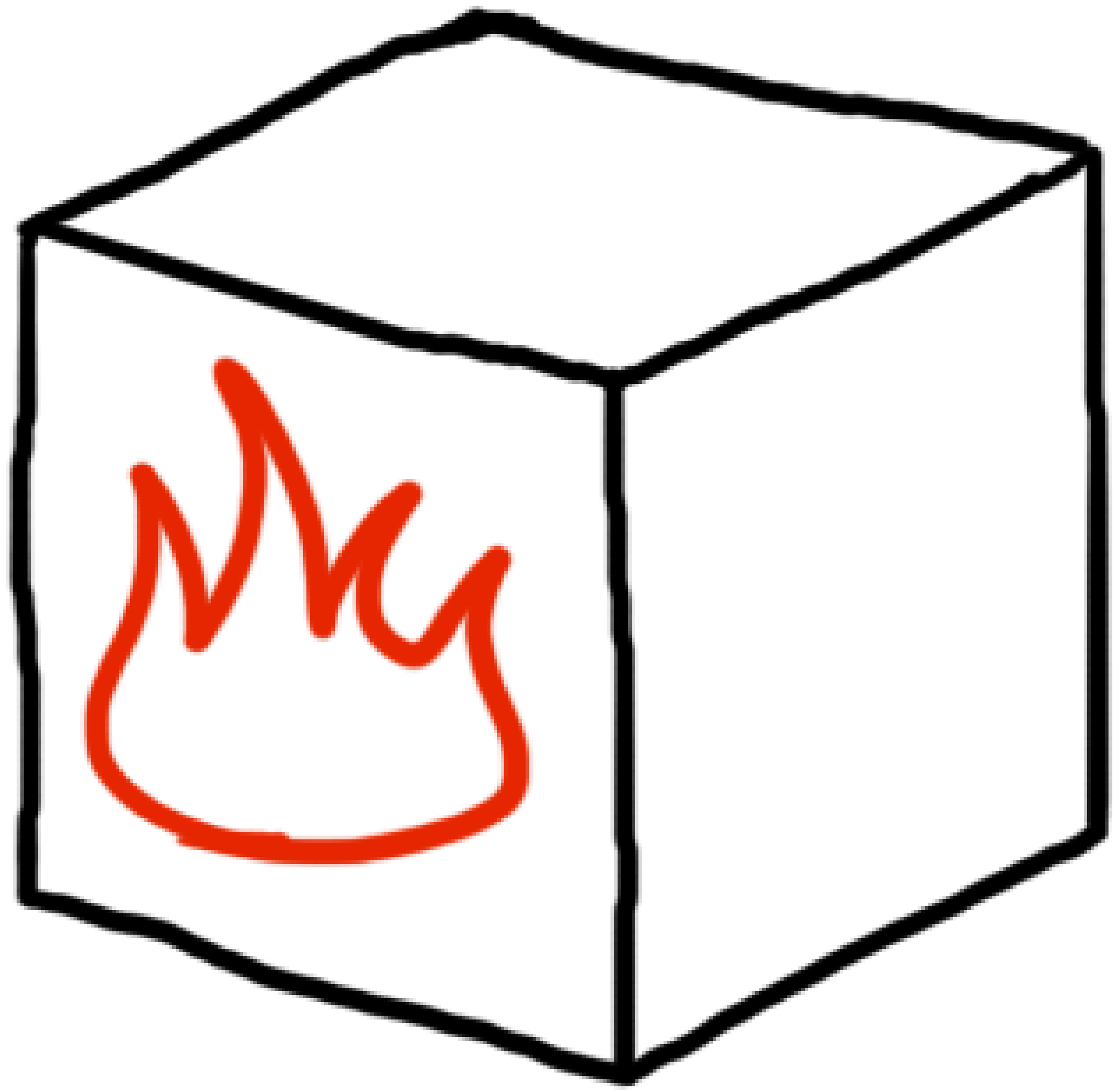
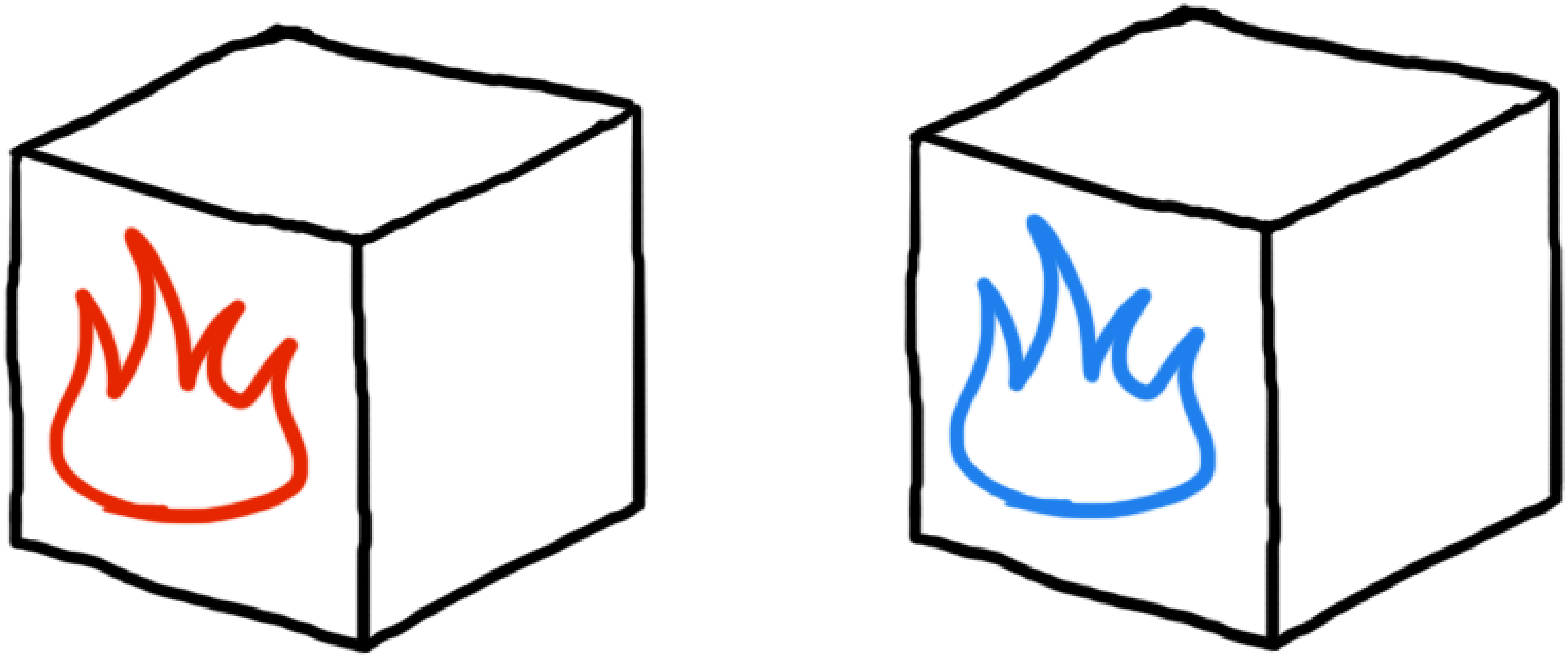
How to Build Complex VFX Systems With Simple Controls

