0)e^{\ln(a)s}$$

I(s) is the intensity of the beam at distance s from the interface, but a generally needs to be hand tuned.

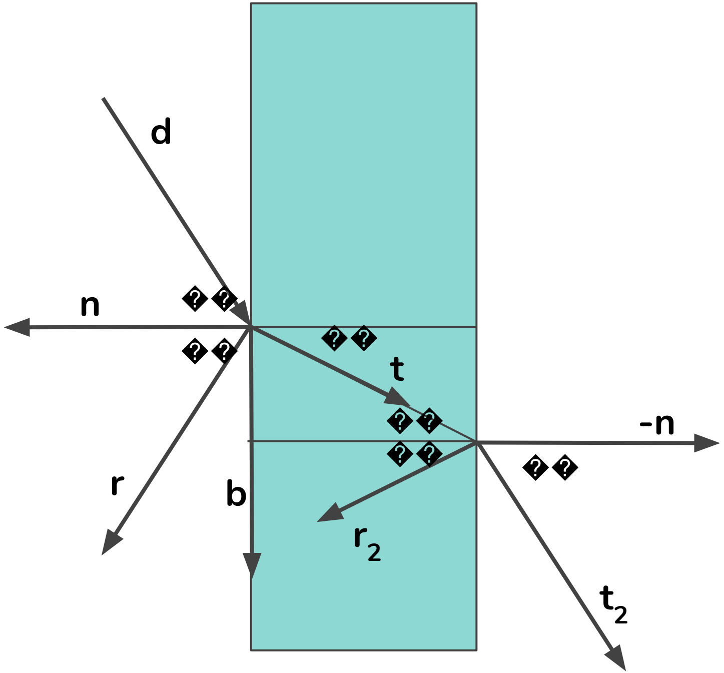


Implementation

Introduce a dielectric material, with information on light attenuation constants.

Check for dielectric materials before you would check for mirror reflection.

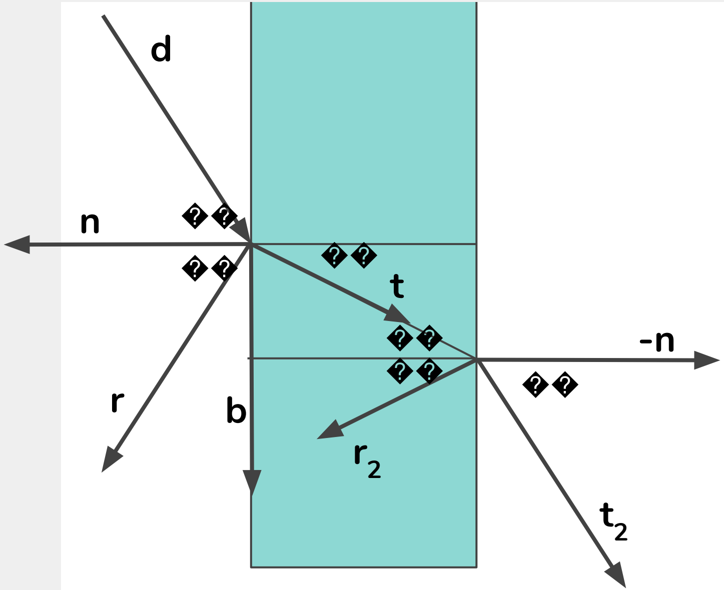
Then use the following pseudo-code to spawn your new rays.



