

# Reflection

---

Video Game Graphics AD-011

# Reflection algorithms

- Cubemap reflections
- Screen space reflections
- Planar Reflections
- Ray traced reflections

# Cubemap reflections

Simplest and fastest algorithm. Pre-rendered reflection is stored in cubemap texture. Commonly used for complex curved surfaces. Currently most popular approach among game developers.



# Cubemap reflections pros & cons

## Pros

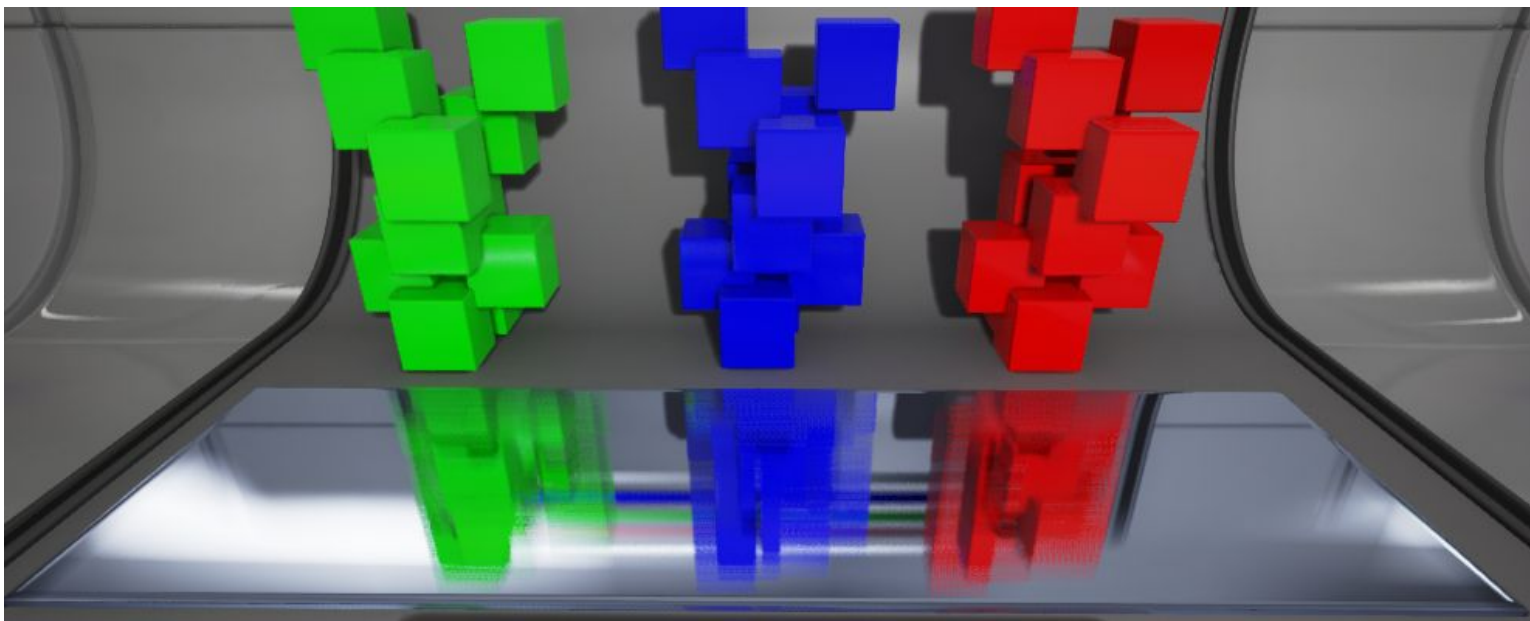
- Fast
- Works good with curved reflective surfaces
- Reflections can be handcrafted by artists
- Cubemap mipmaps can be used for rough material simulation

## Cons

- Reflection is usually static
- Doesn't work for moving objects

# Screen space reflections

Technique for reusing screen data to calculate reflections. It is commonly used to create more subtle reflections such as on wet floor surfaces or in puddles.