Anastasia Sopikova
Senior VFX Artist @ Creative Assembly

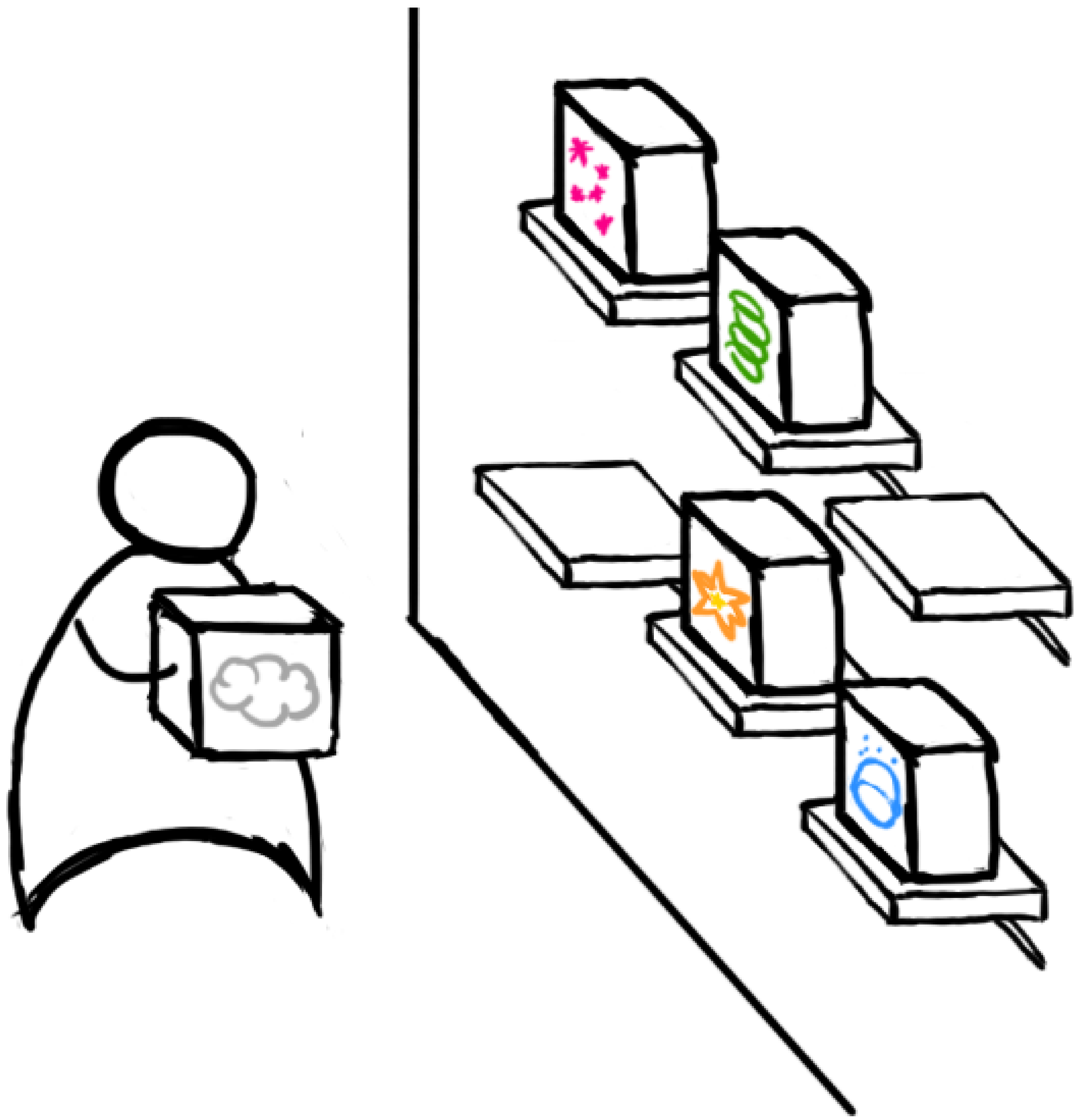
GOOD OLD DAYS



GOOD (?) OLD DAYS

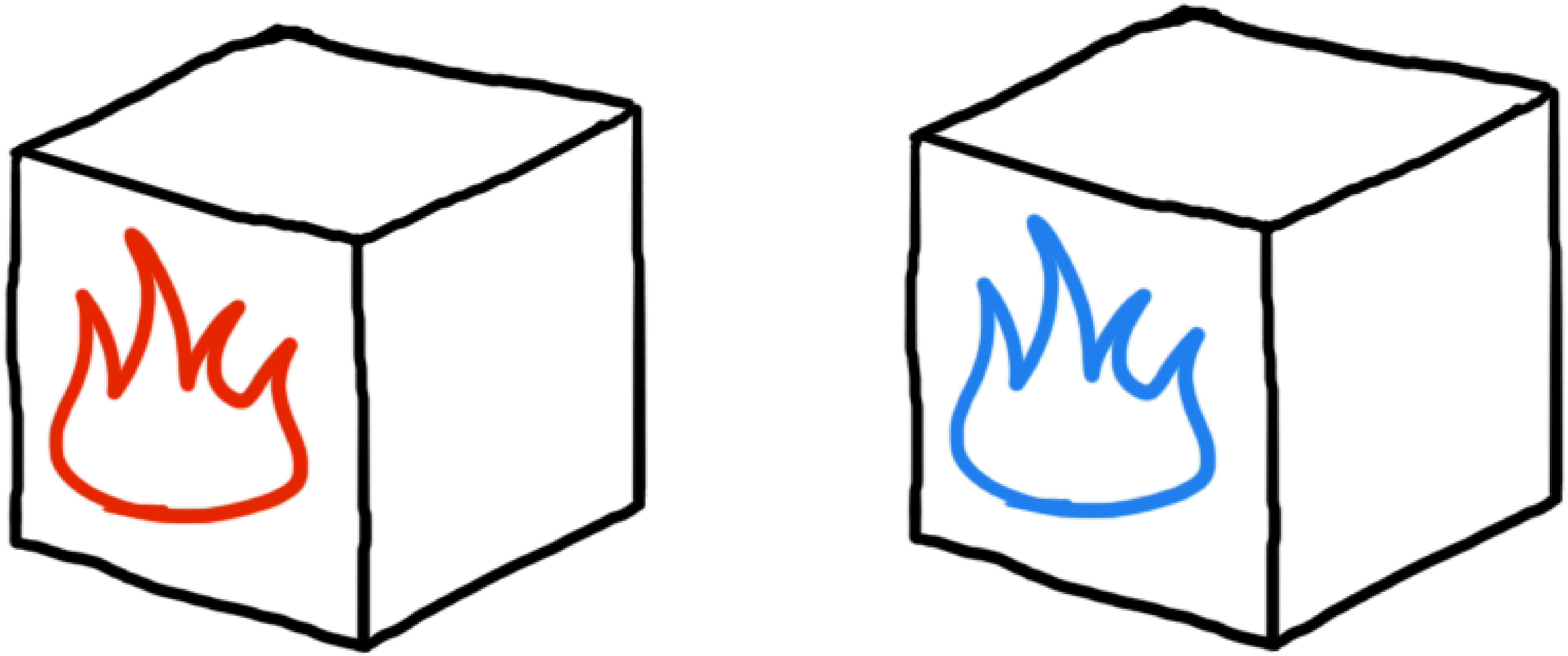


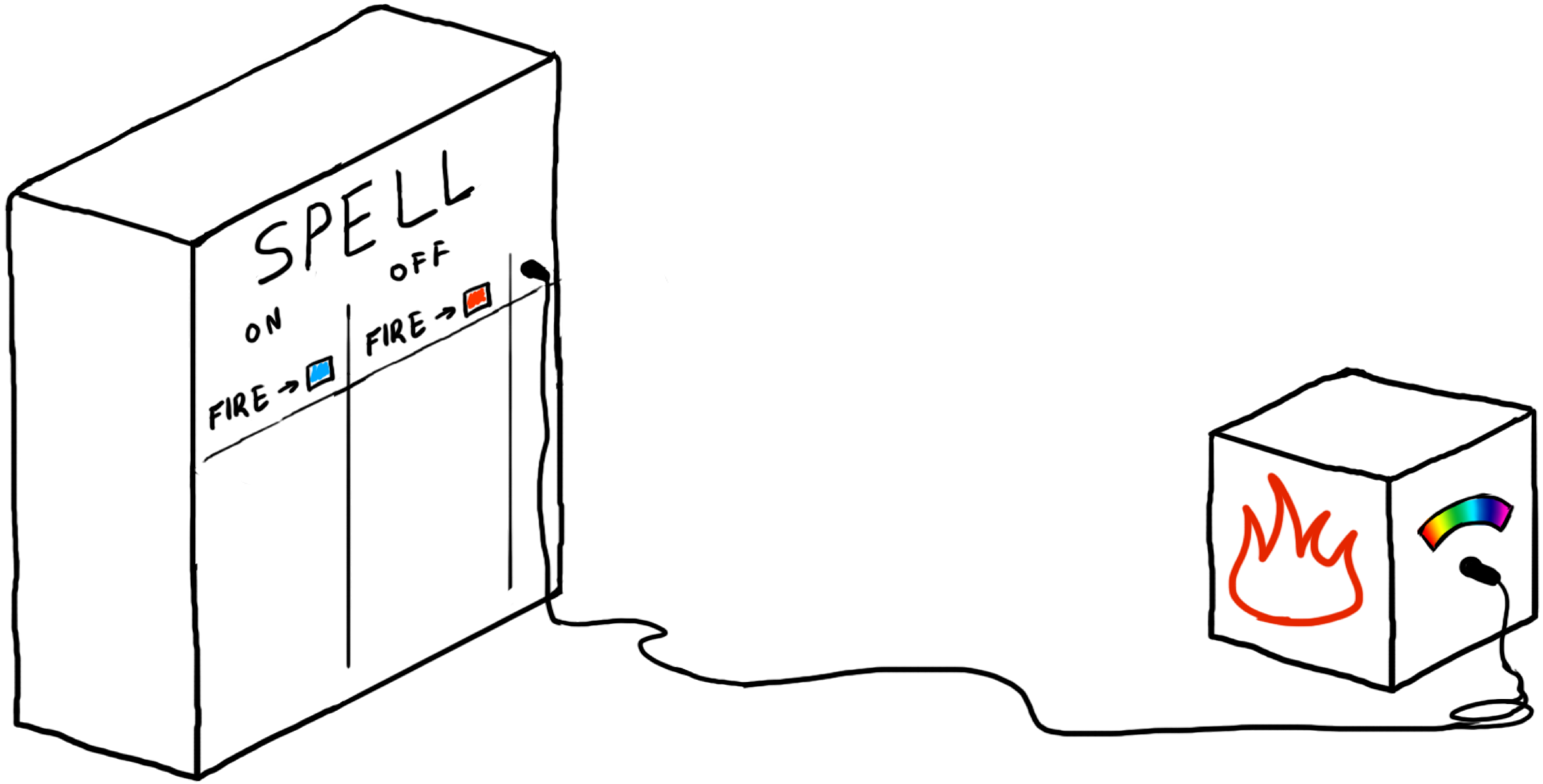
TOOLS

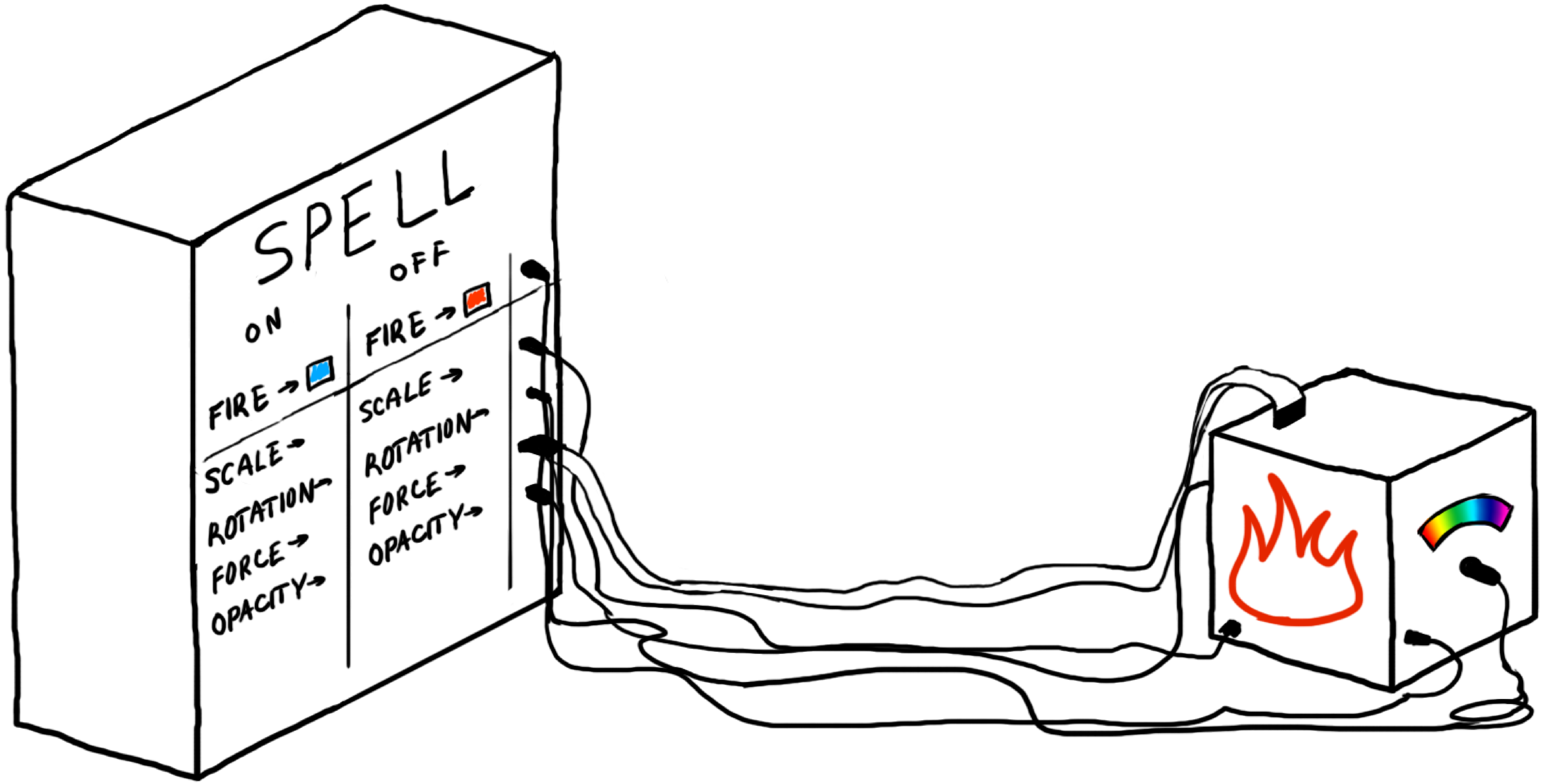


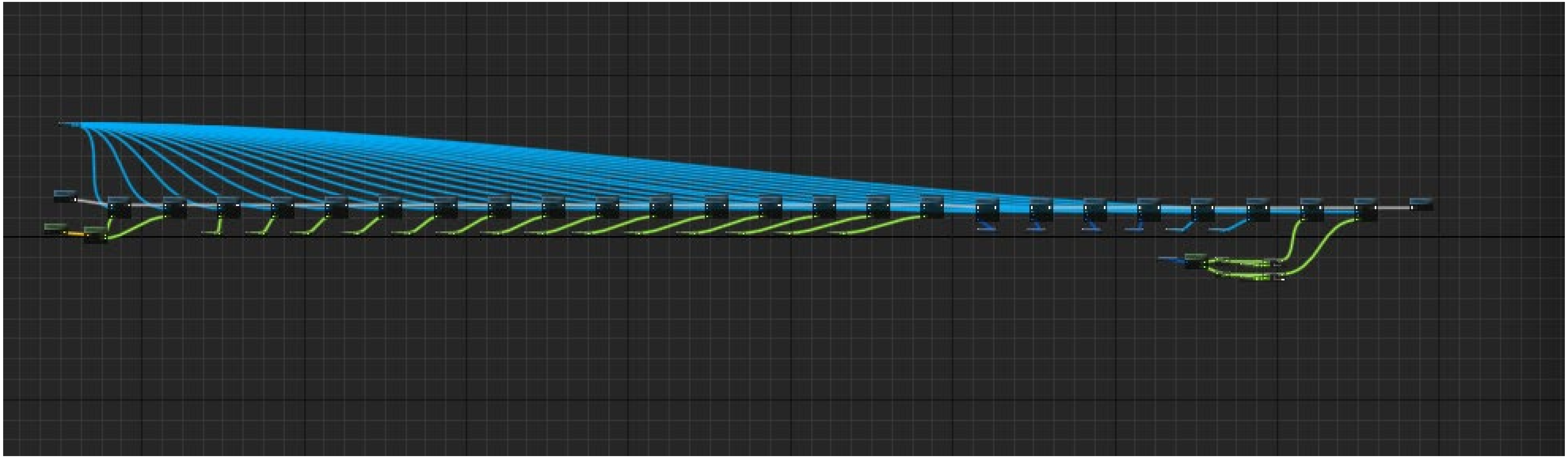
TOOLS

EXAMPLE

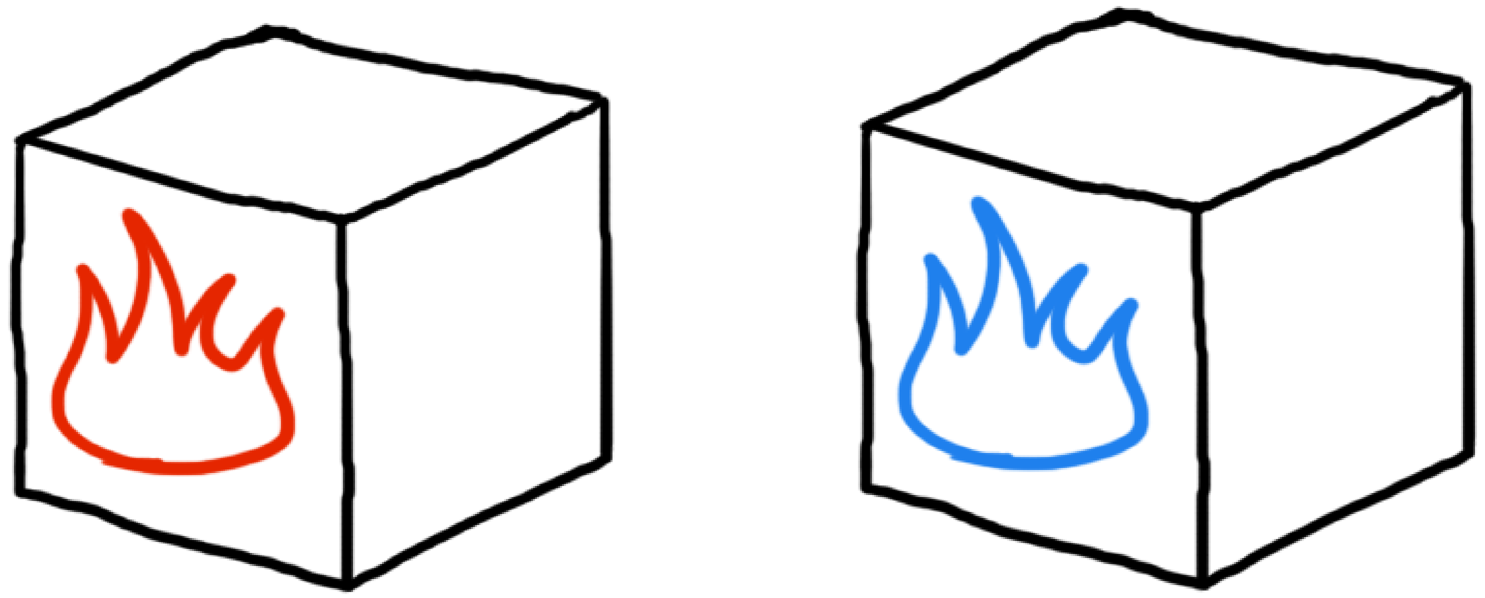



SPELL
ON
OFF
FIRE → 🟦
FIRE → 🟥