Pseudo Code

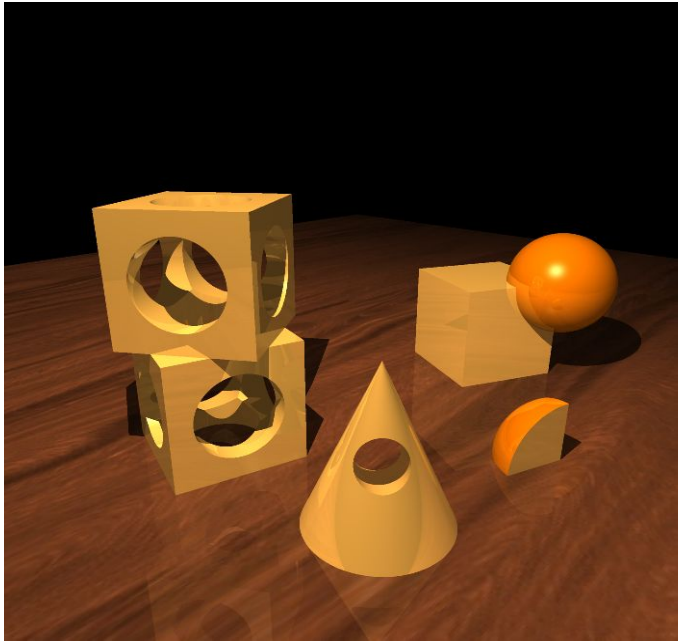
```
if ($\mathbf{p}$ is on a dielectric)
        $\mathbf{r}$ = reflect($\mathbf{d}$, $\mathbf{n}$ )
        if ($\mathbf{d} \cdot \mathbf{n} < 0$)
                refract($\mathbf{d}$, $\mathbf{n}$, n, $\mathbf{t}$)
                $c = -\mathbf{d} \cdot \mathbf{n}$
                $k_r = k_g = k_b = 1$
        else
                $k_r = \exp(-a_r t)$
                $k_g = \exp(-a_g t)$
                $k_b = \exp(-a_b t)$
                if refract($\mathbf{d}$, $-\mathbf{n}$, 1/n, $\mathbf{t}$)
                        $c = \mathbf{t} \cdot \mathbf{n}$
                else
                        return $k * \text{traceray}(\mathbf{p} + t\mathbf{r})$
        $R_0 = ((n-1)/(n+1))^2$
        $R = R_0 + (1 - R_0)(1 - c)^5$
        return $k(R\ \text{traceray}(\mathbf{p} + t\mathbf{r}) + (1 - R)\ \text{traceray}(\mathbf{p} + t\mathbf{t}))$
```

```
# refract function returns false if there is total internal reflection,
# otherwise fills t
```



Constructive Solid Geometry (CSG)

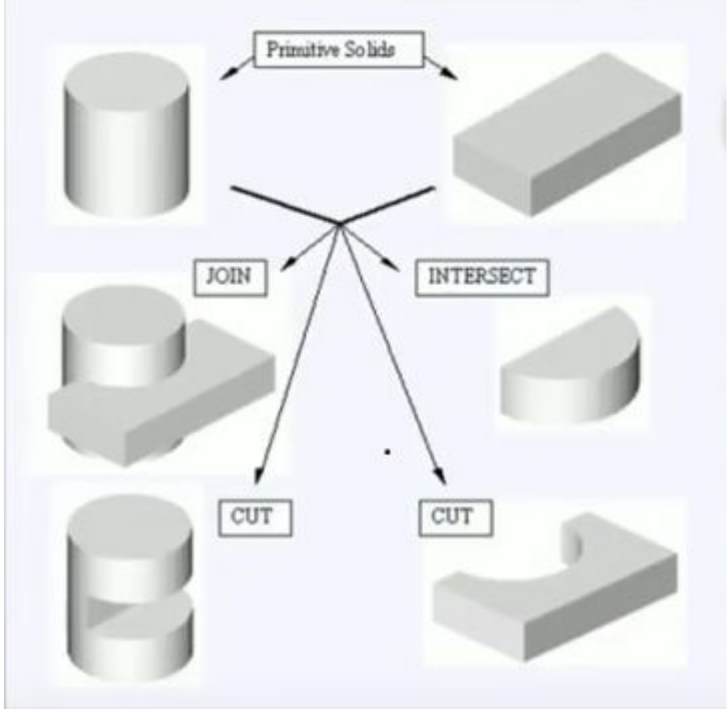
- Constructive solid geometry is using combinations of basic 3D shapes to create models.
- Major advancement in solid modeling.
- The shapes are joined using various operations.
- Final geometry is created.