# Screen space reflections pros & cons

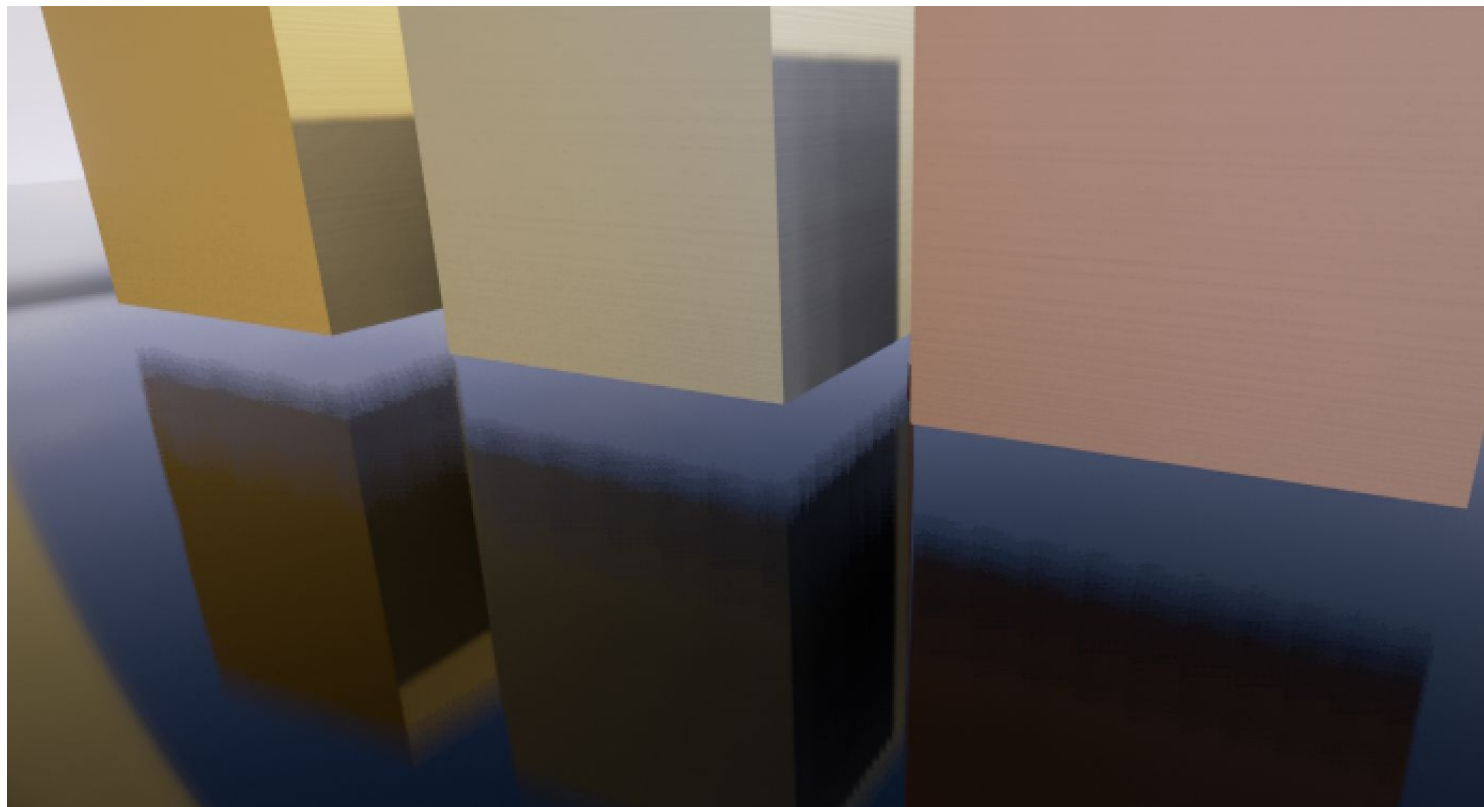
## Pros

- Relatively fast as scene is rendered only once
- Realtime and good quality

## Cons

- Don't reflect objects that are out of camera view
- Requires deferred rendering pipeline, as it relies on normal buffer

# Screen space reflection artifacts



# Planar reflections

Whole scene is rendered once again into the offscreen reflection buffer, later this buffer is used to render the reflective surface.





# Planar reflections pros & cons

## Pros

- Realtime and good quality
- Easy to implement and can be integrated with any kind of pipeline

## Cons

- Slow, as scene is rendered multiple times
- Can be used only on flat reflective surfaces
- Often optimized by rendering scene partially and with lower resolution

# Planar reflection artifacts



# Ray-traced reflections

Simulation of real light physics. Reverse tracing of photon rays from camera to reflective surface and bounces.



# Ray-traced reflections pros & cons

## Pros

- Very accurate realistic reflections
- Can be used with any kind of surfaces
- Easy to implement

## Cons

- Too slow for modern hardware, probably will be used with next generation of GPUs