Armory

Real-time rendering in Blender

- Cycles workflow
- Reflect subset of Blender API
- Independent core

No Blender forking

- Addon + player patch, no fragmentation

No Cycles running in real-time

- Trade quality for speed

No viewport

- Custom renderer

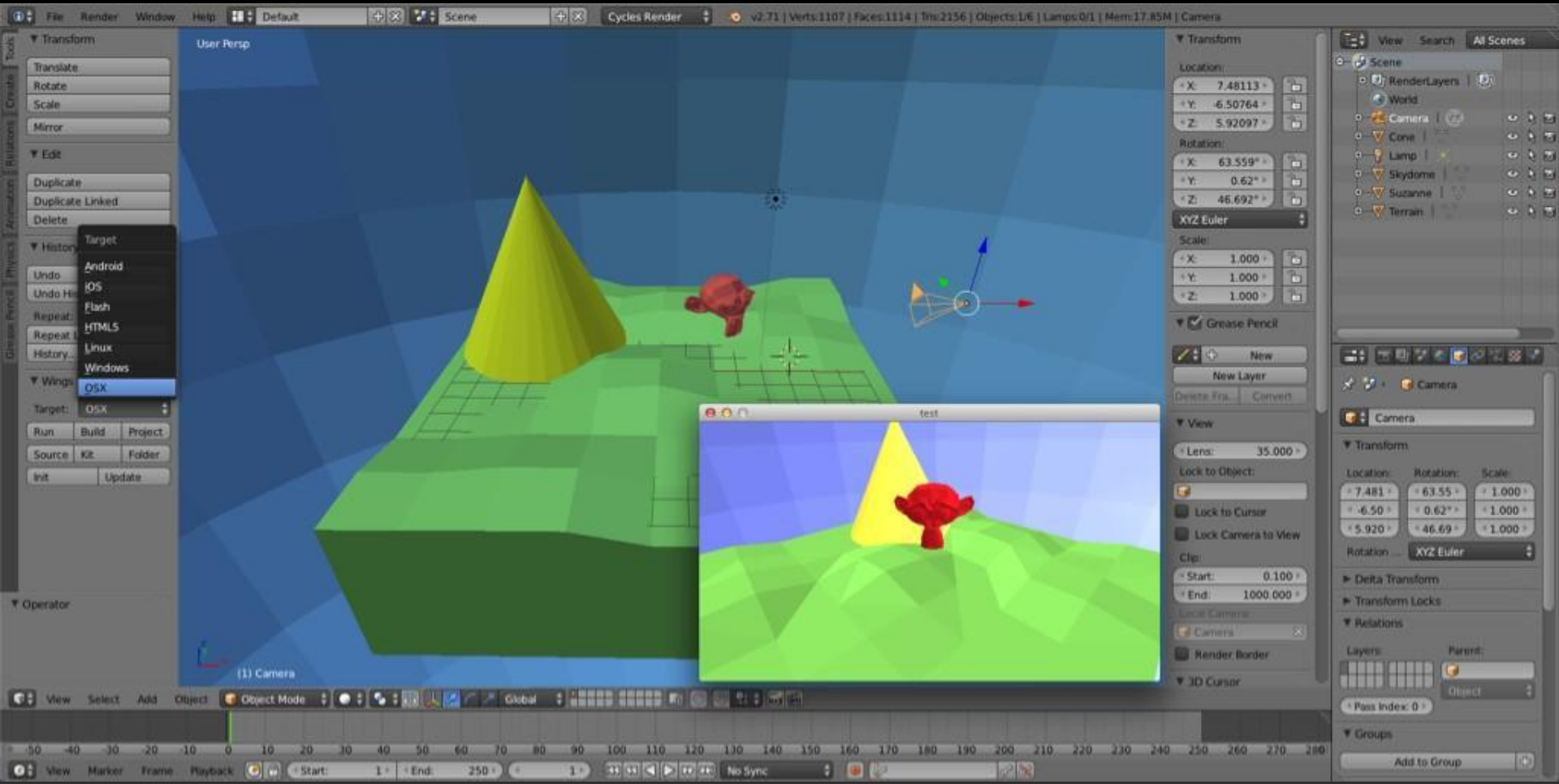
Building a 3D Engine

Dumbest idea ever!

Needs to be a lot better..

