Mechanics and Optics in Game Engines *(and in Game Design)*

Alejandro Garcia
Dept. Physics & Astronomy
San Jose State University

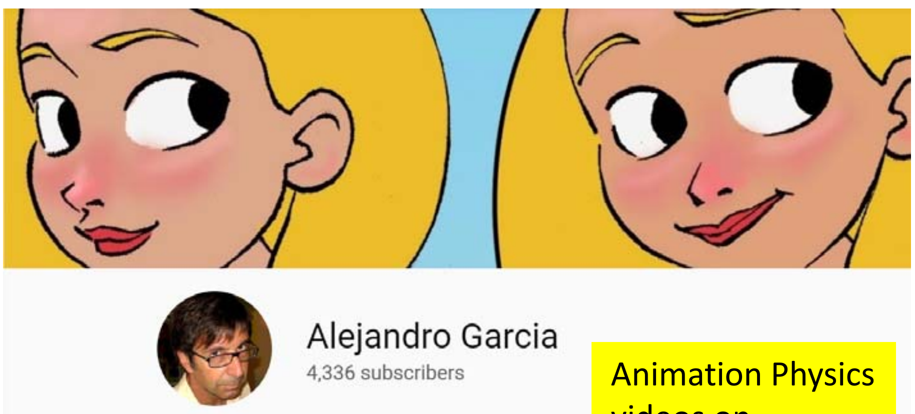
AAPT PHYSICS EDUCATION 2019 SUMMER MEETING
JULY 20-24 Provo, Utah

Gaming & Animation in the Bay Area

The San Francisco Bay area has a large concentration of animation and game studios.

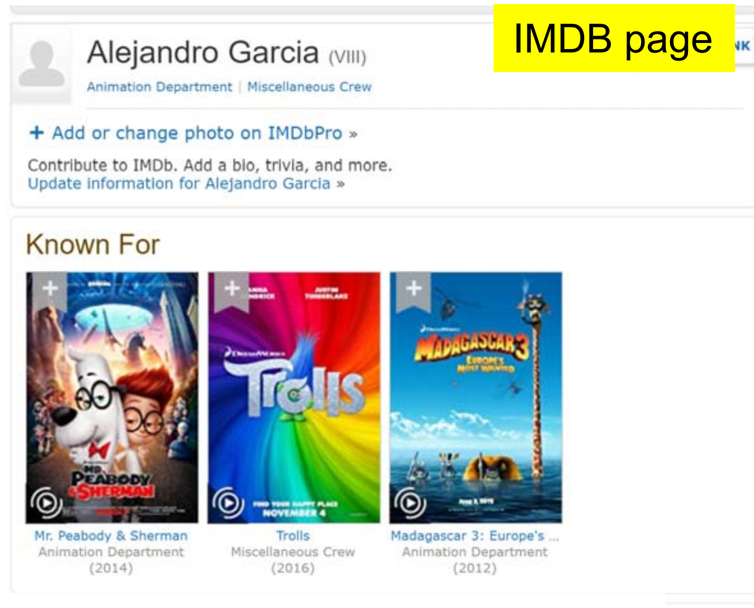
San Jose State has a large animation program (over 400 students) who graduate with an art degree

Created an Animation Physics course at SJSU (with NSF funding). Lower-division General Ed science course taken by animation majors.



Animation Physics videos on YouTube

IMDB page



Have also been a physics consultant for various film and gaming studios.

Case Study #1
Rigid Body Dynamics

What game designers care about
How physicists do it
How game engines do it
How physicists can help game designers

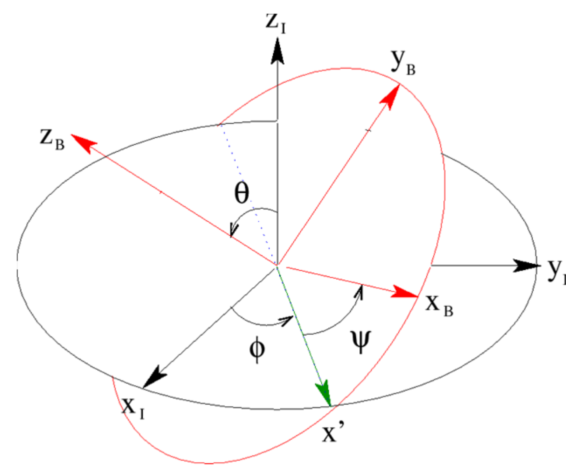
What game designers care about

Many video games require the movement of objects and characters to be physically realistic (or at least physically believable).

Some more so than others.

Rigid Body Dynamics - How physicists do it

Euler angles



Represent the orientation in terms of a Euler angles, (θ, ϕ, ψ)
Write the angular velocity, $\vec{\Omega}$, in the body frame as,

$$
\begin{aligned}
\Omega_1 &= \dot{\phi}\sin\theta\sin\psi + \dot{\theta}\cos\psi \\
\Omega_2 &= \dot{\phi}\sin\theta\cos\psi - \dot{\theta}\sin\psi \\
\Omega_1 &= \dot{\phi}\cos\theta + \dot{\psi}
\end{aligned}
$$

The Euler equations give the dynamics in the body frame as,

$$
\begin{aligned}
I_1\dot{\Omega}_1 &= (I_2 - I_3)\Omega_2\Omega_3 + \tau_1 \\
I_2\dot{\Omega}_2 &= (I_3 - I_1)\Omega_1\Omega_3 + \tau_2 \\
I_3\dot{\Omega}_3 &= (I_1 - I_2)\Omega_1\Omega_2 + \tau_3
\end{aligned}
$$

I : Moments of inertia
$\vec{\tau}$: Torque

From $\vec{\Omega}(t)$ convert back to Euler angles to find the orientation of the rigid body.

Rigid Body Dynamics- How game engines do it

Game engines compute rotation using **quaternions**, which are a 4D extension of complex numbers.

$$\tilde{q} = a + i\,b + j\,c + k\,d$$

where

$$i^2 = -1, \quad i\,j = k, \quad i\,k = -j$$
$$j^2 = -1, \quad j\,i = -k, \quad j\,k = i$$
$$k^2 = -1, \quad k\,i = j, \quad k\,j = -i$$

Quaternion arithmetic is similar to complex arithmetic but remember that i, j, and k do <u>not</u> commute.

Rotation of a position vector can be defined in terms of a rotation matrix, R, as

$$\vec{r}\,' = R\,\vec{r}$$

It can also be defined in terms of a rotation quaternion, $\tilde{q}$, as

$$\tilde{r}\,' = \tilde{q}\,\tilde{r}\,\tilde{q}^{-1}$$

where the position quaternion is defined as,

$$\tilde{r} = i\,r_x + j\,r_y + k\,r_z$$

It's straight-forward (but messy) to switch from rotation matrix to rotation quaternion.

$$R \leftrightarrow \tilde{q}$$

Rigid Body Dynamics- How game engines do it

Dynamics calculation is:

1) Update center of mass position using $\frac{d}{dt}\vec{x} = \vec{v}$

2) Update center of mass velocity using $\frac{d}{dt}\vec{v} = \vec{F}/m$

3) Update rotation quaternion using $\frac{d}{dt}\tilde{q} = \frac{1}{2}\tilde{\omega}\tilde{q}$

4) Update angular momentum using $\frac{d}{dt}\vec{L} = \vec{\tau}$

5) From the updated rotation quaternion calculate new rotation matrix

$$R \leftarrow \tilde{q}$$

7) Calculate the rotational inertia matrix as

$$I = R\, I_0^{-1}\, R^{\mathrm{T}}$$

8) Calculate the new angular velocity as

$$\vec{\omega} = I^{-1}\, \vec{L}$$

9) Recalculate the net force and net torque, if needed.