CSG - 3 Steps

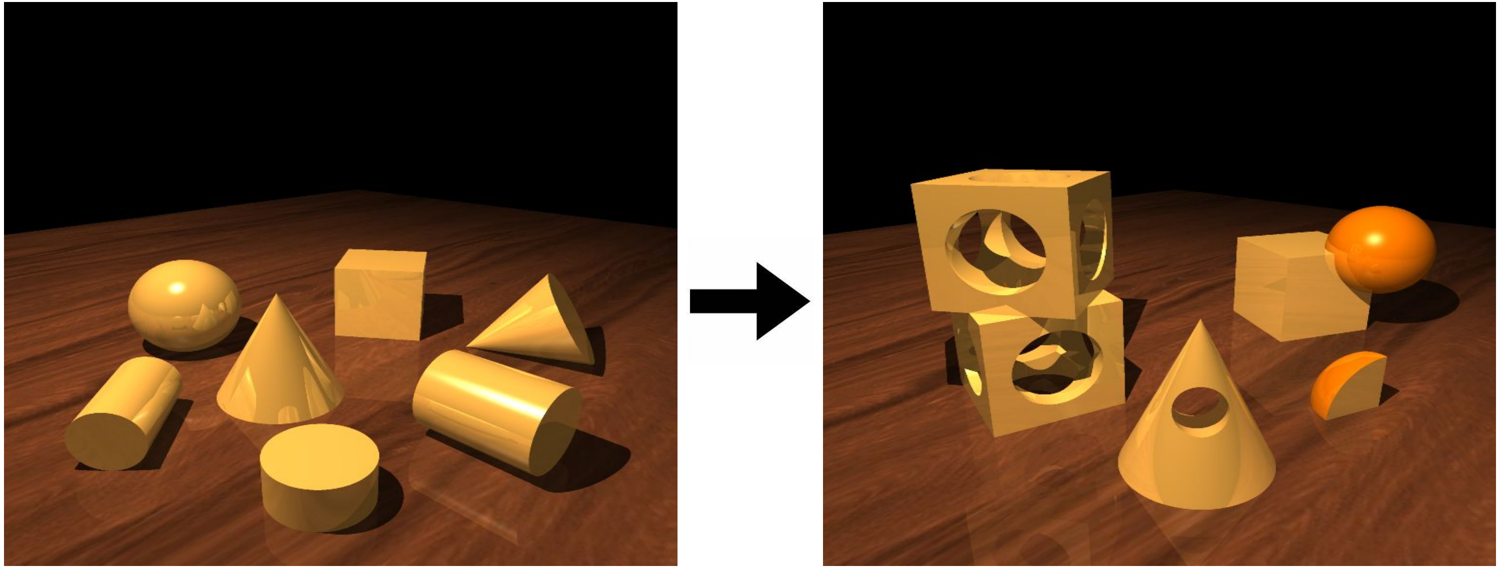
There are three steps for solid modeling using constructive solid geometry :