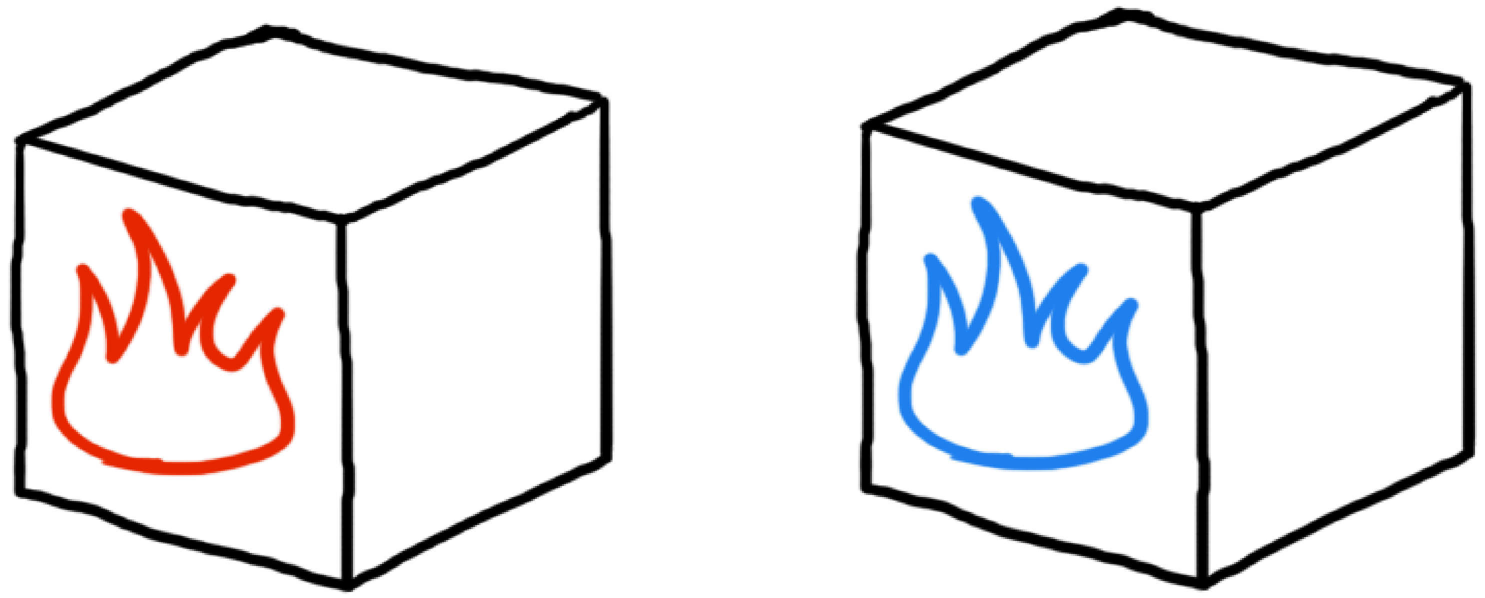

SPELL
ON
FIRE →
SCALE →
ROTATION →
FORCE →
OPACITY →
OFF
FIRE →
SCALE →
ROTATION →
FORCE →
OPACITY →



https://blueprintsfromhell.tumblr.com/



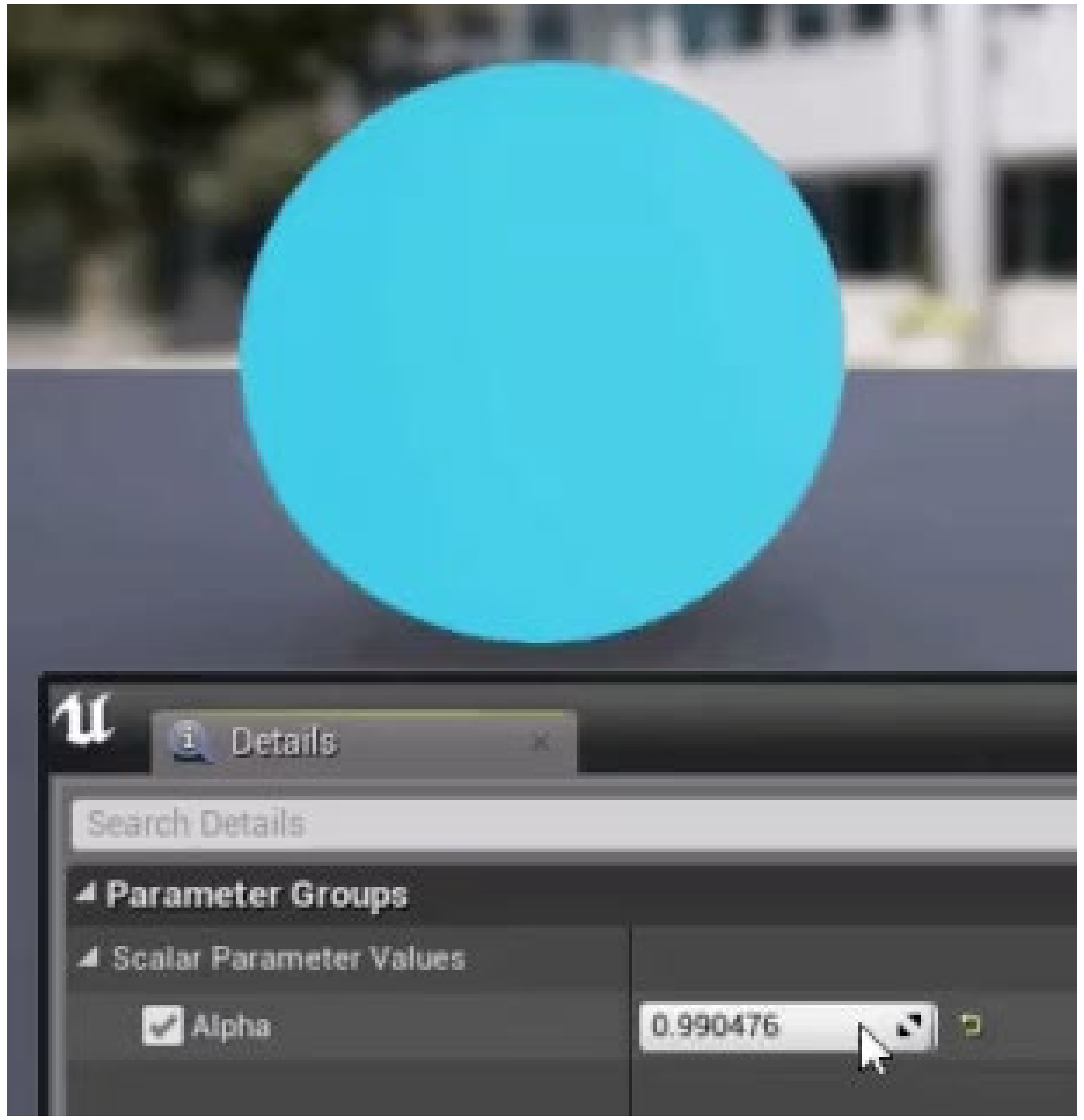
- Red/Blue
- Big/Small
- Opaque/transparent
- High turbulence/low turbulence



- Red/Blue
- Big/Small
- Opaque/transparent
- High turbulence/low turbulence
- **Normal/Under the spell influence**