Further complexities:

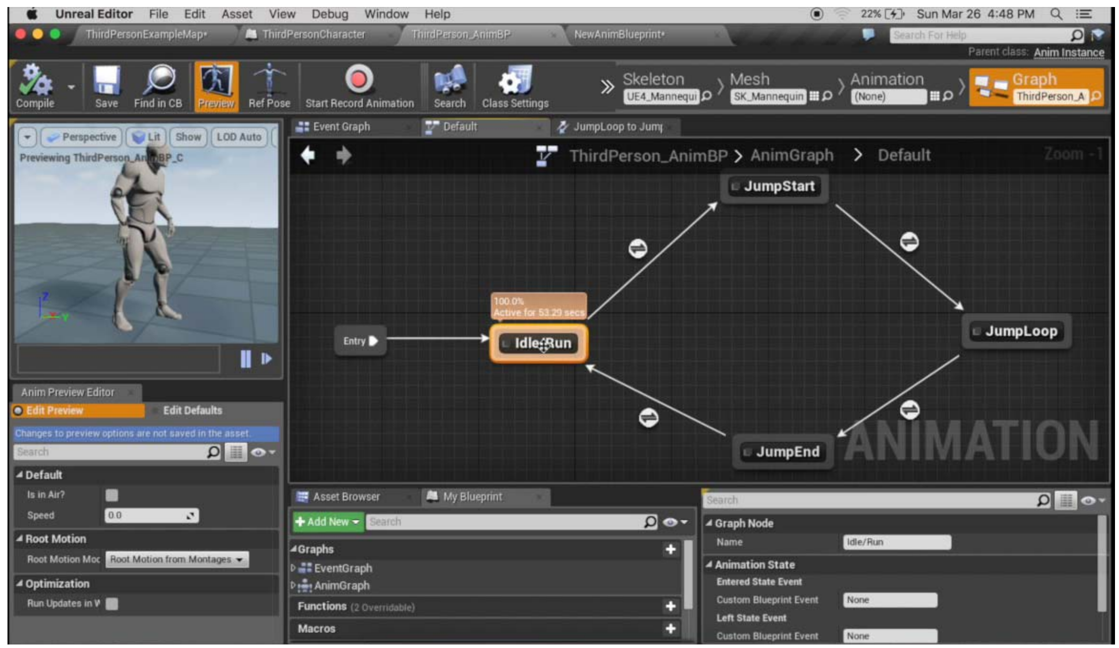
- Nonholonomic constraints (e.g., object collisions)
- Non-rigid bodies (e.g., Spider-man)

How physicists can help game designers

The motion of an object is not always calculated by the game engine's algorithms.

Game designers often have to create believable animations of physical motion by hand.

They either write special procedural routines or set individual poses as keyframes.



Wobbling

A symmetric object, like a football, may wobble as it spins.

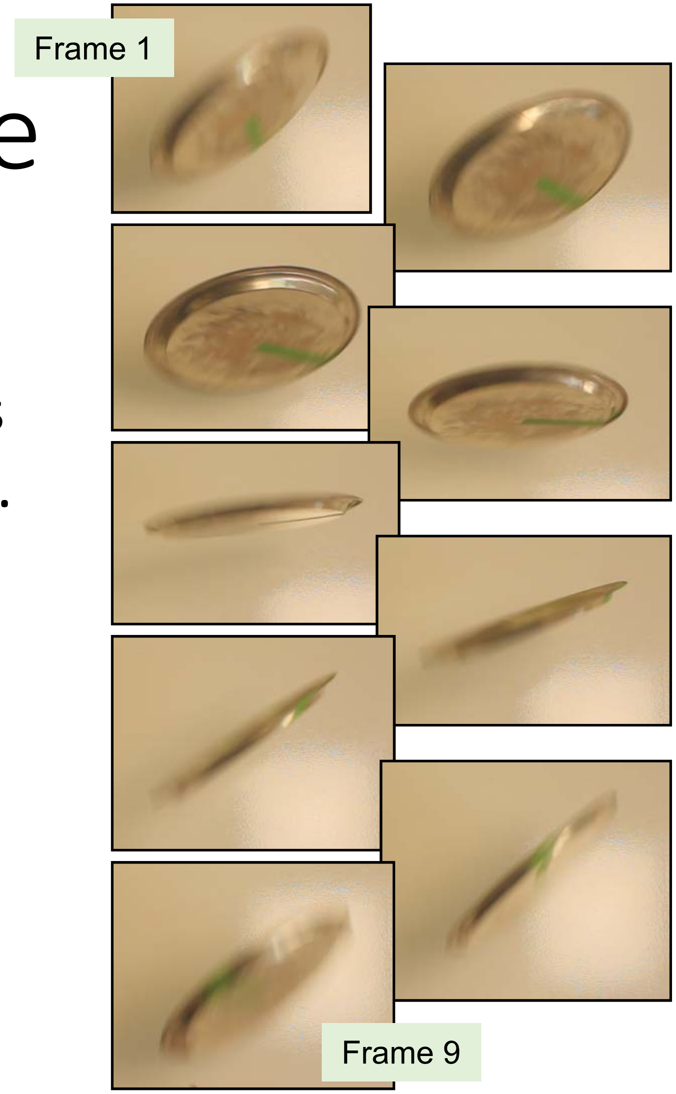
electron9.phys.utk.edu

The ratio of wobbles per spinning turn depends on the objects shape, for example:
<u>Football</u>: About 3 wobbles each 5 spin turns.
<u>Disk or Plate</u>: About 2 wobbles for each spin turn.

Wobbling Plate

The plate spins for a half-turn in eight frames (watch the green stripe).

During this time it does one full wobble (sides tilting up and down), returning to its original tilt angle.