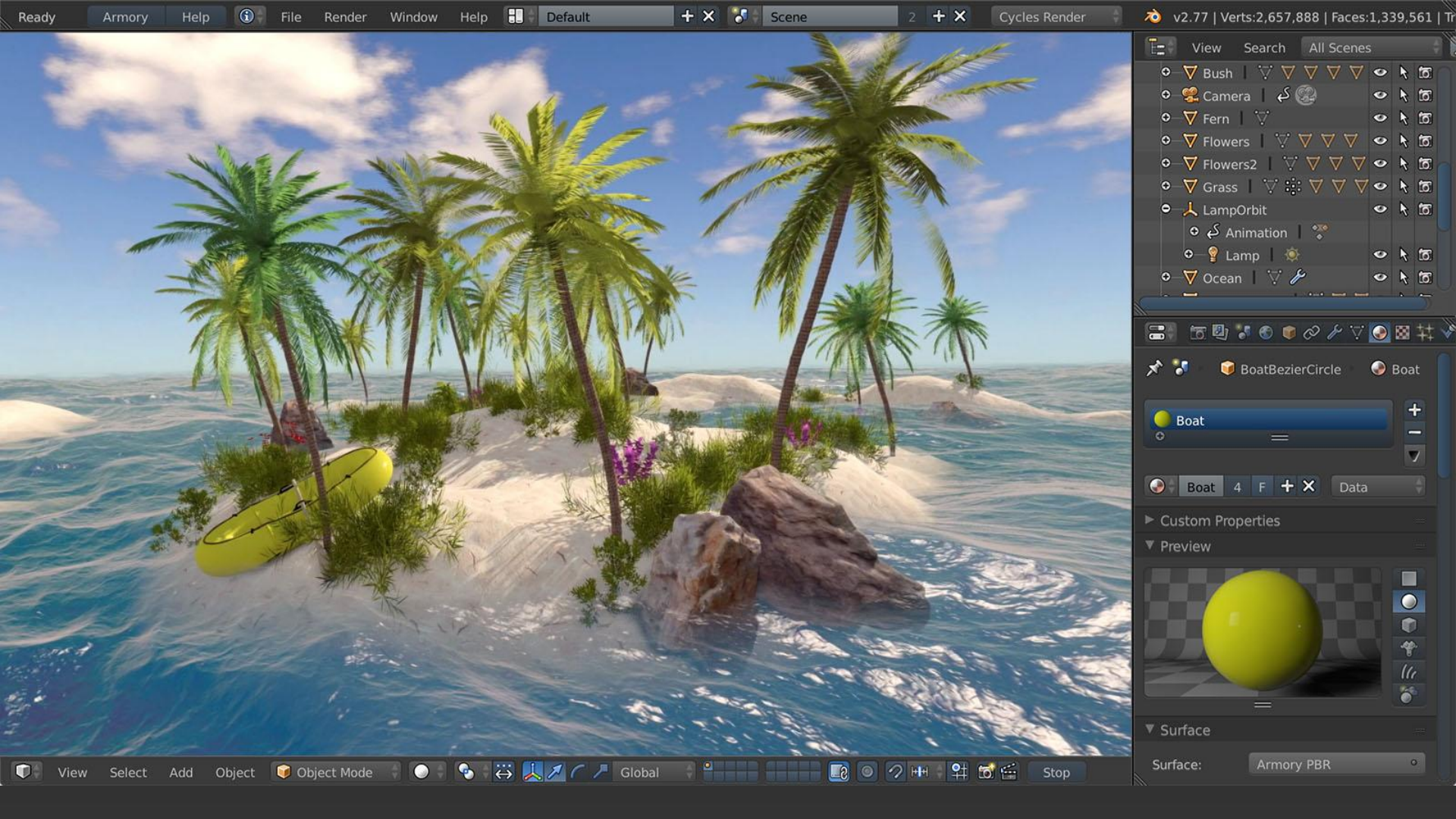
1

Editing


File Render Window Help Default Scene Cycles Render
v2.71 | Verts:1107 | Faces:1114 | Tris:2156 | Objects:1/6 | Lamps:0/1 | Mem:17.85M | Camera
Transform
Translate
Rotate
Scale
Mirror
Edit
Duplicate
Duplicate Linked
Delete
Target
Android
iOS
Flash
HTML5
Linux
Windows
OSX
Target: OSX
Run Build Project
Source Kit Folder
Init Update
Operator
User Persp
(1) Camera
test
Transform
Location:
X: 7.48113
Y: -6.50764
Z: 5.92097
Rotation:
X: 63.559°
Y: 0.62°
Z: 46.692°
XYZ Euler
Scale:
X: 1.000
Y: 1.000
Z: 1.000
Grease Pencil
New
New Layer
View
Lens: 35.000
Lock to Object:
Lock to Cursor
Lock Camera to View
Clip:
Start: 0.100
End: 1000.000
Render Border
3D Cursor
View Search All Scenes
Scene
RenderLayers
World
Camera
Cone
Lamp
Skydome
Suzanne
Terrain
Camera
Transform
Location: Rotation: Scale:
7.481 63.55 1.000
-6.50 0.62° 1.000
5.920 46.69 1.000
Rotation ... XYZ Euler
Delta Transform
Transform Locks
Relations
Layers: Parent:
Pass Index: 0
Groups
Add to Group
View Select Add Object Object Mode Global
View Marker Frame Playback Start: 1 End: 250 1 No Sync

August 2014

1000's hours of frustration and joy later..


Ready
Armory
Help
File
Render
Window
Help
Default
Scene
Cycles Render
v2.77 | Verts:2,657,888 | Faces:1,339,561
View
Search
All Scenes
Bush
Camera
Fern
Flowers
Flowers2
Grass
LampOrbit
Animation
Lamp
Ocean
BoatBezierCircle
Boat
Boat
Boat
4
F
Data
Custom Properties
Preview
Surface
Surface:
Armory PBR
View
Select
Add
Object
Object Mode
Global
Stop