ABSTRACTION

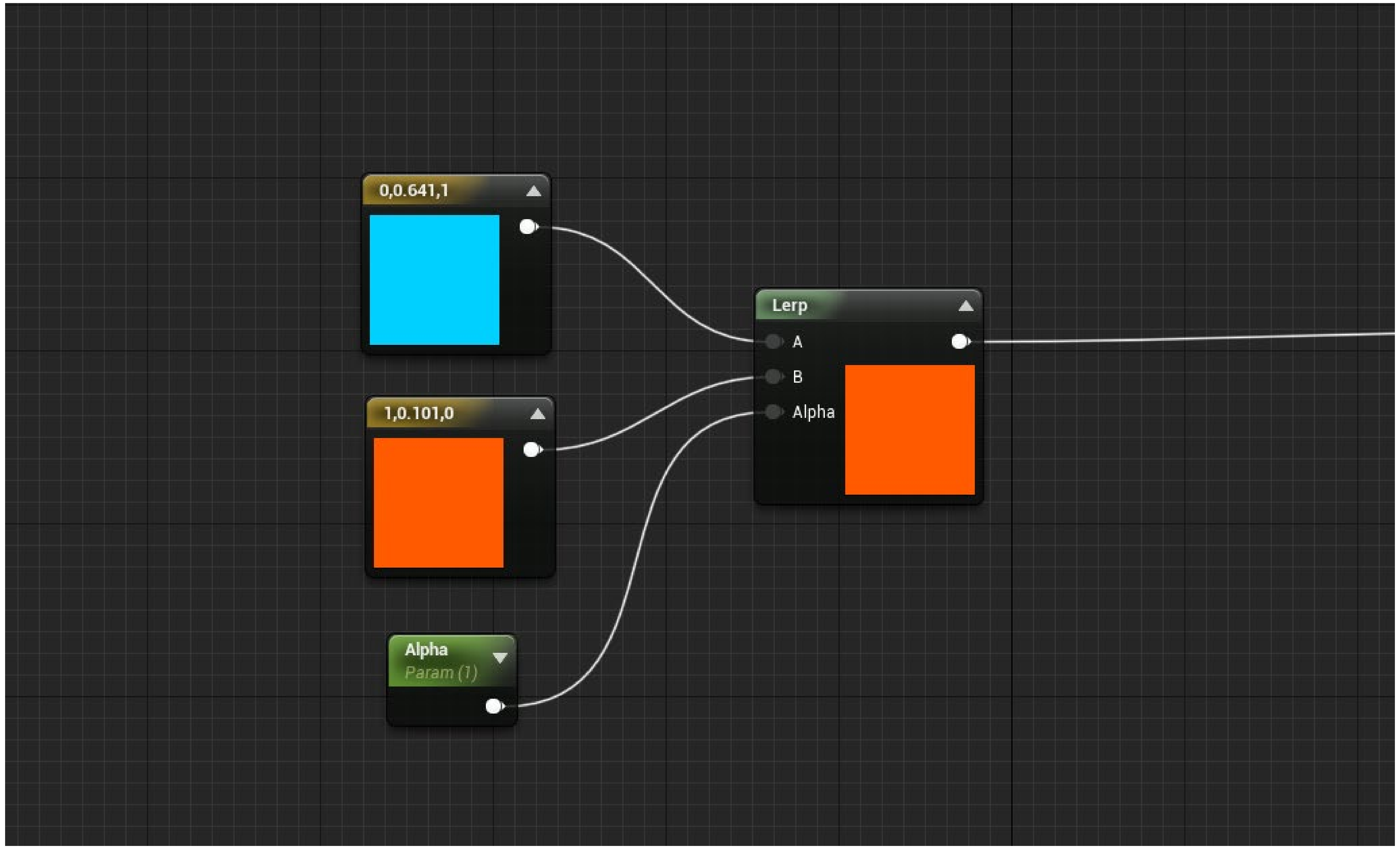
ABSTRACTION



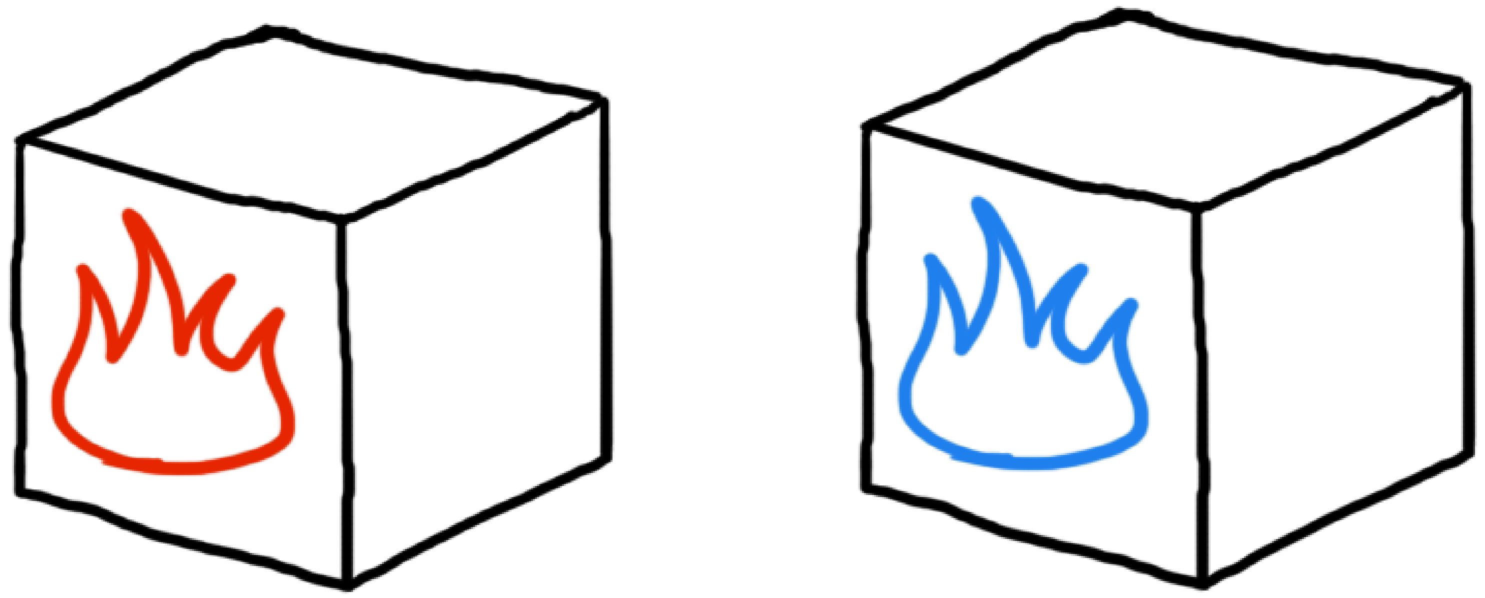
Parameter Groups

Scalar Parameter Values

☑ Alpha 0.990476



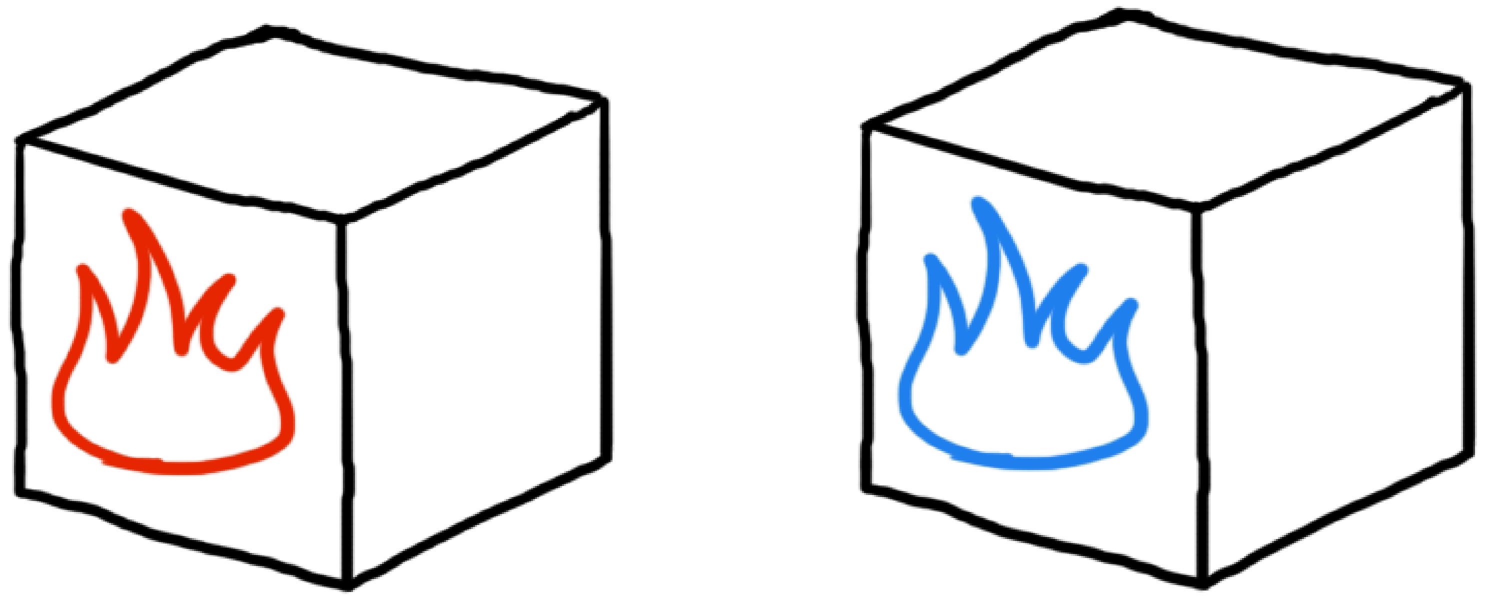
Concrete details



- **Red/Blue**
- **Big/Small**
- **Opaque/transparent**
- **High turbulence/low turbulence**