Spinning & Wobbling Frequencies

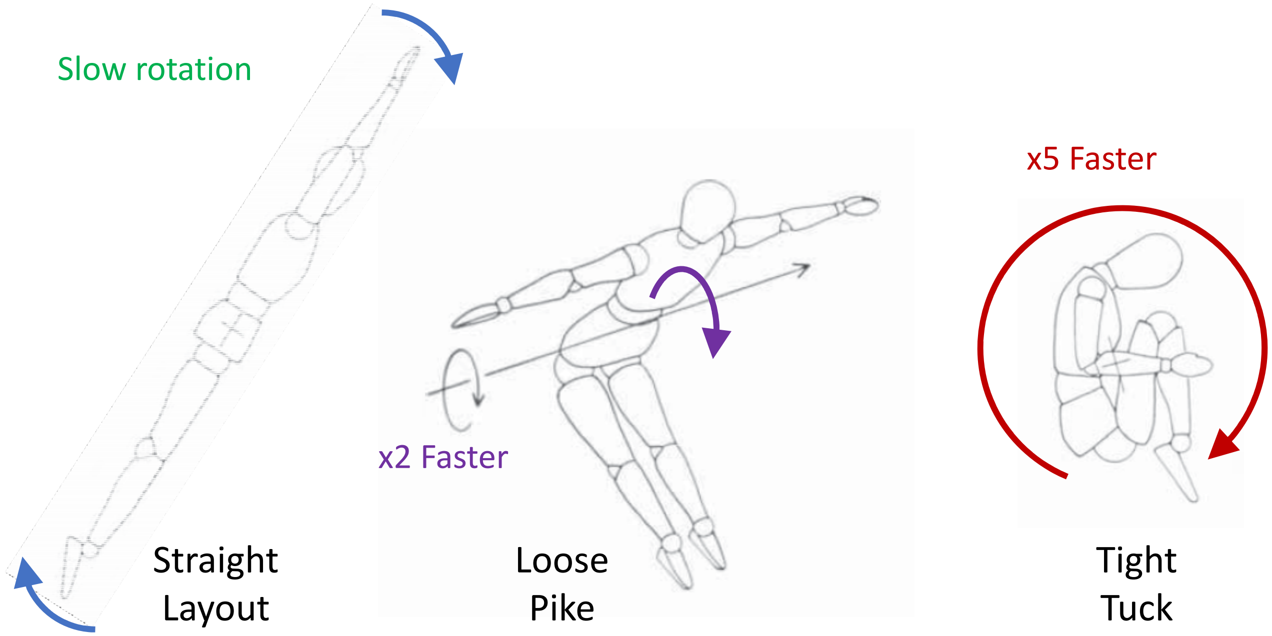
Objects can spin faster or slower but then the wobble frequency has to be consistent with the spinning frequency.

Increasing the spinning frequency increases the wobble frequency by the same proportion.

Wobble radius does not affect the frequency.

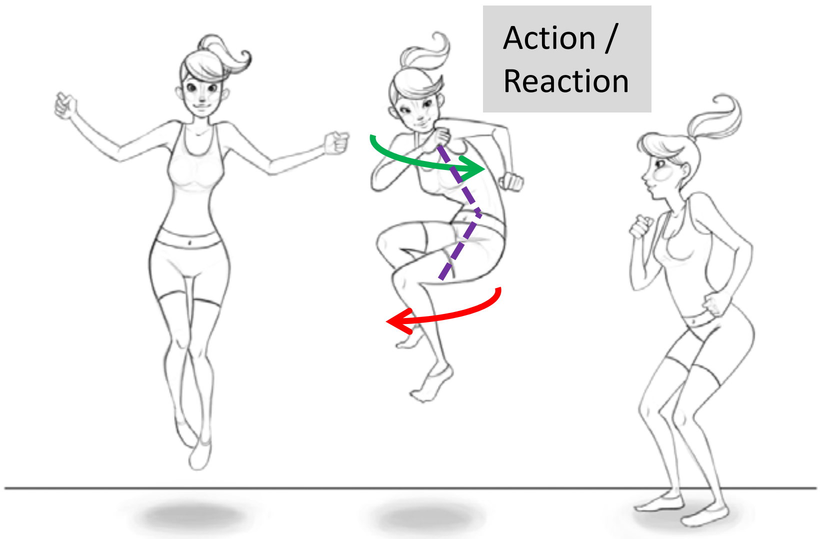
Rotation and Poses

Large change in rotation for various poses.

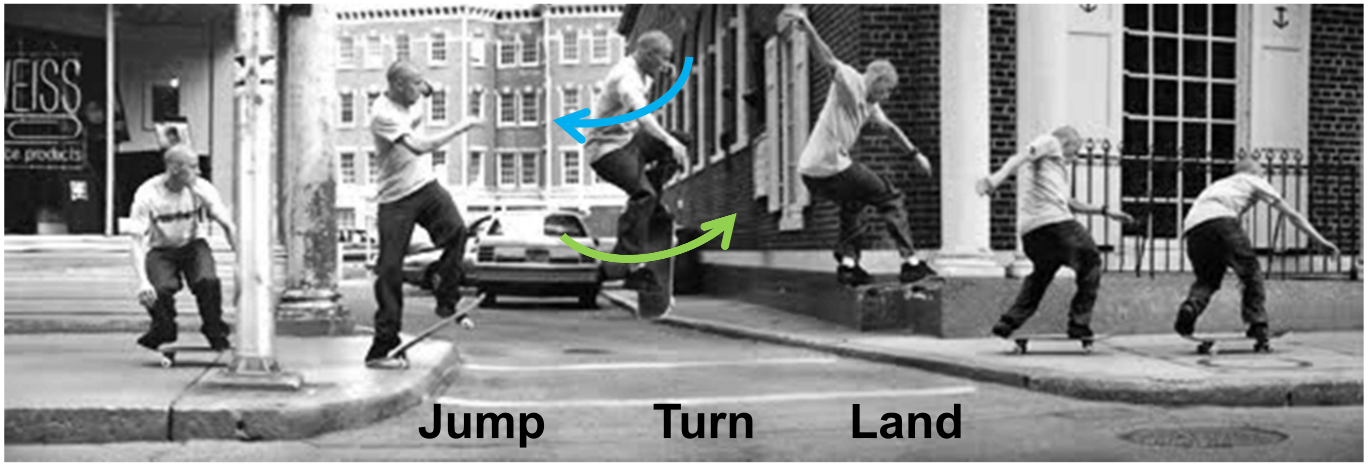


Torque-less Rotation

Twist upper body and lower body in opposite directions; after landing you straighten.



Front Side 180



The same principle is used in skateboarding tricks, such as a front side 180, in which a skater does a half turn in mid-air, turning upper and lower torso in opposite directions.

Case Study #2
Geometric Optics

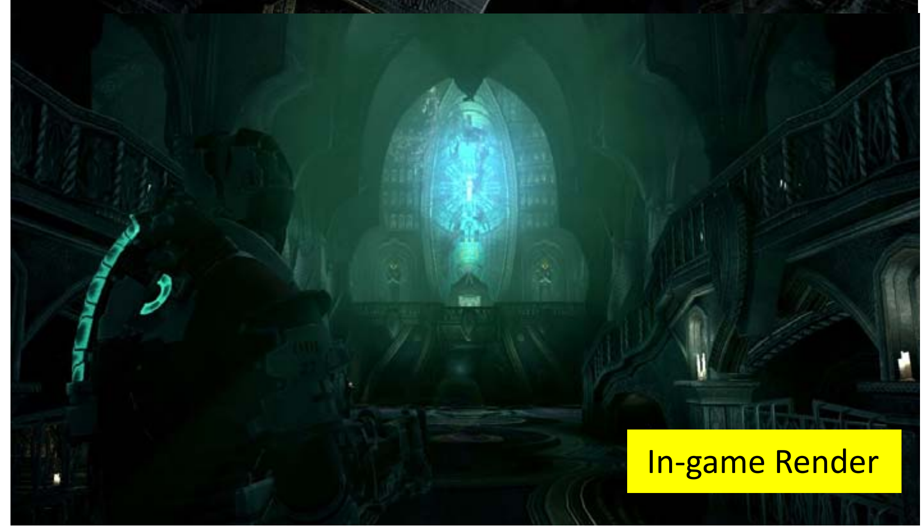
What game designers care about
How physicists do it
How game engines do it
How physicists can help game designers

What game designers care about

The visual style of a video game is established by concept art created before the game starts production.