Unified workflow

No exporting trouble

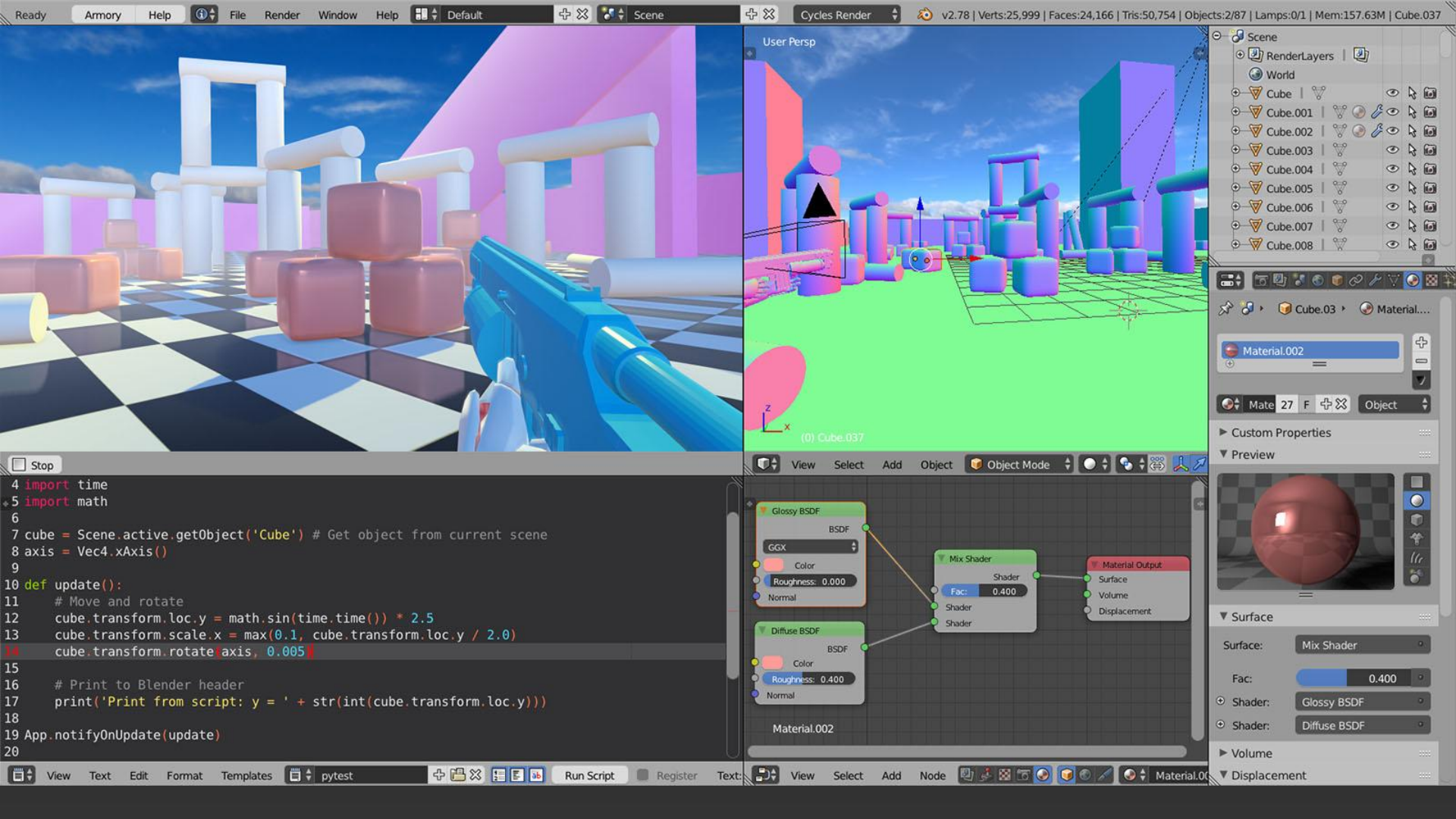
Blender capabilities + ecosystem

Low learning curve - same UI

Player synchronization

Translate operators

Stream scene changes

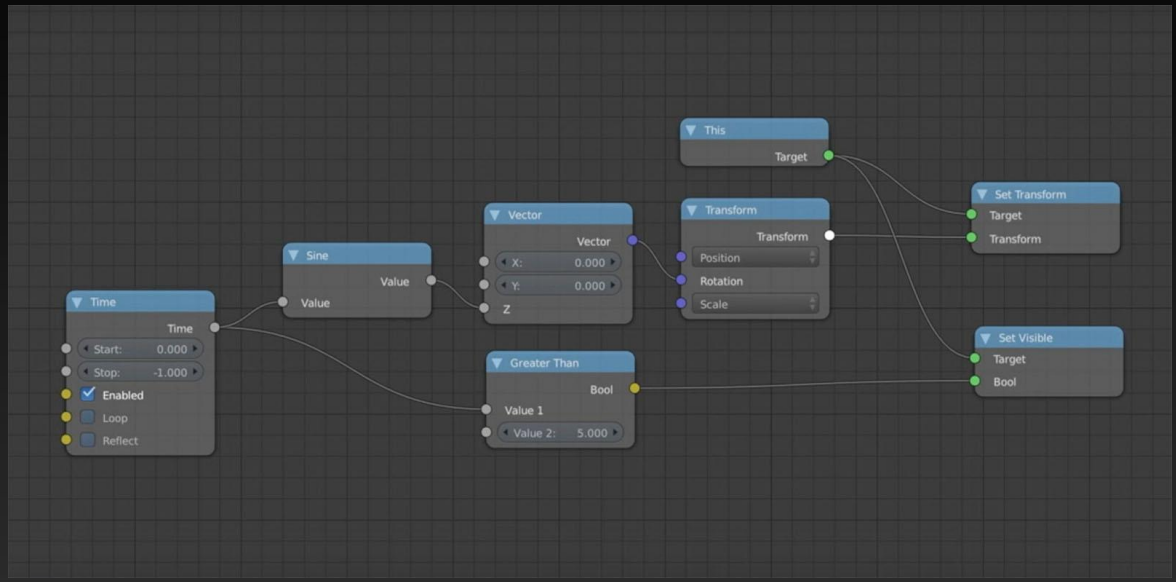

Ready
Armory
Help
File
Render
Window
Help
Default
Scene
Cycles Render
v2.78 | Verts:25,999 | Faces:24,166 | Tris:50,754 | Objects:2/87 | Lamps:0/1 | Mem:157.63M | Cube.037
User Persp
(0) Cube.037
Scene
RenderLayers
World
Cube
Cube.001
Cube.002
Cube.003
Cube.004
Cube.005
Cube.006
Cube.007
Cube.008
Cube.03
Material....
Material.002
Mate
27
F
Object
Custom Properties
Preview
Surface
Surface:
Mix Shader
Fac:
0.400
Shader:
Glossy BSDF
Shader:
Diffuse BSDF
Volume
Displacement
Stop
import time
import math
cube = Scene.active.getObject('Cube') # Get object from current scene
axis = Vec4.xAxis()
def update():
# Move and rotate
cube.transform.loc.y = math.sin(time.time()) * 2.5
cube.transform.scale.x = max(0.1, cube.transform.loc.y / 2.0)
cube.transform.rotate(axis, 0.005)
# Print to Blender header
print('Print from script: y = ' + str(int(cube.transform.loc.y)))
App.notifyOnUpdate(update)
View
Text
Edit
Format
Templates
pytest
Run Script
Register
Text:
View
Select
Add
Object
Object Mode
Glossy BSDF
BSDF
GGX
Color
Roughness: 0.000
Normal
Diffuse BSDF
BSDF
Color
Roughness: 0.400
Normal
Mix Shader
Shader
Fac: 0.400
Shader
Shader
Material Output
Surface
Volume
Displacement
Material.002
View
Select
Add
Node
Material.00


Armory
Help
File
Render
Window
Help
Back to Previous
Cycles Render
v2.78 | Verts:22,362 | Faces:43,317 | Tris:43,317 | Objects:1/6 | Lamps:0/1 | Mem:385.20M | Body
Stop
Pause