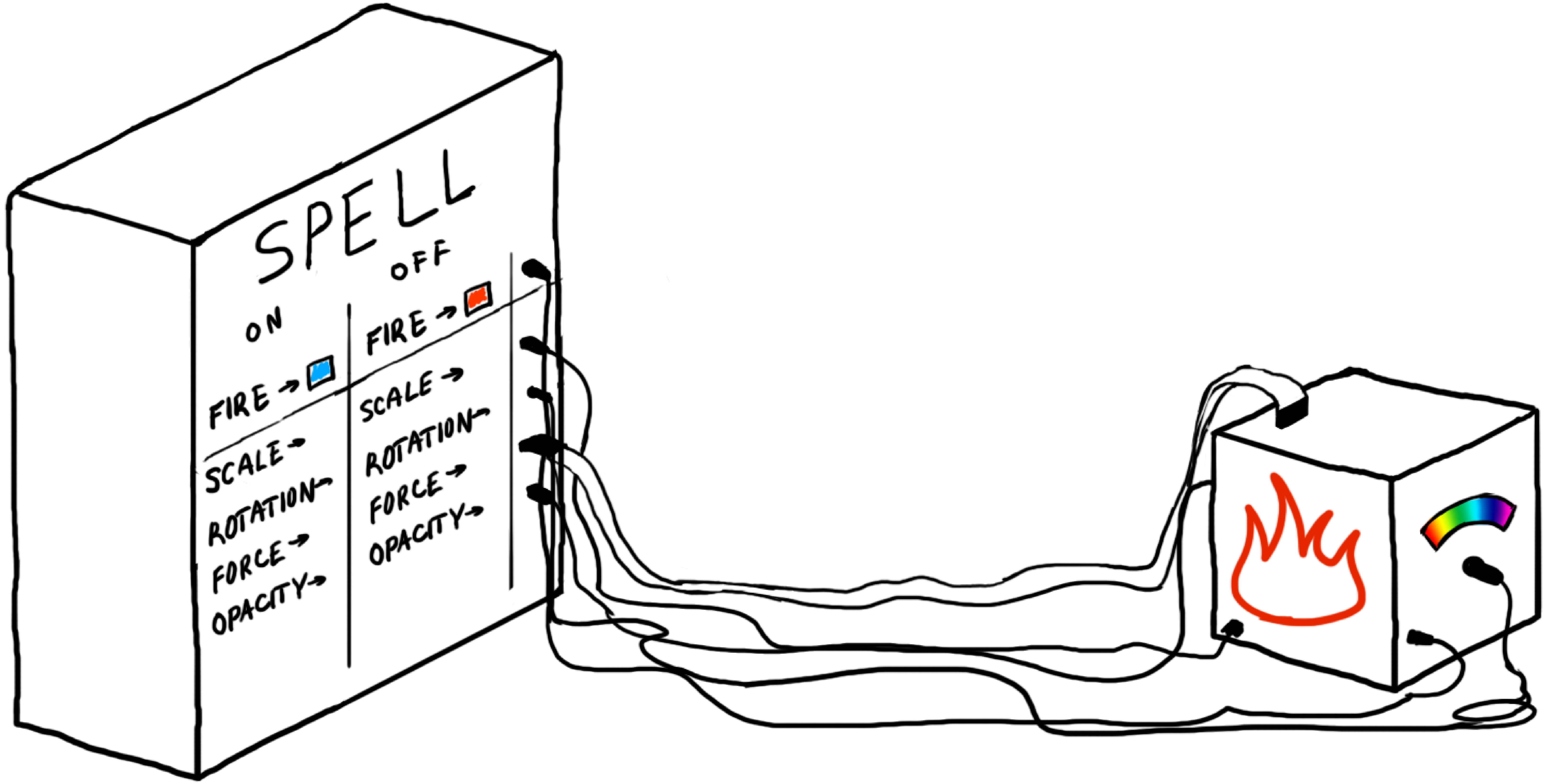
- Normal/Under the spell influence

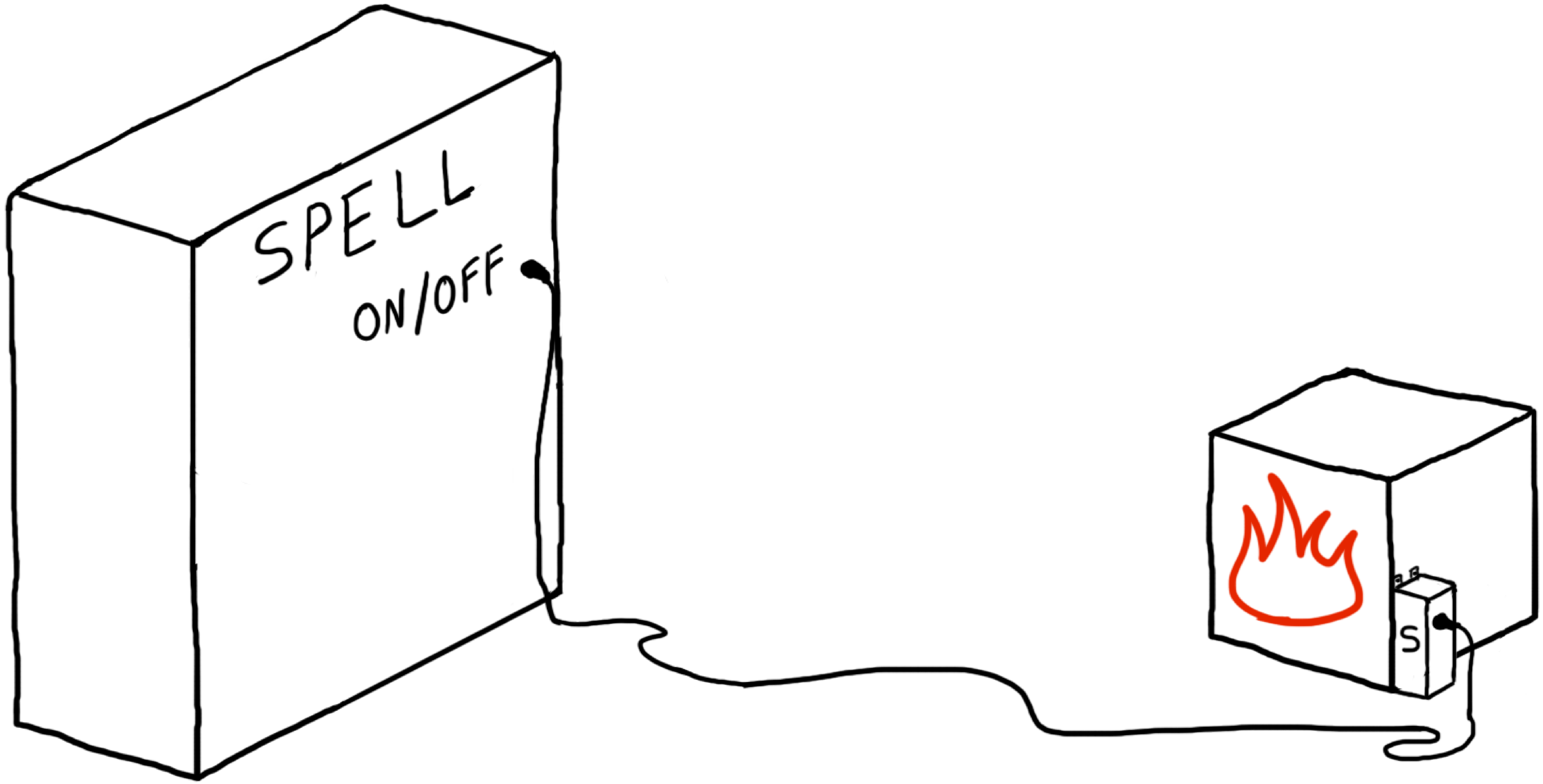
Concrete details

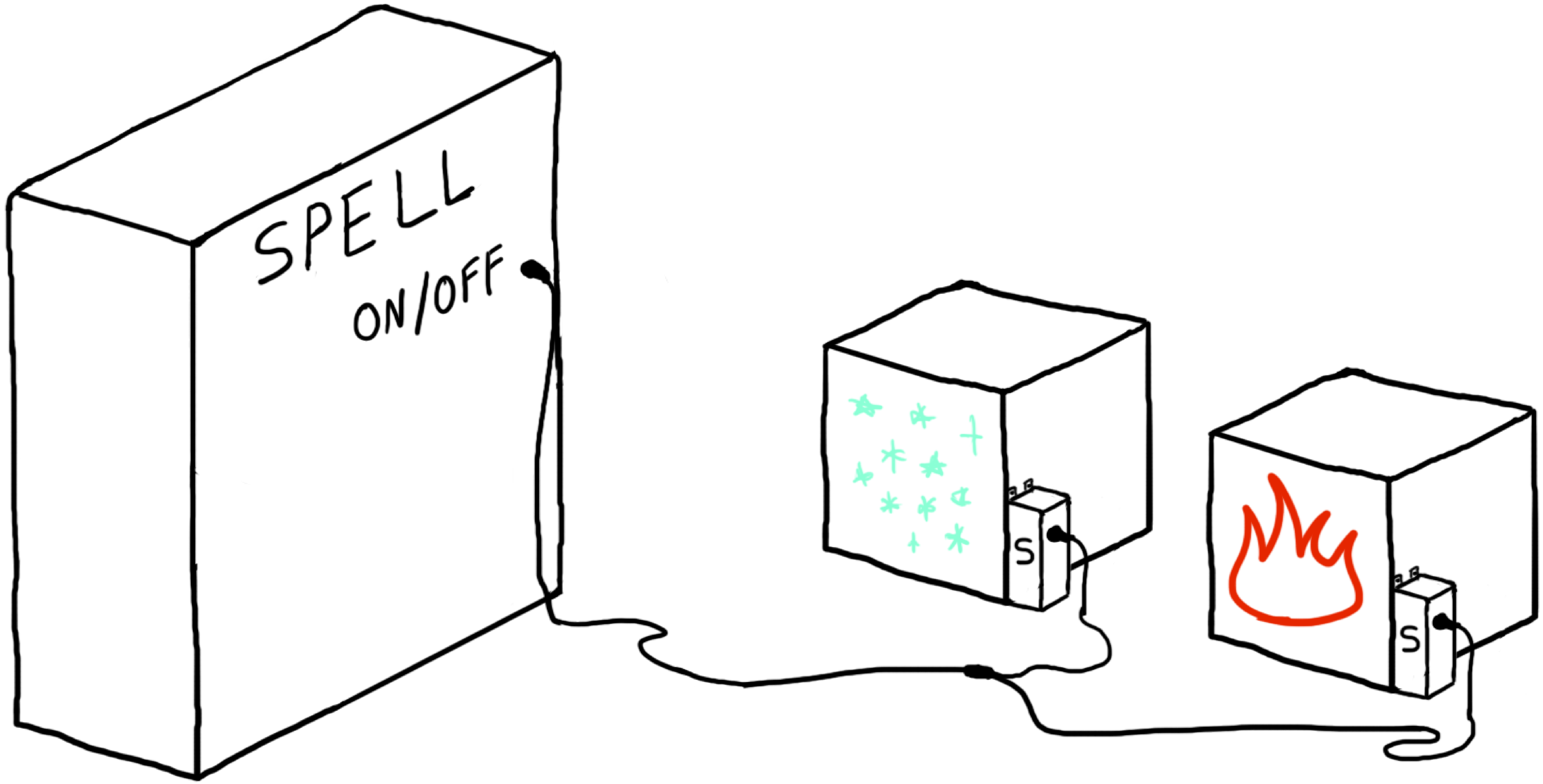


- **Red/Blue**
- **Big/Small**
- **Opaque/transparent**
- **High turbulence/low turbulence**
- **Normal/Under the spell influence**