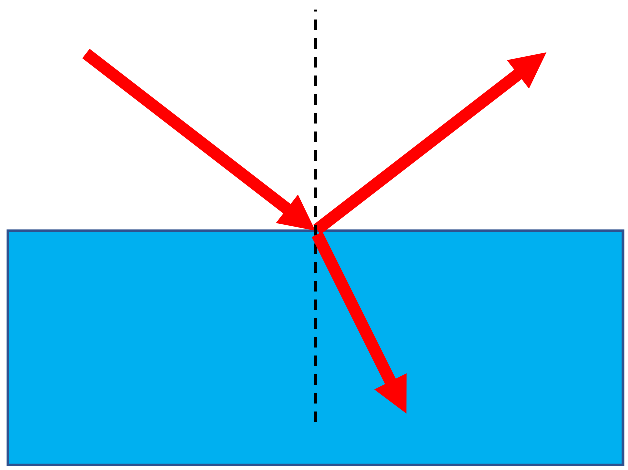
The game designers are guided by this concept art and will strive to reproduce within the game.

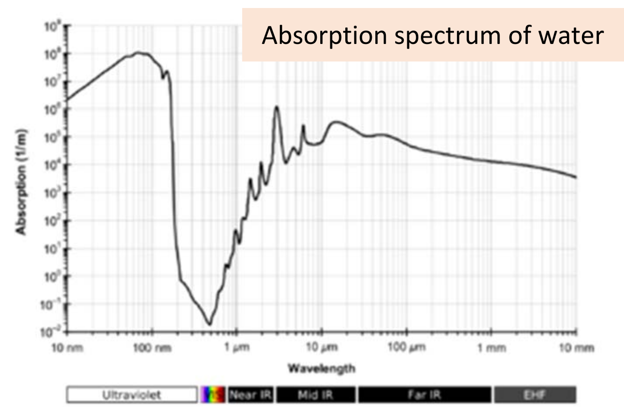
Modern games are increasingly adopting a cinematic visual style.



Geometric Optics
- How physicists do it

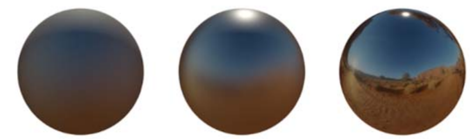
Light rays are calculated using basic principles such as specular reflection and Snell's law.

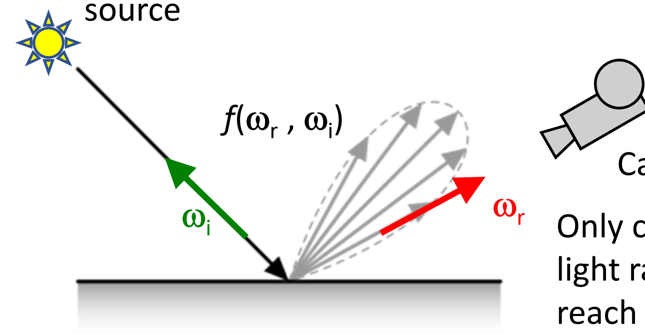




Color is defined by the absorption and transmission spectra of materials

Geometric Optics
- How game engines do it





A bidirectional reflectance distribution functions (**BRDF**) are defined specifying the ratio of reflected radiance exiting along ω_r to the irradiance incident on the surface from direction ω_i.

Also define

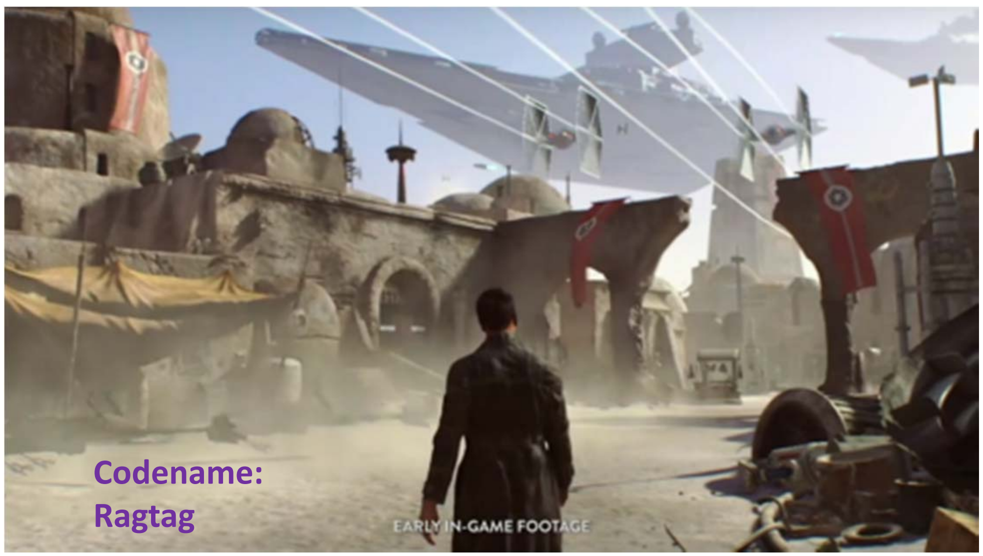
- BTDF - bidirectional transmittance distribution function
- BSDF - bidirectional scattering distribution function
- BSSRDF - bidirectional scattering-surface reflectance distribution function

A BxDF can also return an RGB value for the reflected radiance (i.e., color of the surface).

How physicists can help game designers

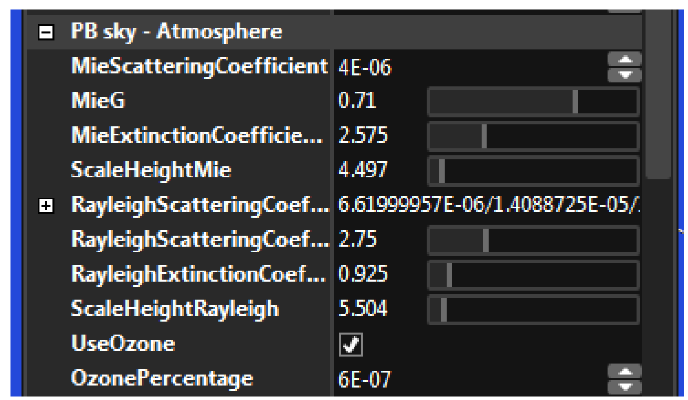
Games using Frostbite engine

In 2016 Visceral Games started working on a new Star Wars game using Electronic Arts' Frostbite game engine. The concept art indicated that the game would have a cinematic visual style similar to the Star Wars films.



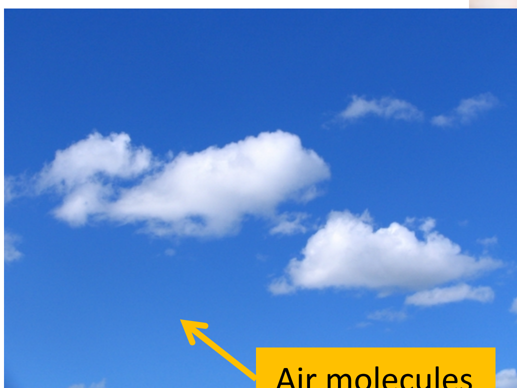
Codename: Ragtag

Artists at Visceral were unfamiliar with Frostbite so they asked me to explain the engine's built-in optics options, especially atmospheric effects.