Make it interactive

Haxe, Python, JS

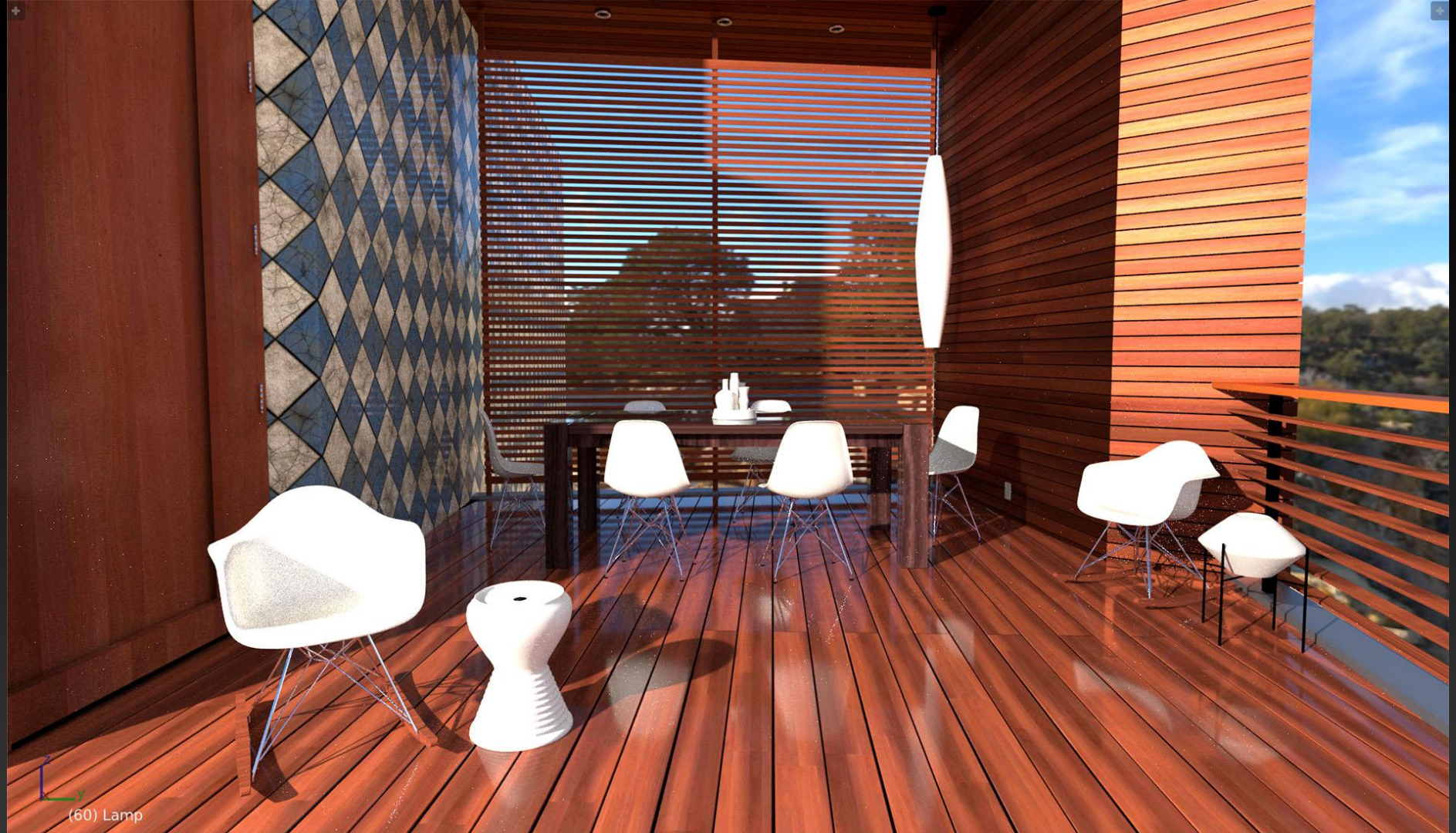
Logic nodes



Comparing output to Cycles

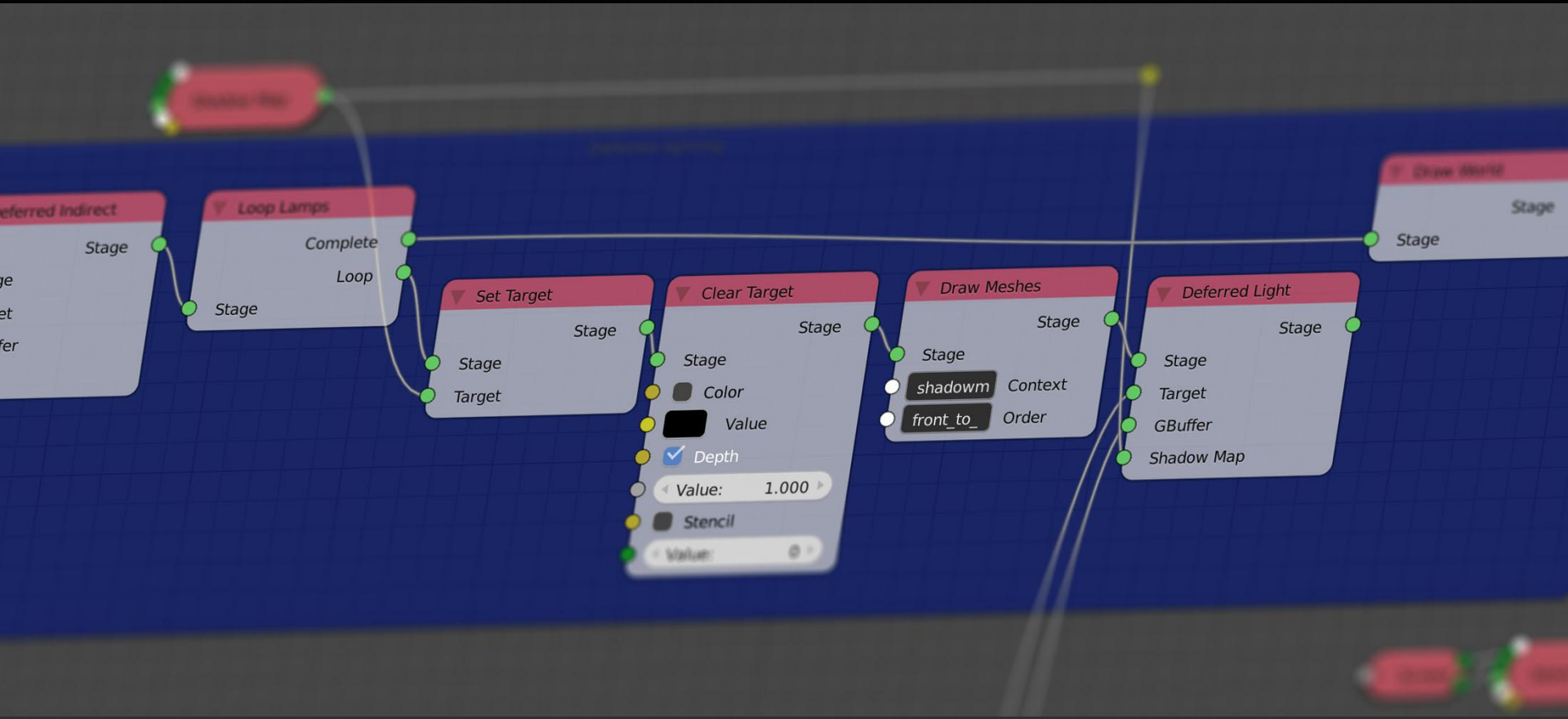
- Improve renderer
- Higher quality output