- Primitive solids
- Boolean operations
- Creating binary tree



CSG - Primitive Solids

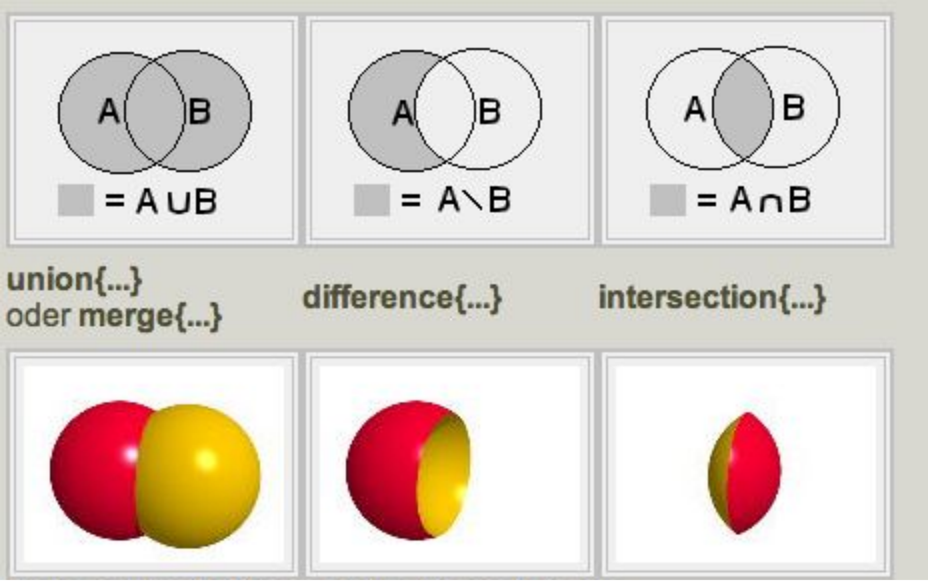
The very first solid that we assemble to create any solid geometry are the primitive solids.



CSG - Boolean Operations

Performing boolean operations on the solids to generate a new geometry

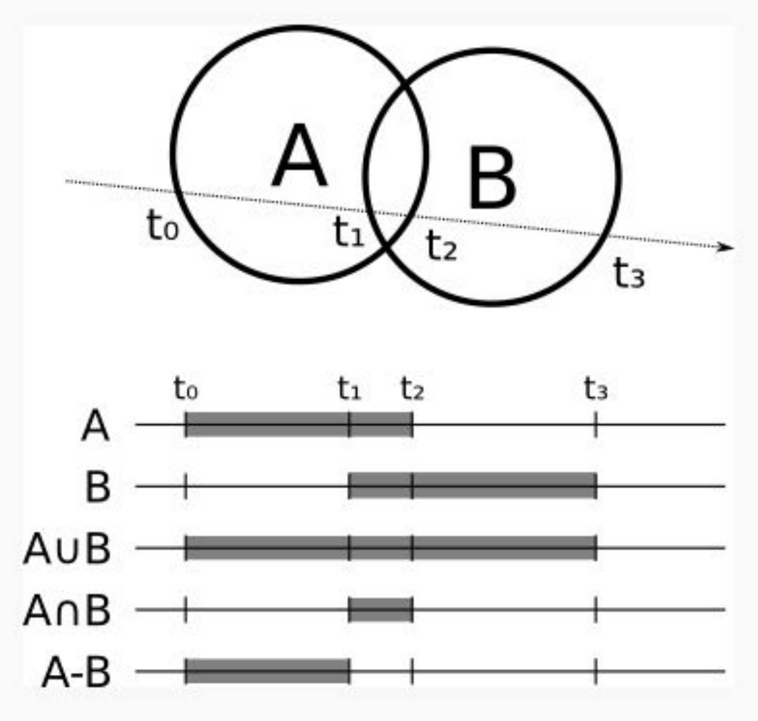
- A union B
- A difference B
- A intersection B



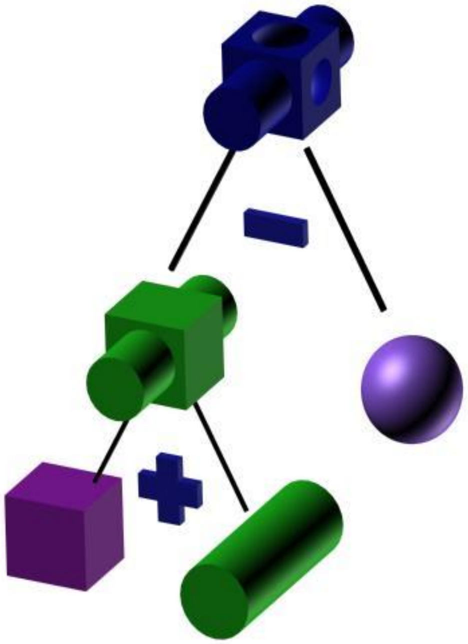
Boolean operations conti.