Rayleigh Scattering

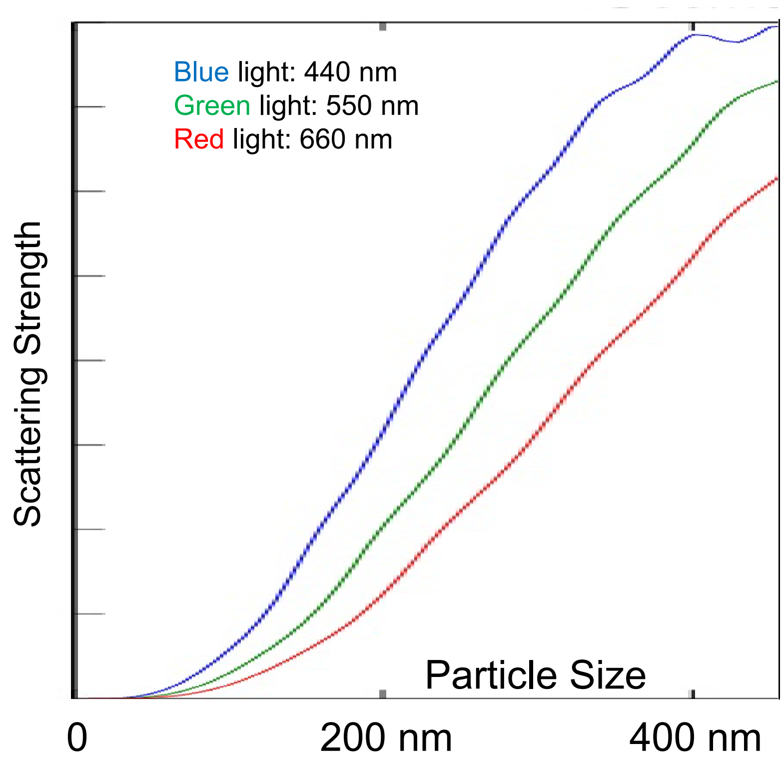
Tiny particles scatter blue light the most, red the least; white light is scattered with a hue shift to blue.



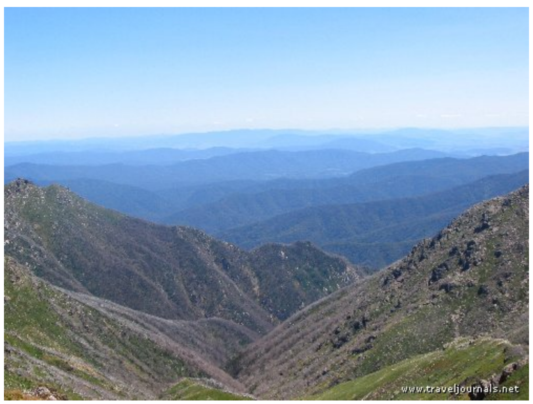
Particles smaller than wavelength of visible light.

Rayleigh Scattering & Color

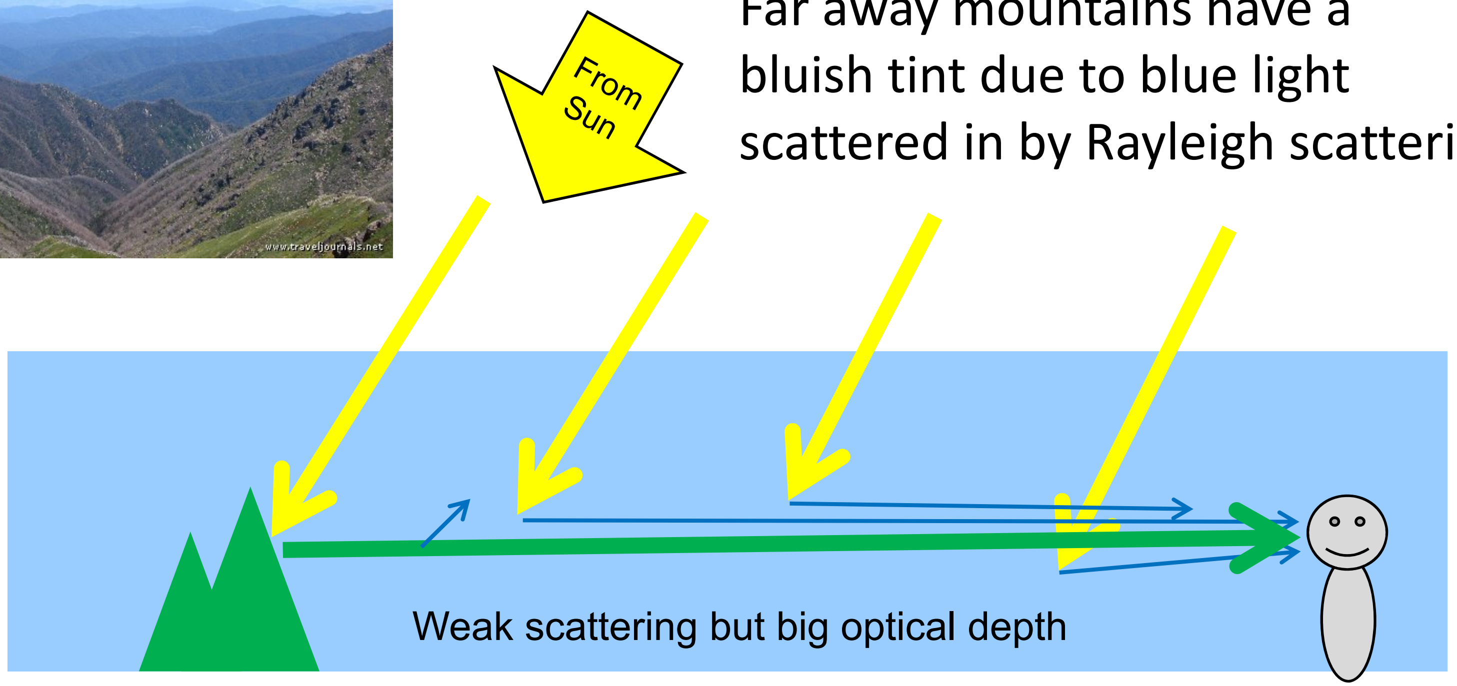
For tiny particles (under 400 nm) scattering is strongest for blue light and weakest for red light.