Goal - render any Cycles scene


(60) Lamp

2

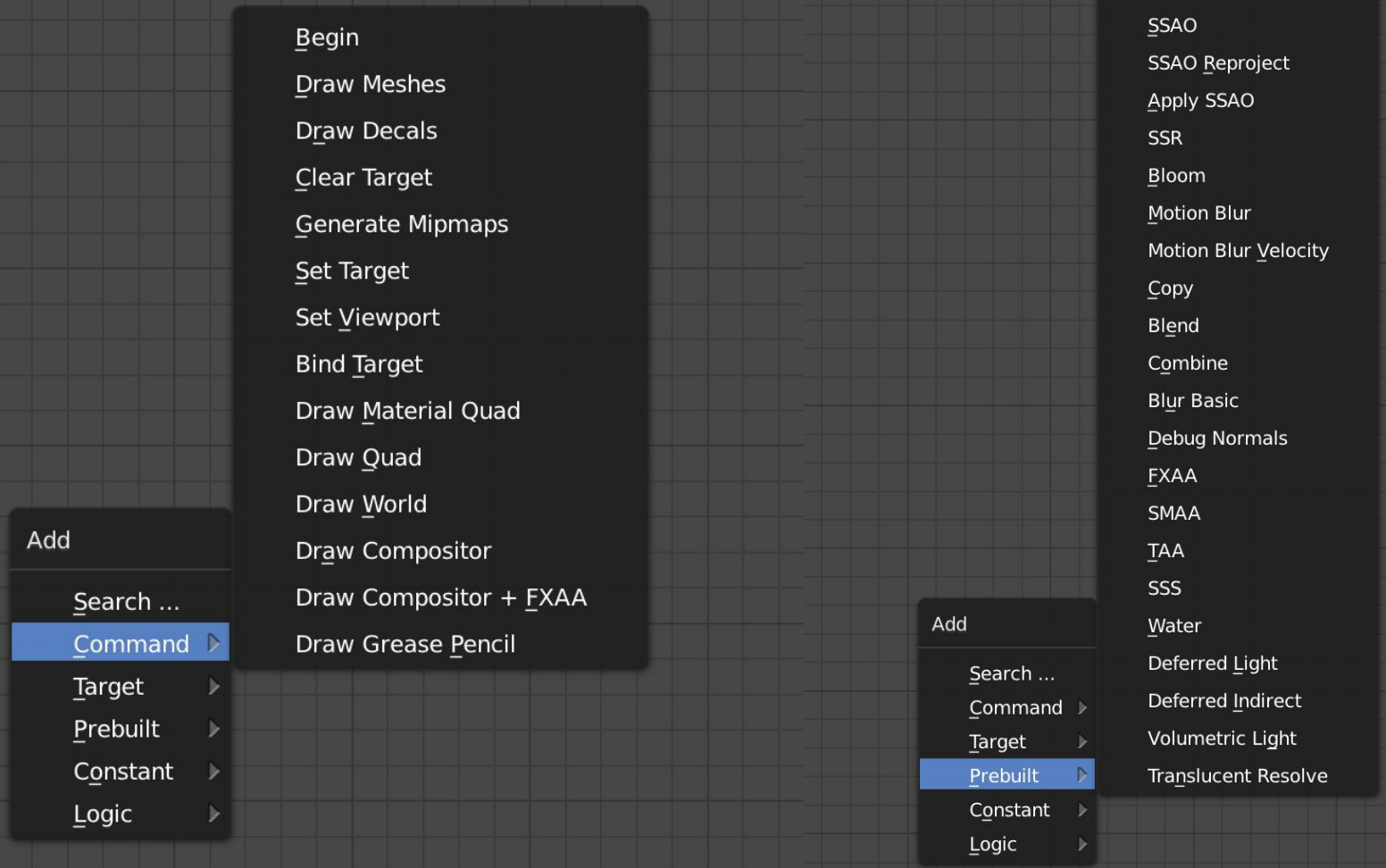
Rendering

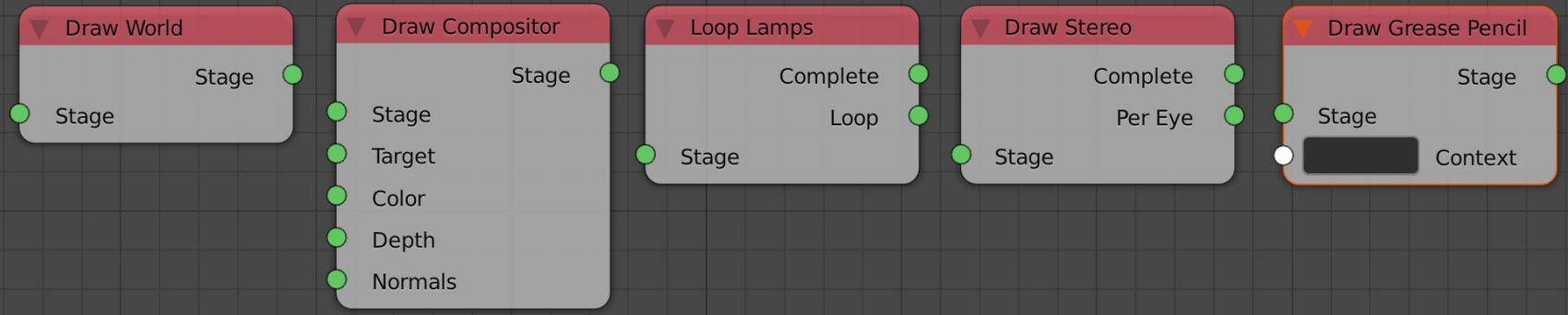

eferred Indirect
Stage
Loop Lamps
Complete
Loop
Stage
Set Target
Stage
Stage
Target
Clear Target
Stage
Stage
Color
Value
Depth
Value: 1.000
Stencil
Value:
Draw Meshes
Stage
Stage
shadowm
Context
front_to_
Order
Deferred Light
Stage
Stage
Target
GBuffer
Shadow Map
Stage
Stage

Renderer written from scratch

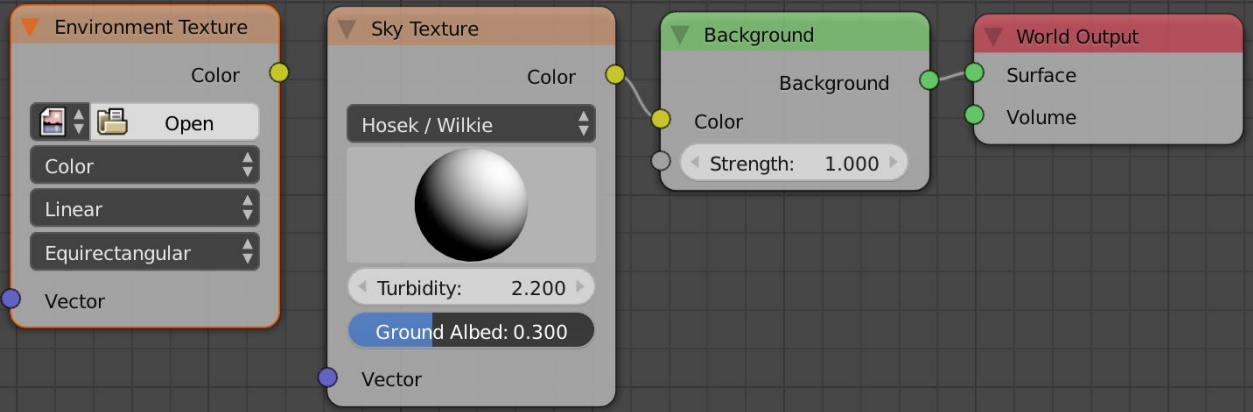
- Modern
- Minimal - removed code = good code
- Follow the needs

“Simplicity is prerequisite for reliability”

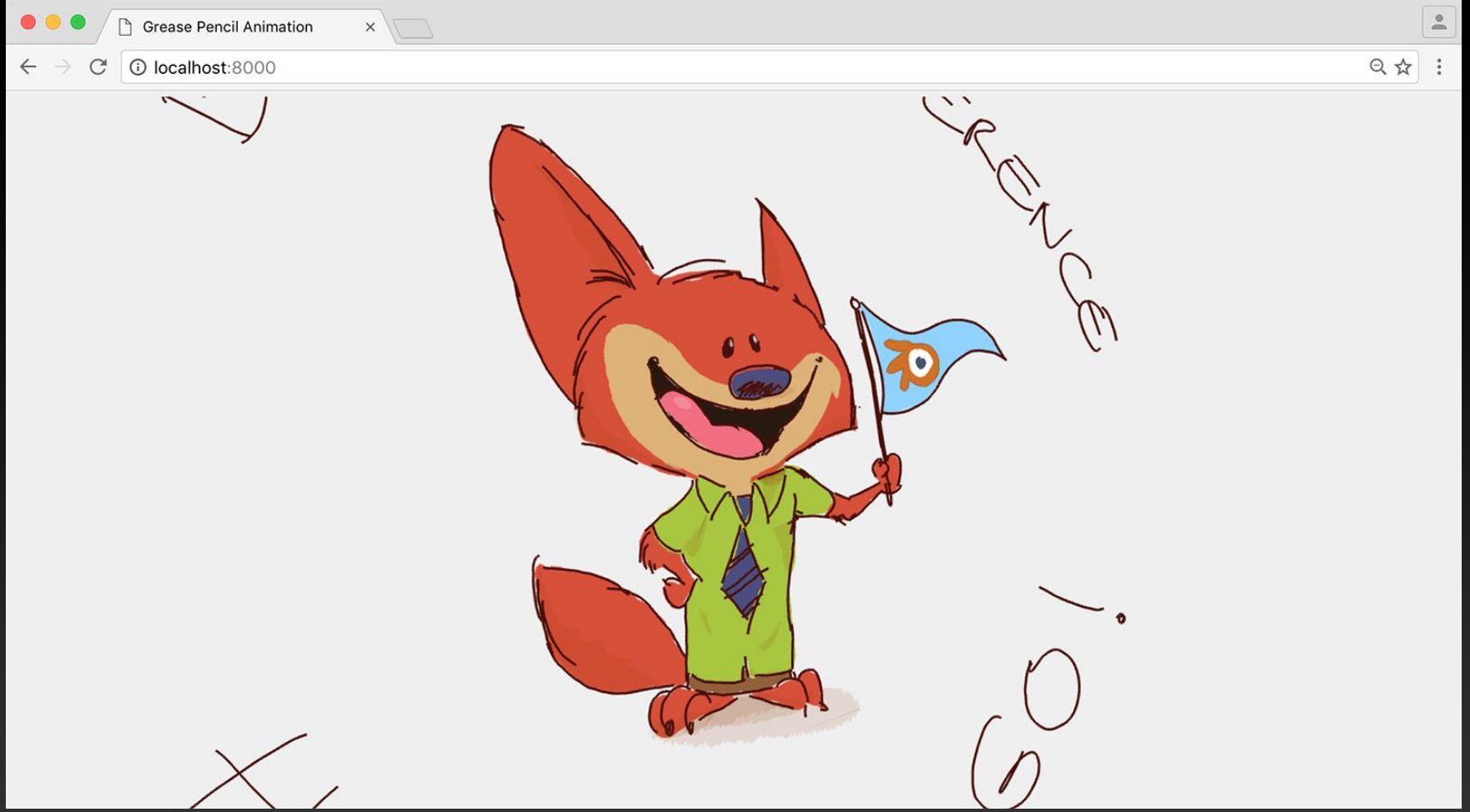

Add
Search ...
Command
Target
Prebuilt
Constant
Logic
Begin
Draw Meshes
Draw Decals
Clear Target
Generate Mipmaps
Set Target
Set Viewport
Bind Target
Draw Material Quad
Draw Quad
Draw World
Draw Compositor
Draw Compositor + FXAA
Draw Grease Pencil
Add
Search ...
Command
Target
Prebuilt
Constant
Logic
Quad Pass
SSAO
SSAO Reproject
Apply SSAO
SSR
Bloom
Motion Blur
Motion Blur Velocity
Copy
Blend
Combine
Blur Basic
Debug Normals
FXAA
SMAA
TAA
SSS
Water
Deferred Light
Deferred Indirect
Volumetric Light
Translucent Resolve