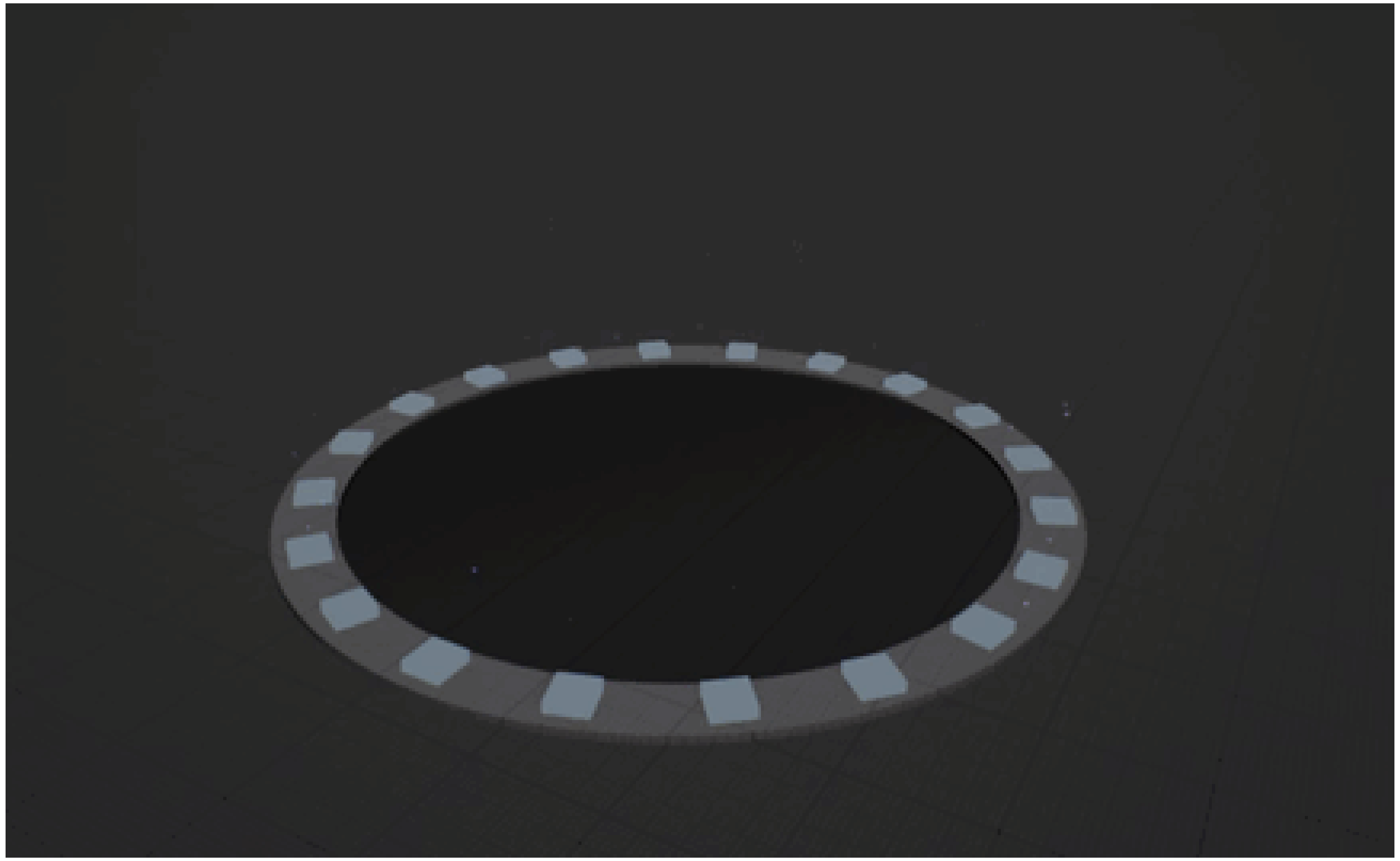
Abstract properties


SPELL
ON
OFF
FIRE →
SCALE →
ROTATION →
FORCE →
OPACITY →
FIRE →
SCALE →
ROTATION →
FORCE →
OPACITY →

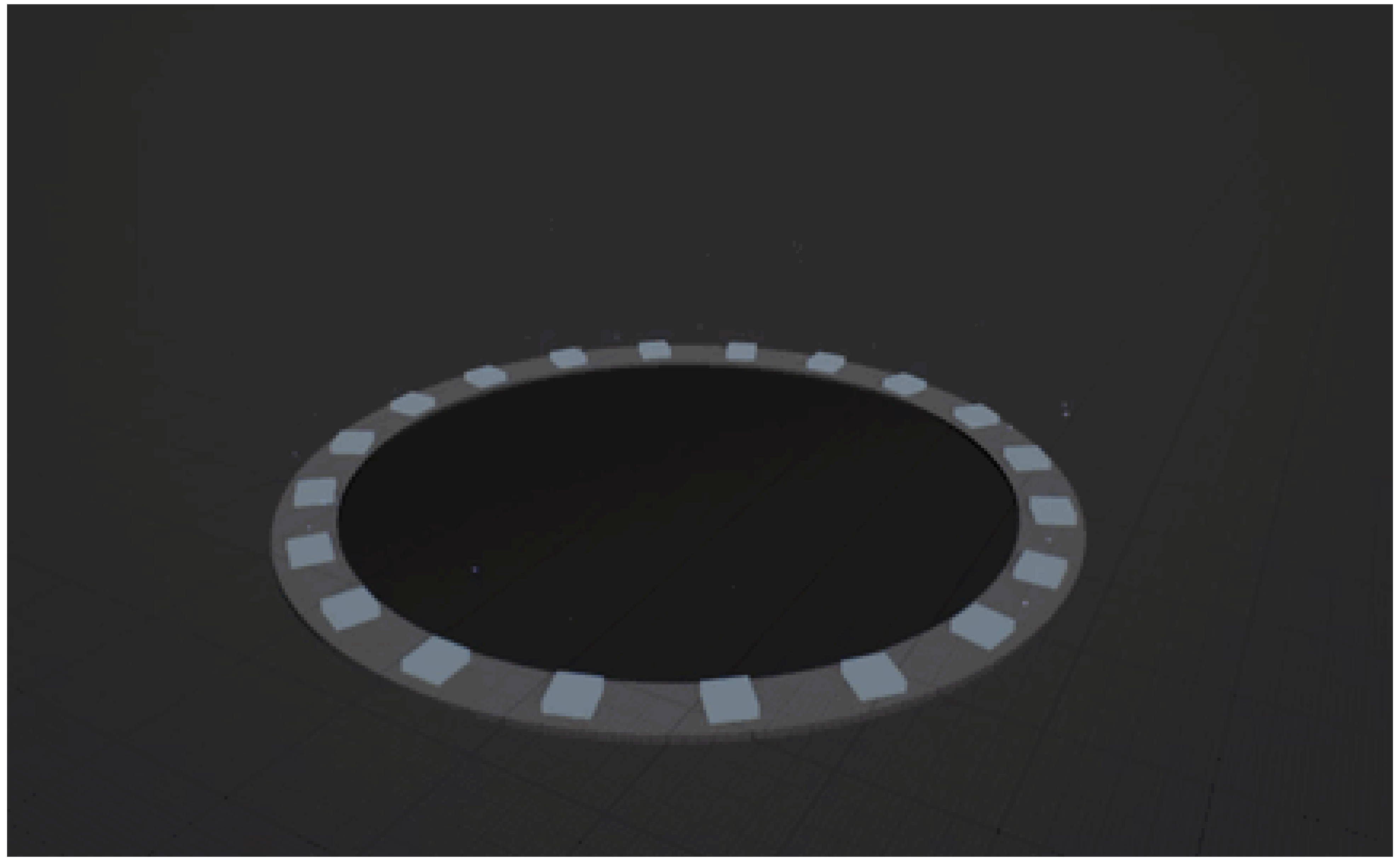

SPELL
ON/OFF
S

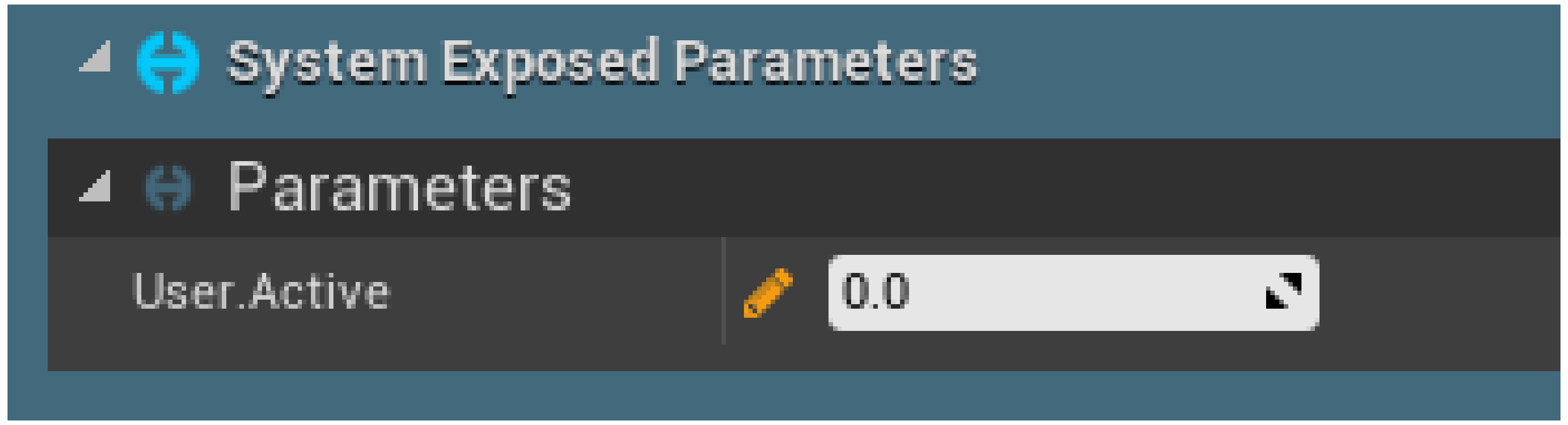

SPELL
ON/OFF
S
S

EXAMPLE



EXAMPLE

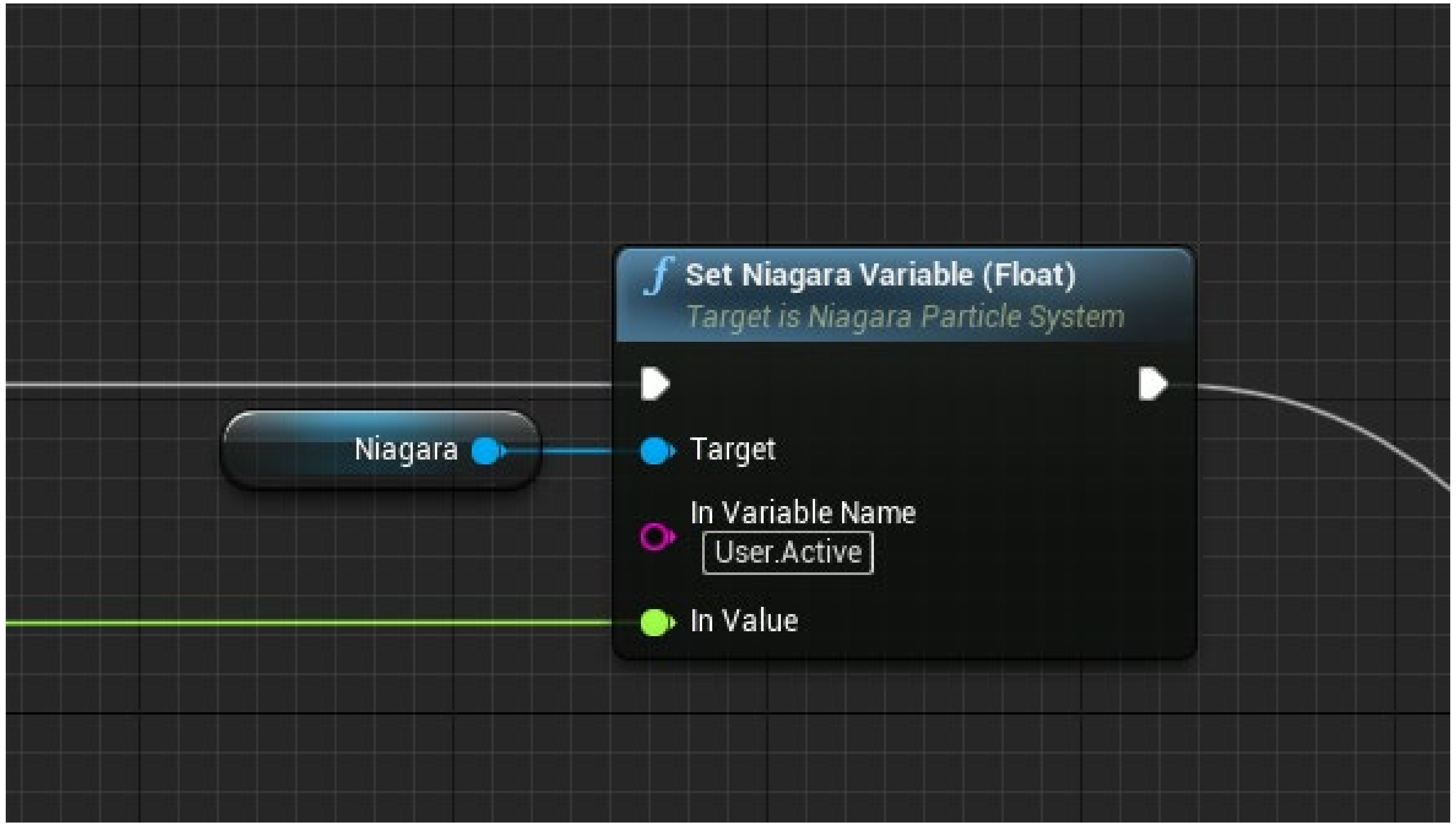




System Exposed Parameters

Parameters

User.Active 0.0



Set Niagara Variable (Float)