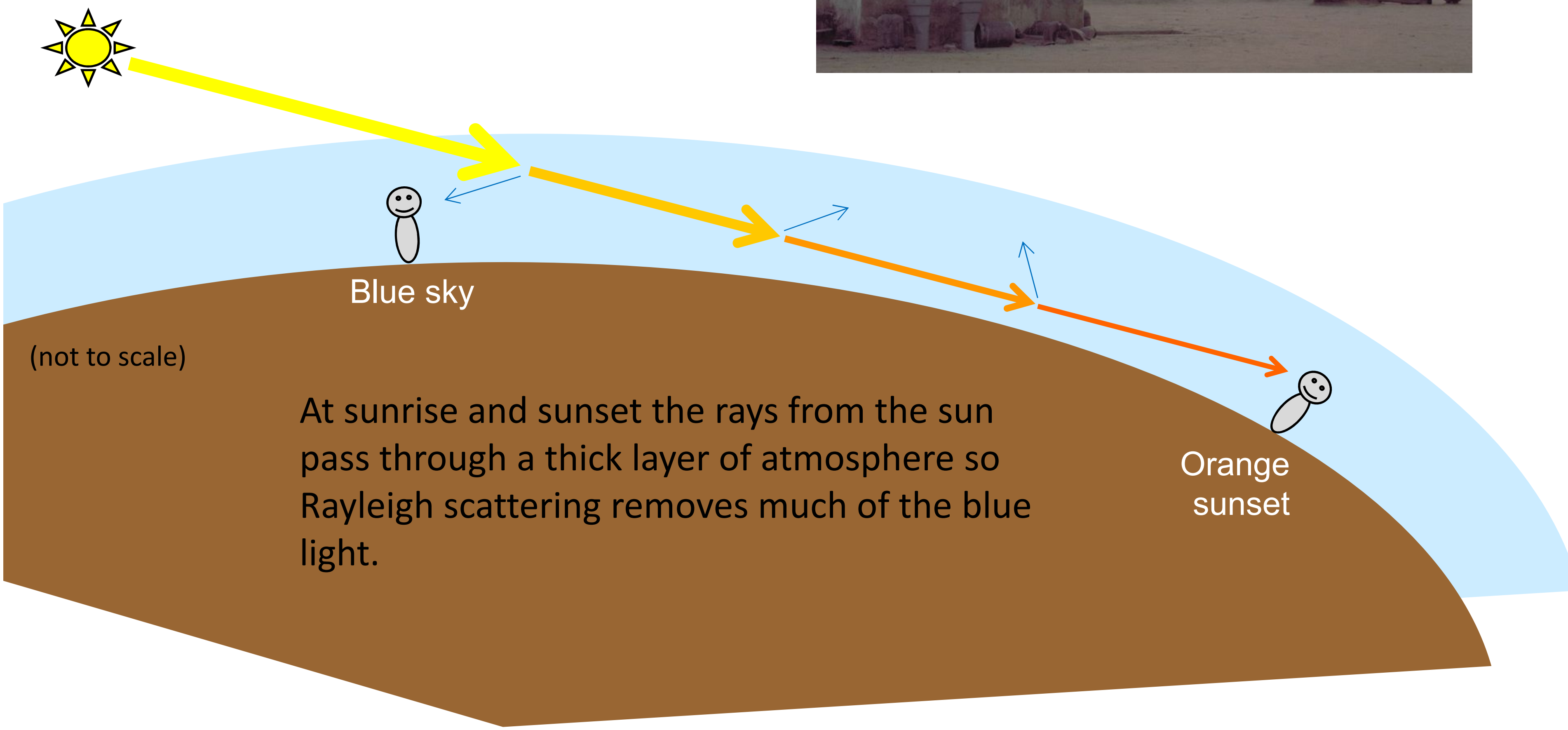
Atmospheric Perspective



Far away mountains have a bluish tint due to blue light scattered in by Rayleigh scattering



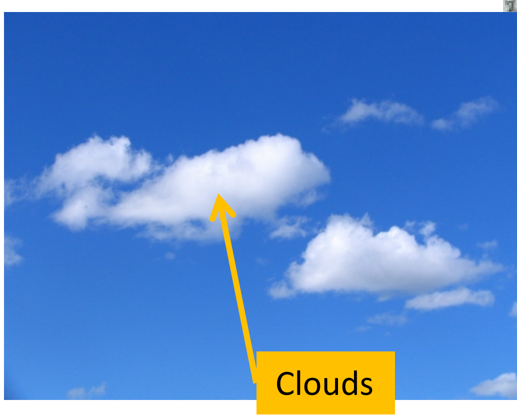
Sunrise & Sunset

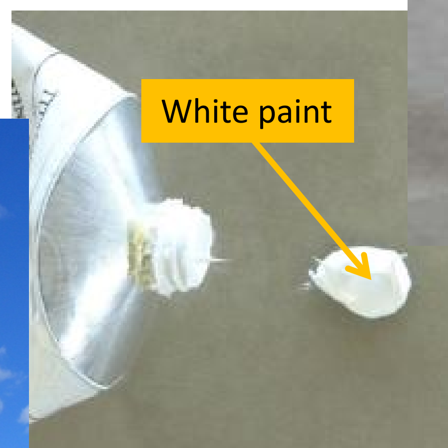


At sunrise and sunset the rays from the sun pass through a thick layer of atmosphere so Rayleigh scattering removes much of the blue light.

Mie Scattering

Scattering by small particles varies with size so the result averages out to white.



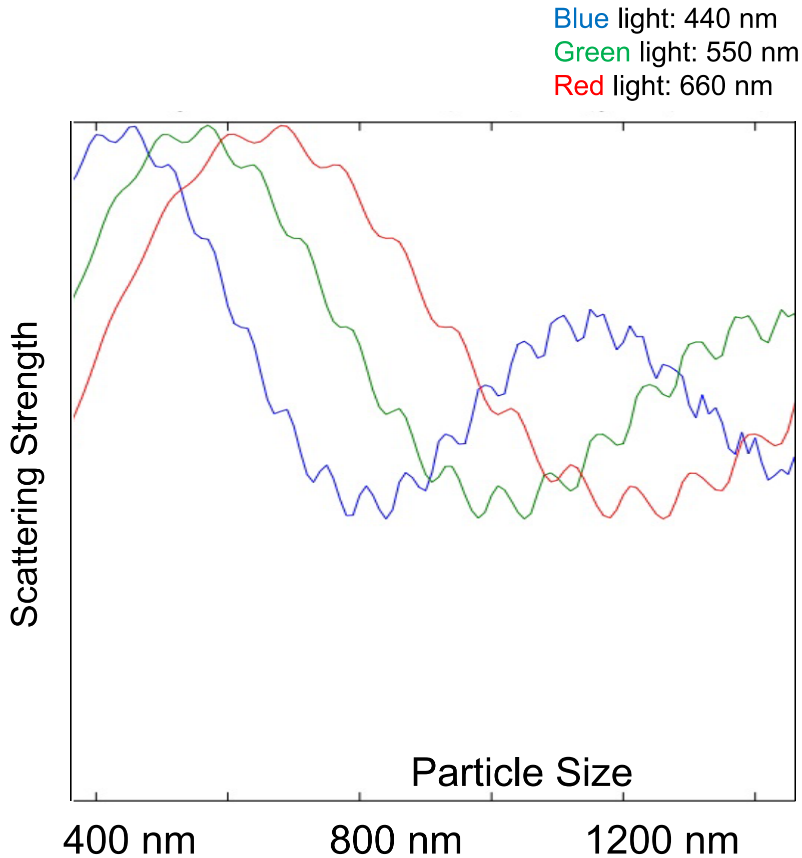


Particles comparable to wavelength of visible light.