Reference world, compositor & lamp nodes



Hosek / Wilkie sky, auto-generated prefiltered maps



Grease Pencil animation by Daniel M. Lara

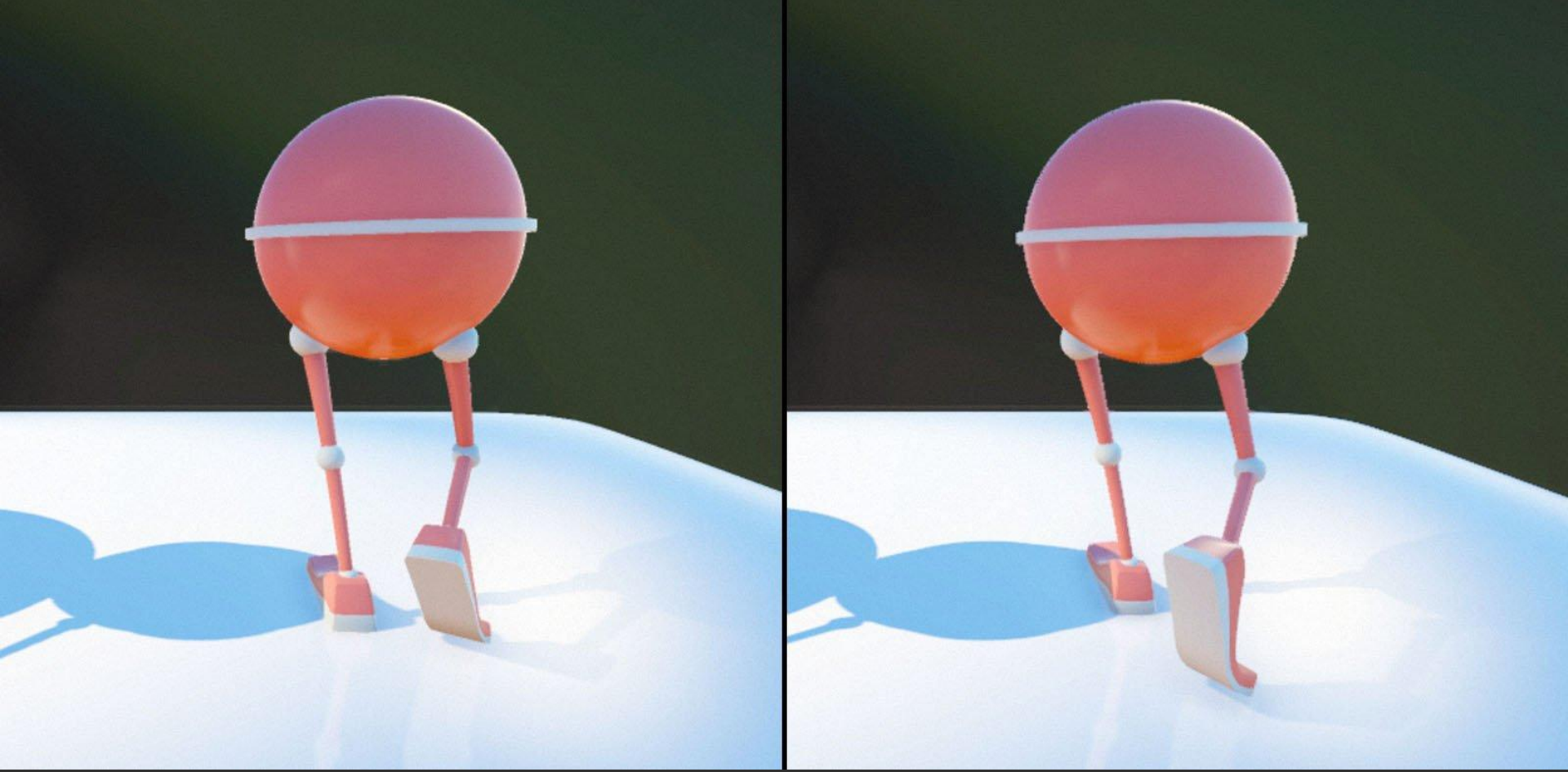
Render path nodes

Execute code - dynamic resolution scaling

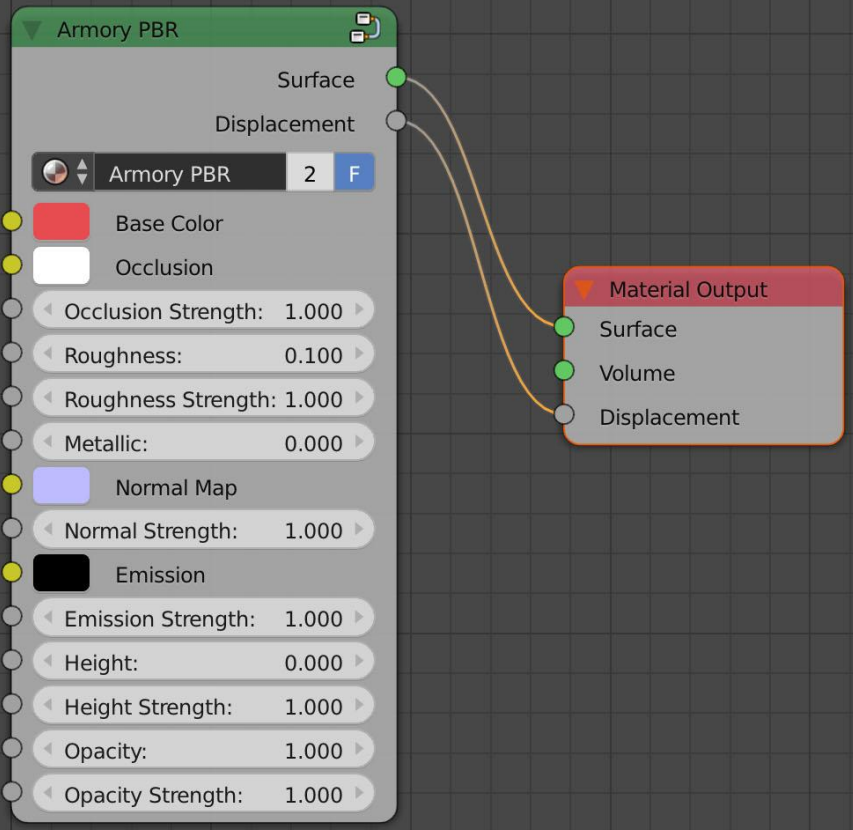
Great place to prototype 3D rendering techniques

Great place to learn graphics pipeline (live render path)