Target is Niagara Particle System

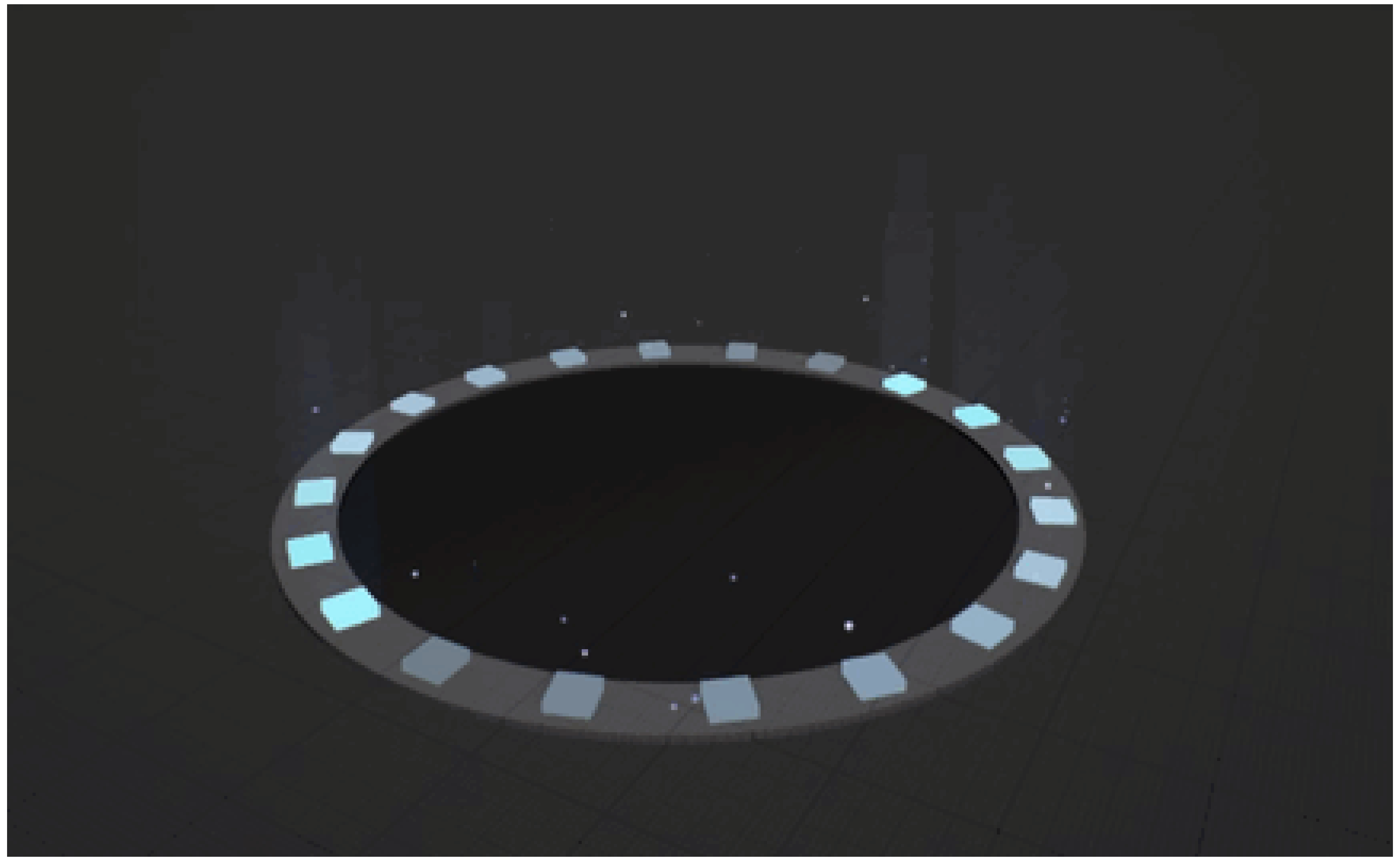
Niagara

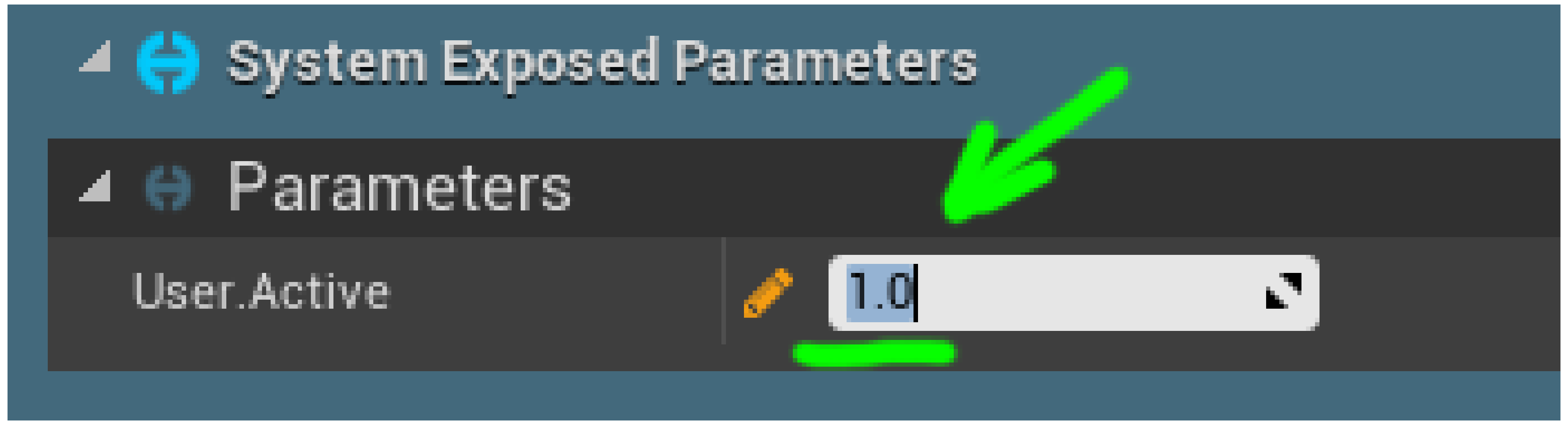
Target

In Variable Name User.Active

In Value

EXAMPLE

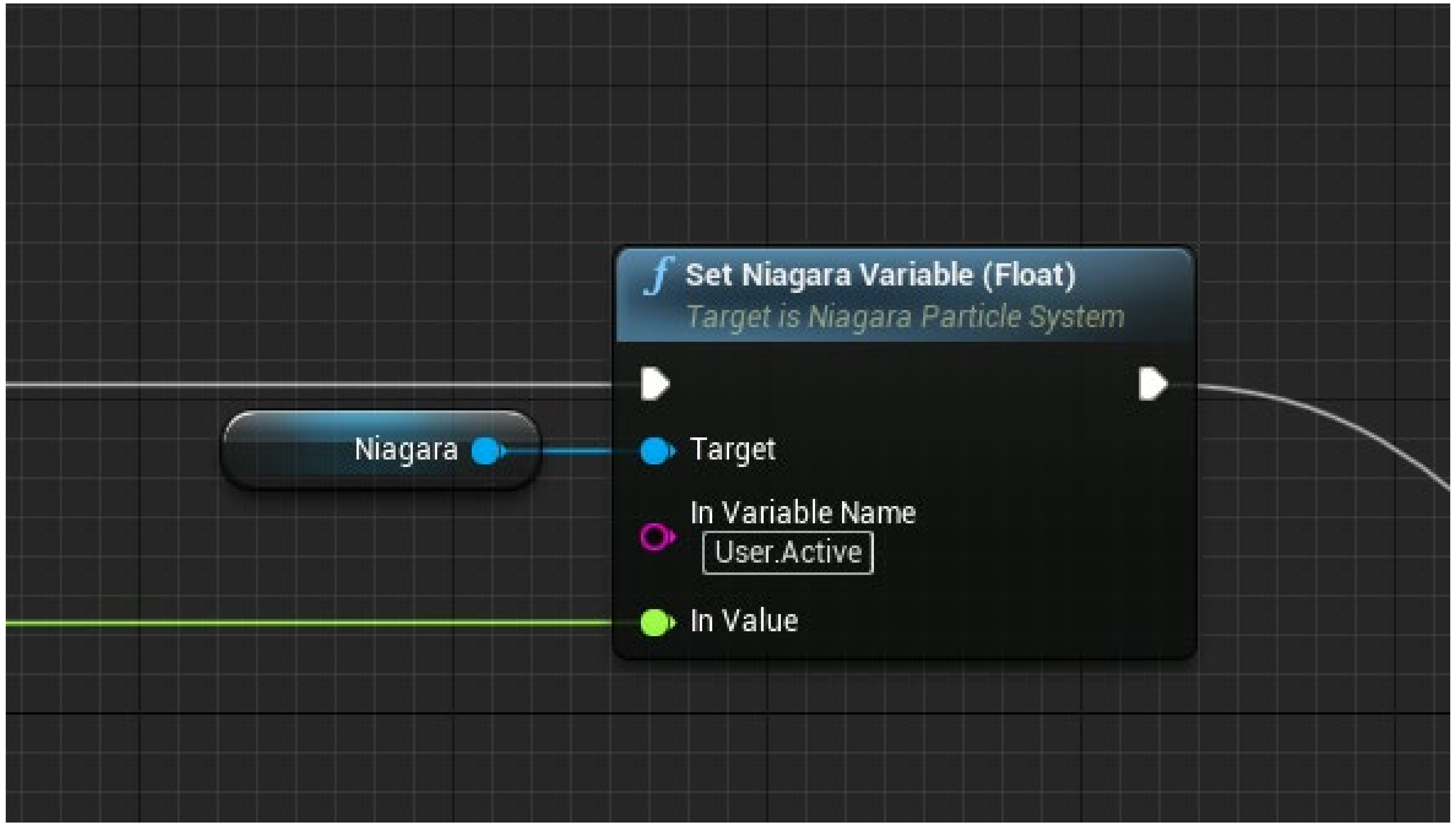




System Exposed Parameters

Parameters

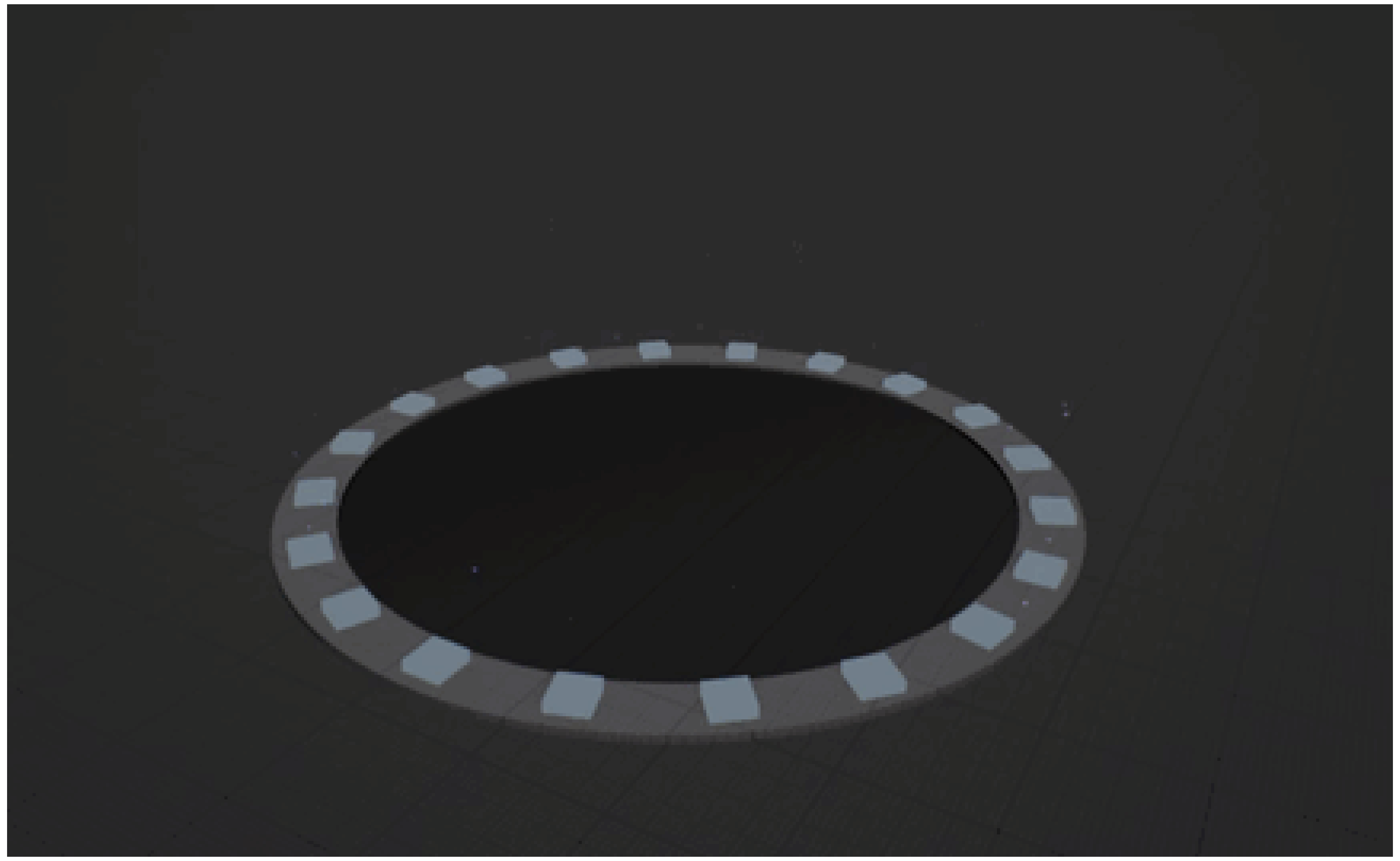
User.Active 1.0