Mie Scattering & Color

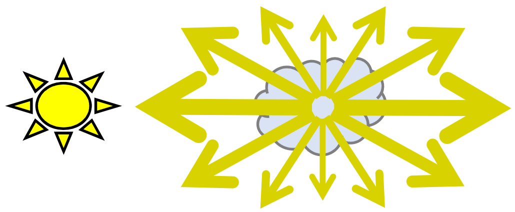
For small particles Mie scattering strength varies greatly with the particle size.

Since particles tend to be a mix of sizes, all hues are scattered equally resulting in white.



Scattering Directions

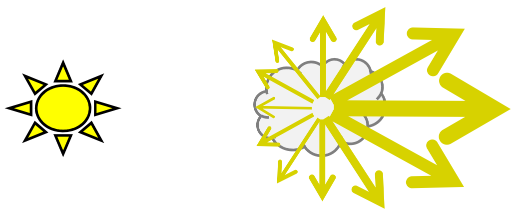
Rayleigh Scattering



Forward & Backward

Thin
Smok
e

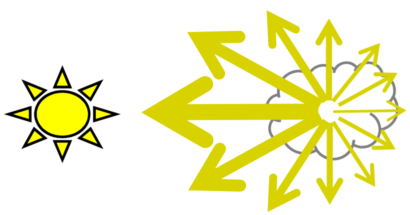
Mie Scattering



Forward

Light Fog

Reflection



Backward

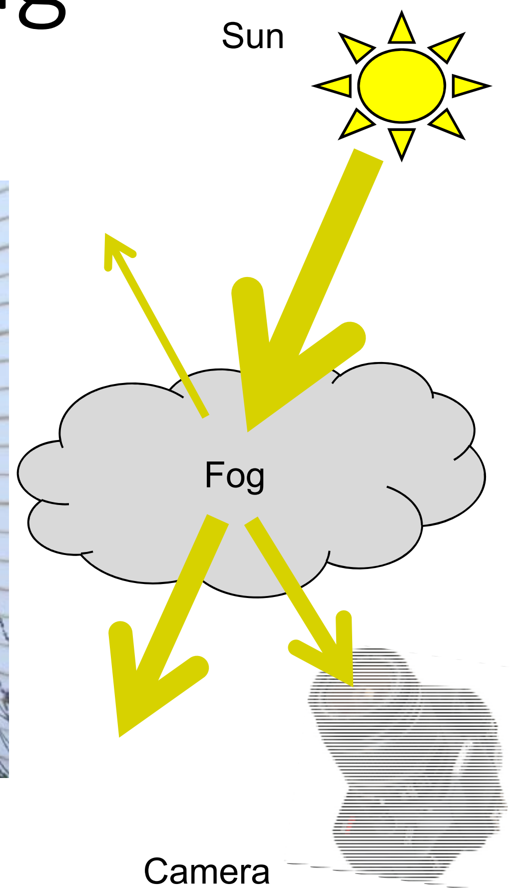
Snow

Mie Forward Scattering

Sun is behind the fog in this photo

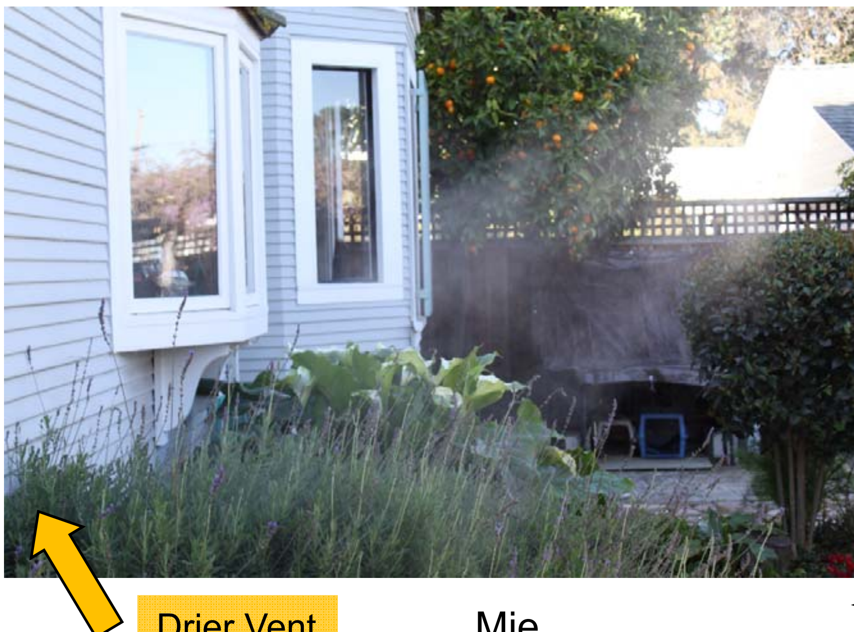
Mie
scattering
from drier fog

Drier Vent



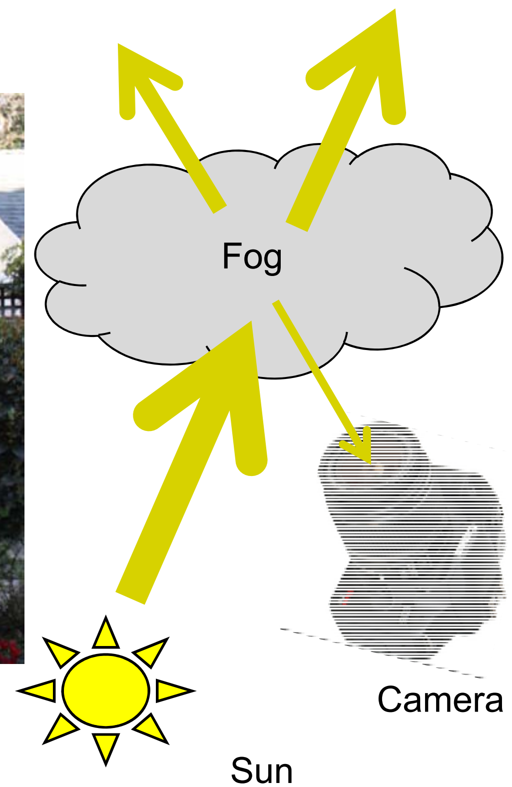
Mie Backward Scattering