Jump to path tracing



Deferred vs forward with thin gbuffer



Standard PBR setup

Import assets

Volumetric lighting

Instanced rendering

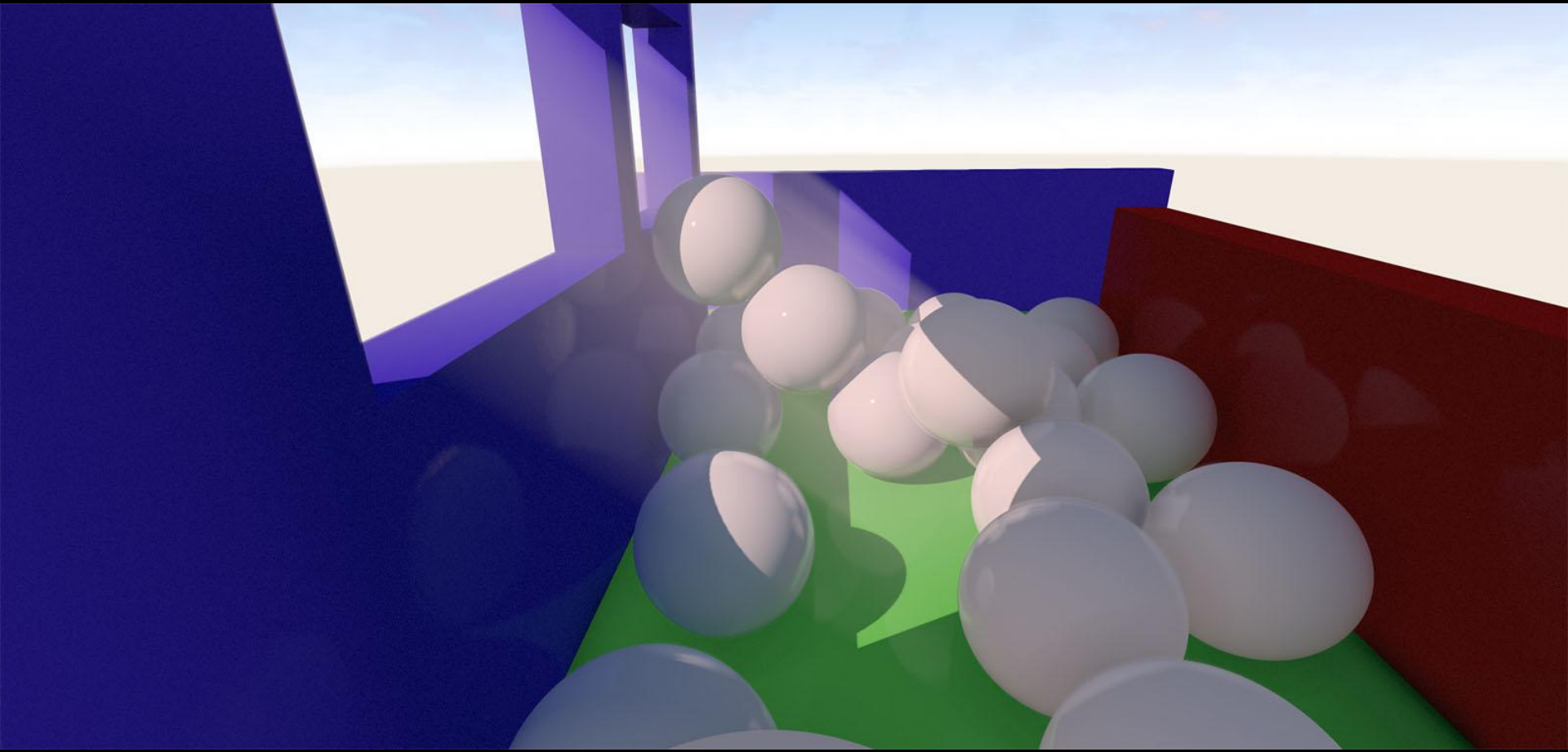
Tessellated displacement

Hotel room by Pietro di Chito

Global Illumination

Offline solution

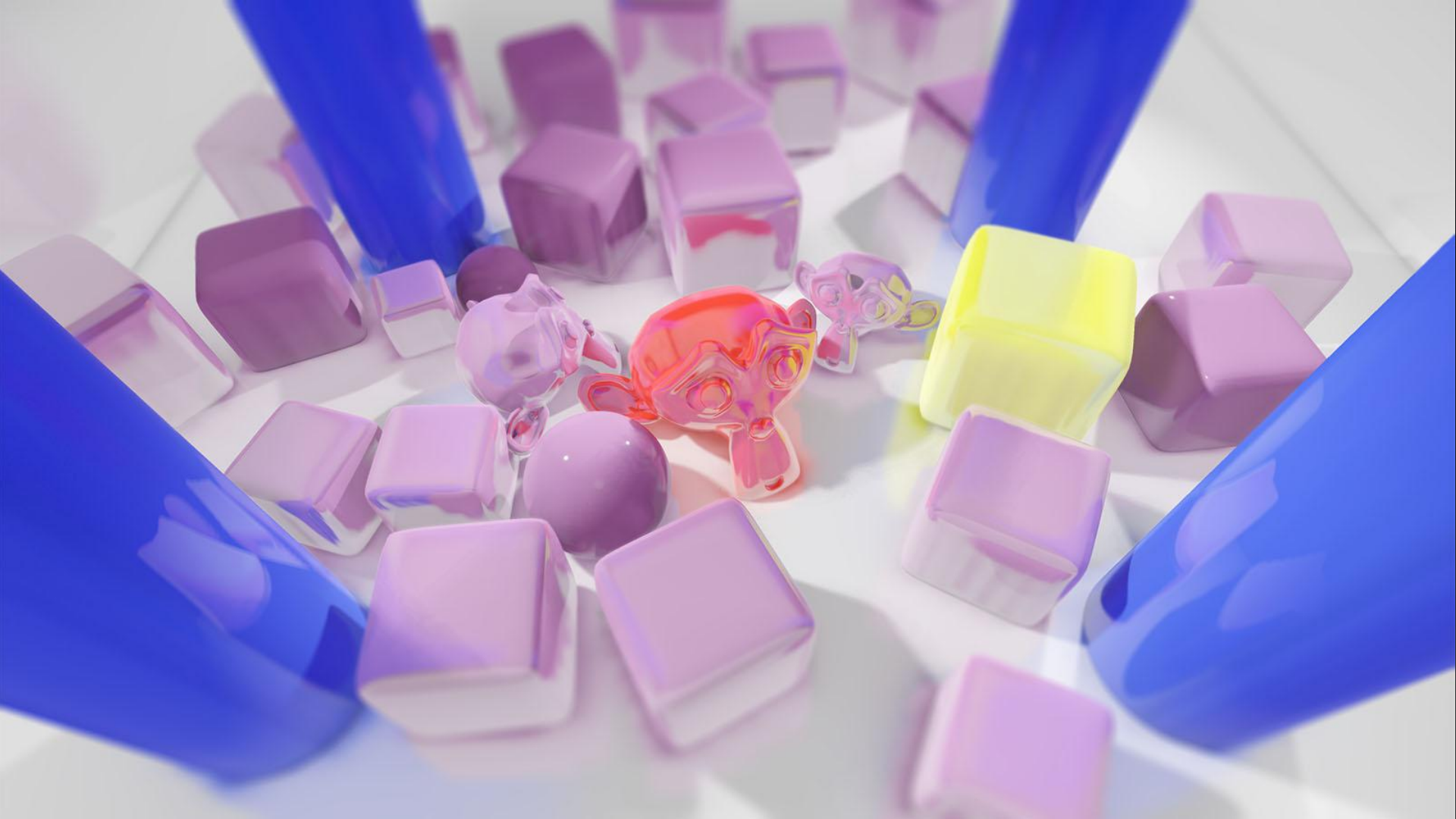
- Cycles rendered probes
- Go performance!



Online solution

Voxel cone tracing

Go dynamic!