For every object you can compute where the ray enters and exits the object.

Compute the intersection between the ray and A and B separately (which gives inside range for each object) to compute intersection between ray and object AB. Then compute inside range for AB.



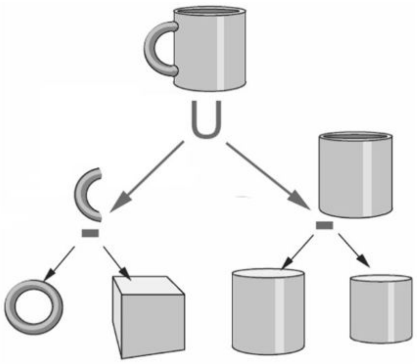
CSG - Binary Tree



To represent CSG method we use binary tree.

Binary trees are used to show how different primitives are used to create new solids.

- Leaves - Primitive solids
- Root - Resulting solid



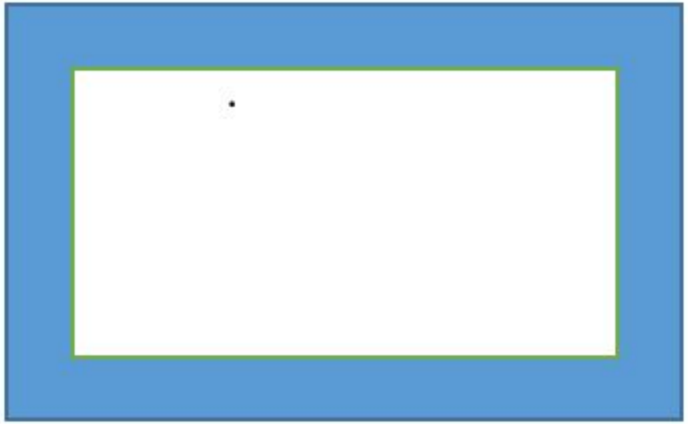
CSG - Example

Create CSG of following solid -

Steps -

- Find primitive used - 2 cuboid
- Boolean operation - A - B
- Tree structure - WhiteBoard

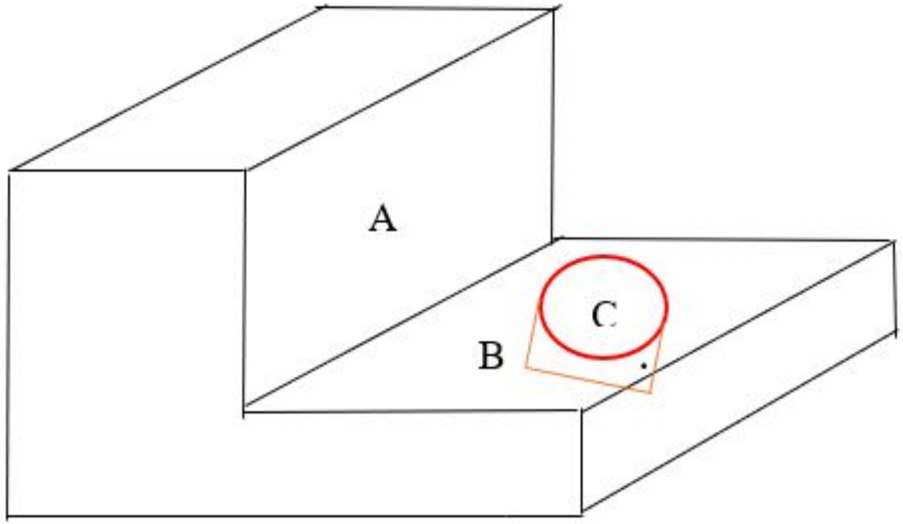
(WhiteBoard figure)



Question

Create CSG model for the given geometry :

- Primitive solid -
- Boolean geometry -
- Tree -



Thank you!!

See chapter 13 of Fundamentals of Computer Graphics, Fourth Edition by Marschner and Shirley for more details on these topics.