Sun is behind the camera in this photo



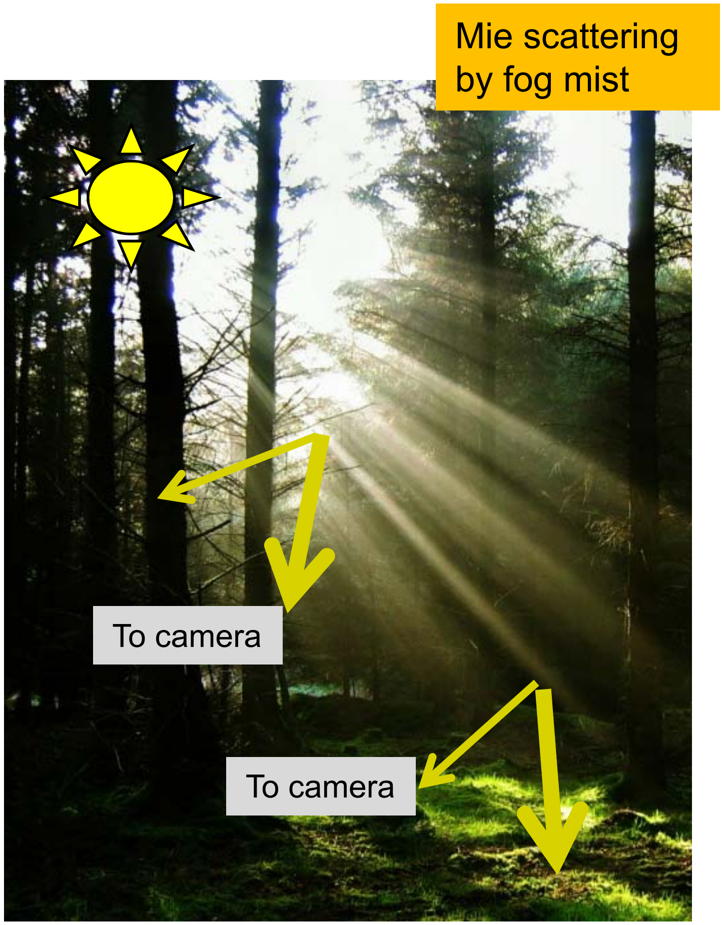
Drier Vent

Mie
scattering
from drier fog



Sun Rays

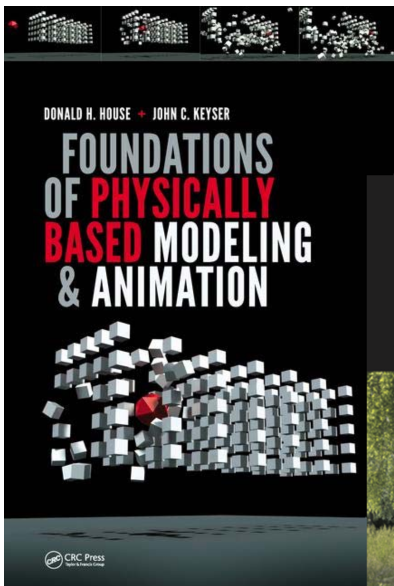
Intensity of sun rays varies with the angle between sun and camera.



Notice that the light on the ground is bright even though the ray's intensity appears to taper off along the sun ray.

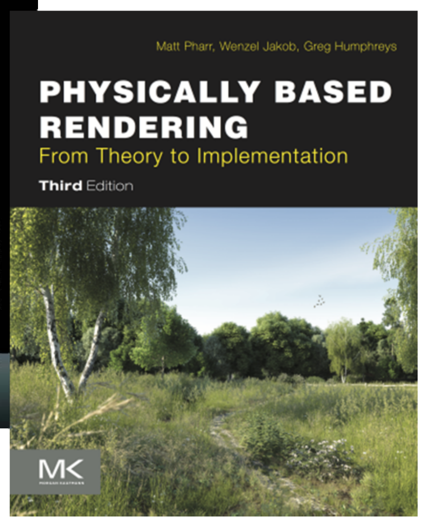
Try this at home!

Books



Mechanics

Optics



Game Engines

UNREAL ENGINE

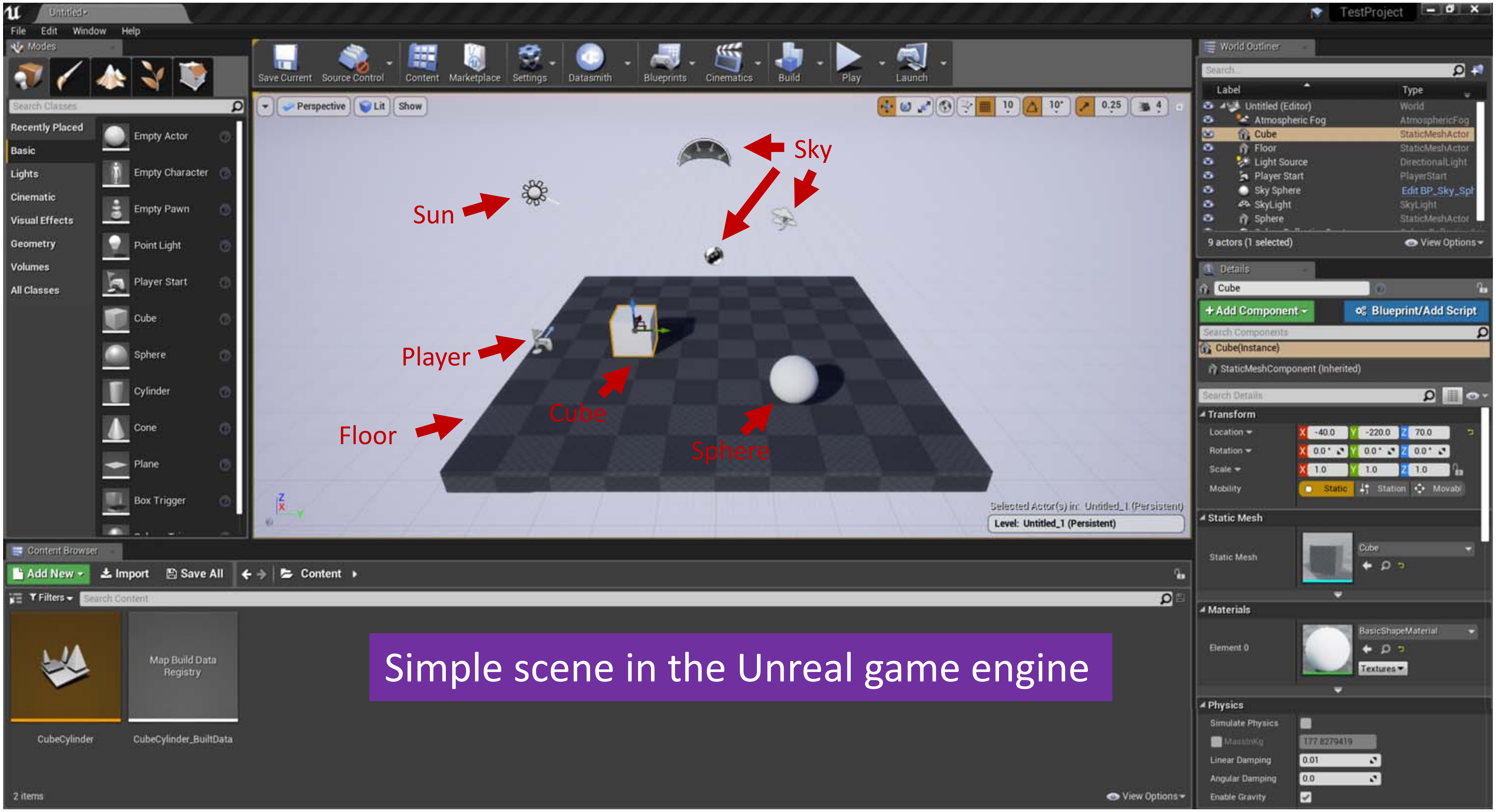
unity

Software is free for educational use

3D Modeling / Animation

AUTODESK® MAYA®

blender®



Simple scene in the Unreal game engine

Thank you!

Alejandro.Garcia@sjsu.edu

www.algarcia.org

www.youtube.com/user/AlejandroLuisGarcia/