3

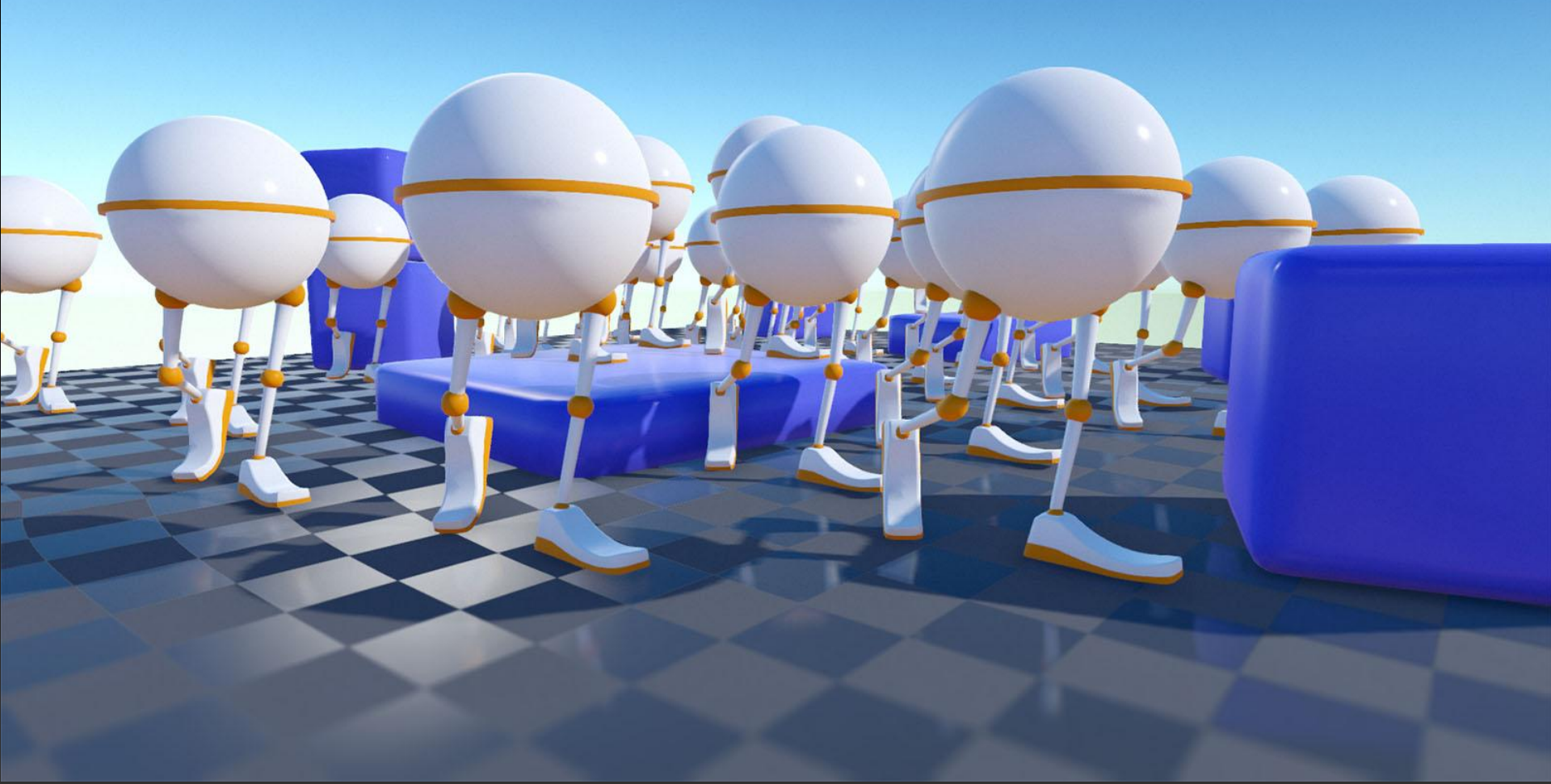
Deploying

Kha - dream come true!

- Hardware abstraction
- No dependency hell
- No bloat
- No (graphics) API specifics
- No recompilation

Unmatched portability

- Any language - *Haxe*
- Any graphics API - *Kha*
- Any render technique - *Armory*



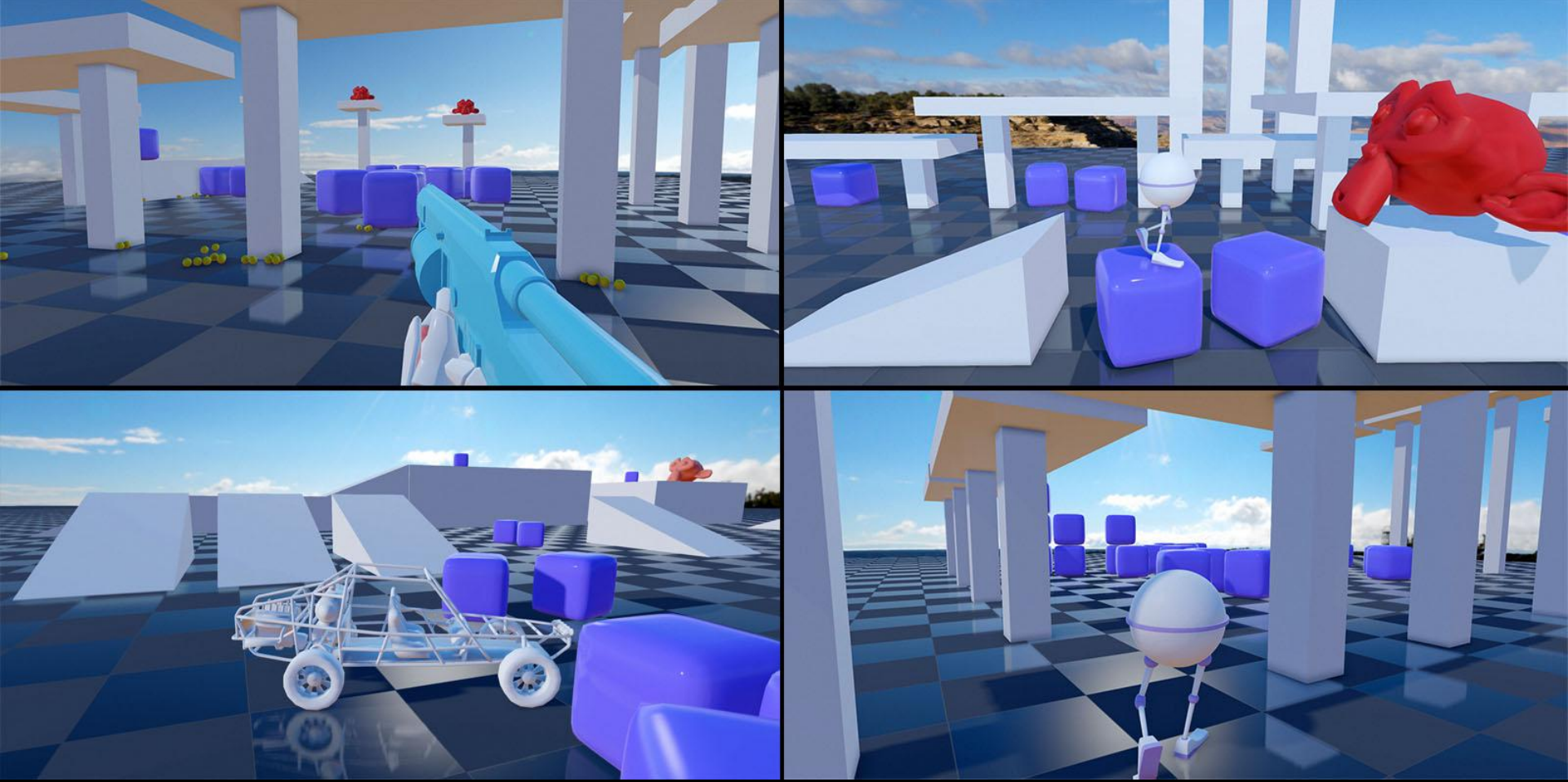
Runs everywhere, fits in 996KB (+ instanced skinning!)

0

Release

Preview ‘zero’ out today

- Many limitations
- Minimal docs
- May not work at all!
- Paid for now - looking for ways to go free



Clunky examples

Thank you

No Blender & Kha - no Armory

Go open source!

armory3d.org