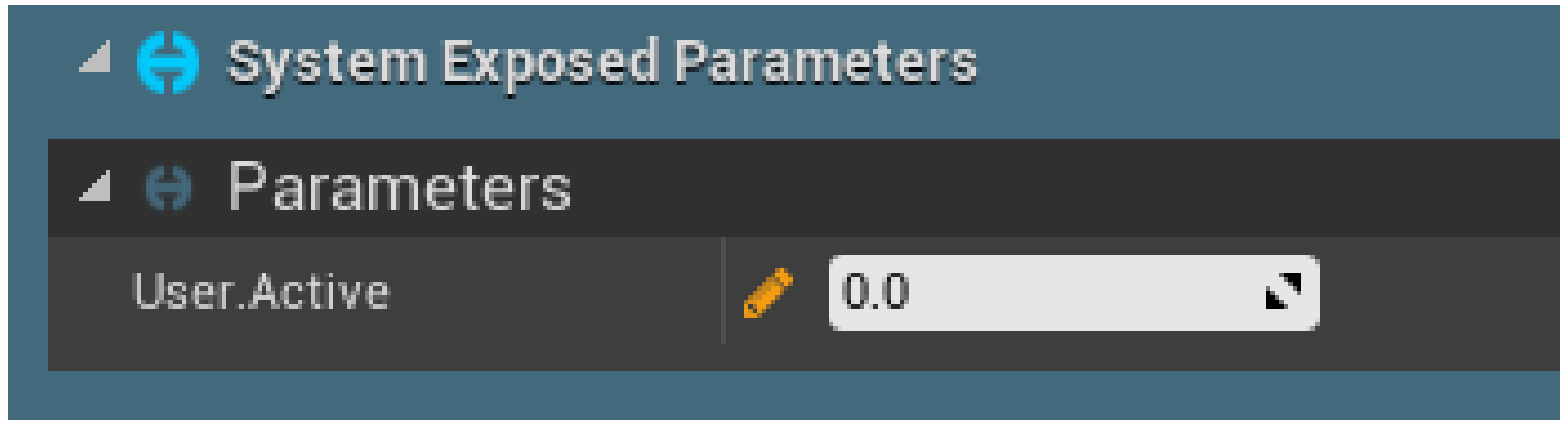


Set Niagara Variable (Float)

Target is Niagara Particle System

EXAMPLE

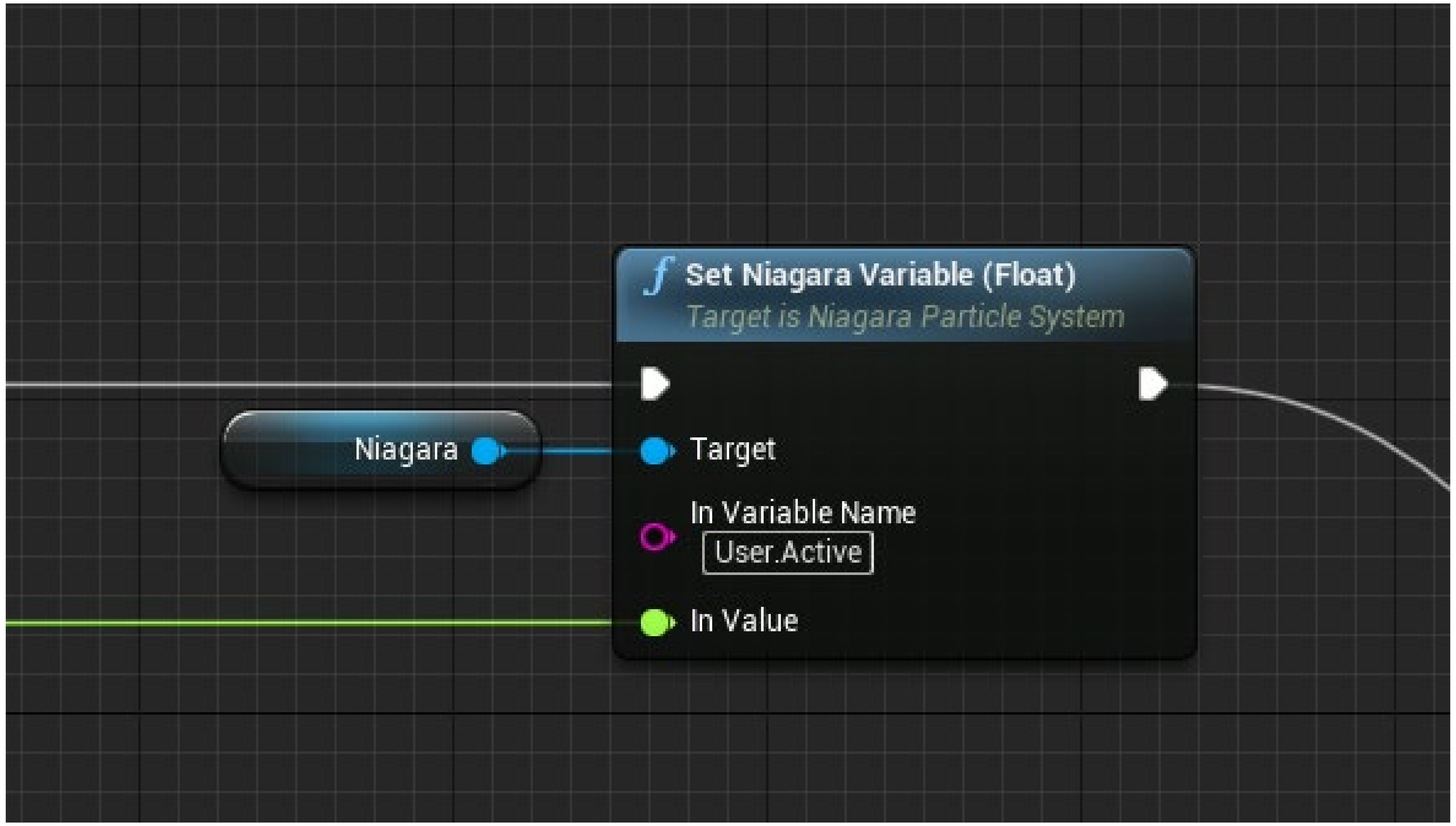




System Exposed Parameters

Parameters

User.Active 0.0



Set Niagara Variable (Float)

Target is Niagara Particle System

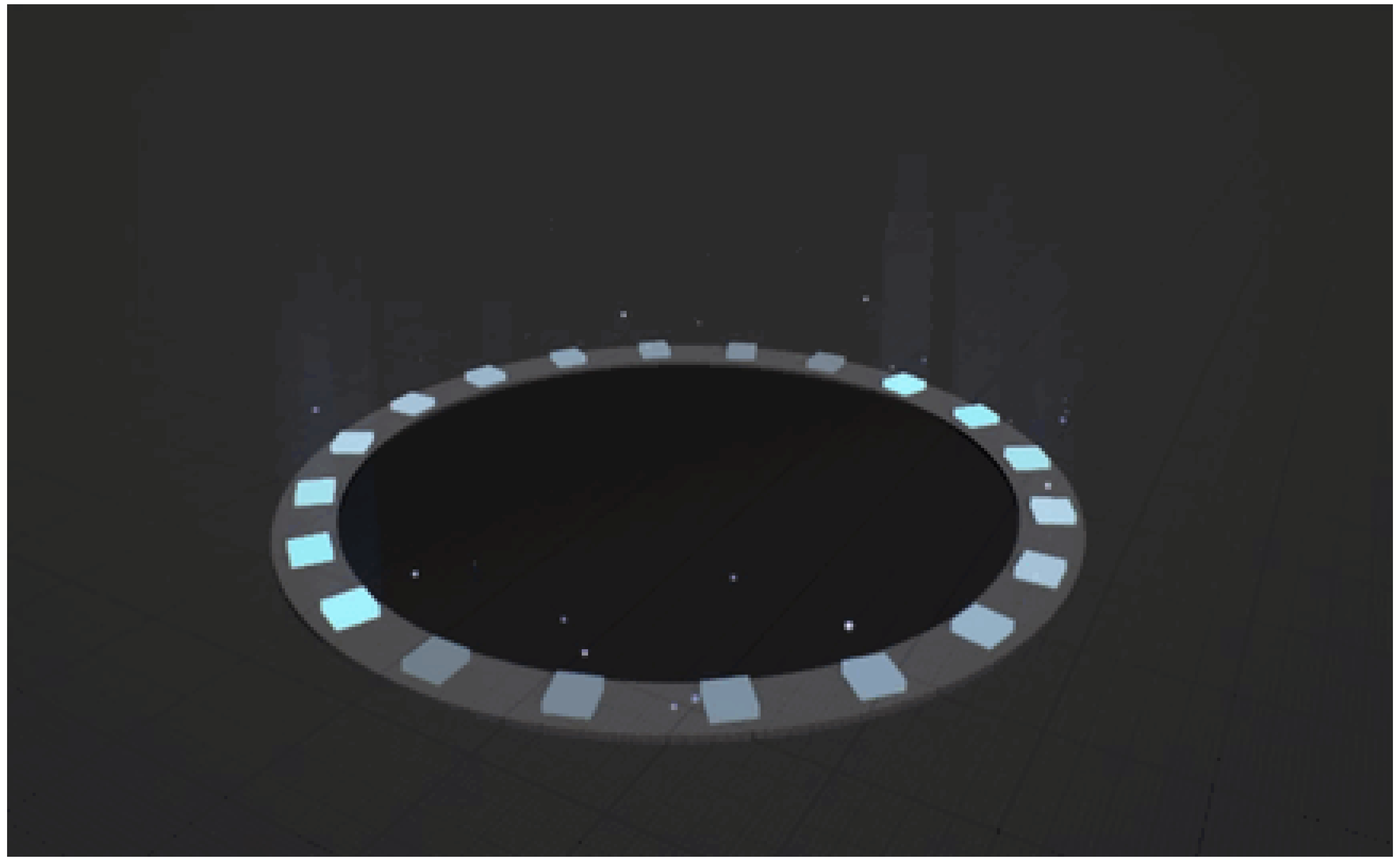
Niagara

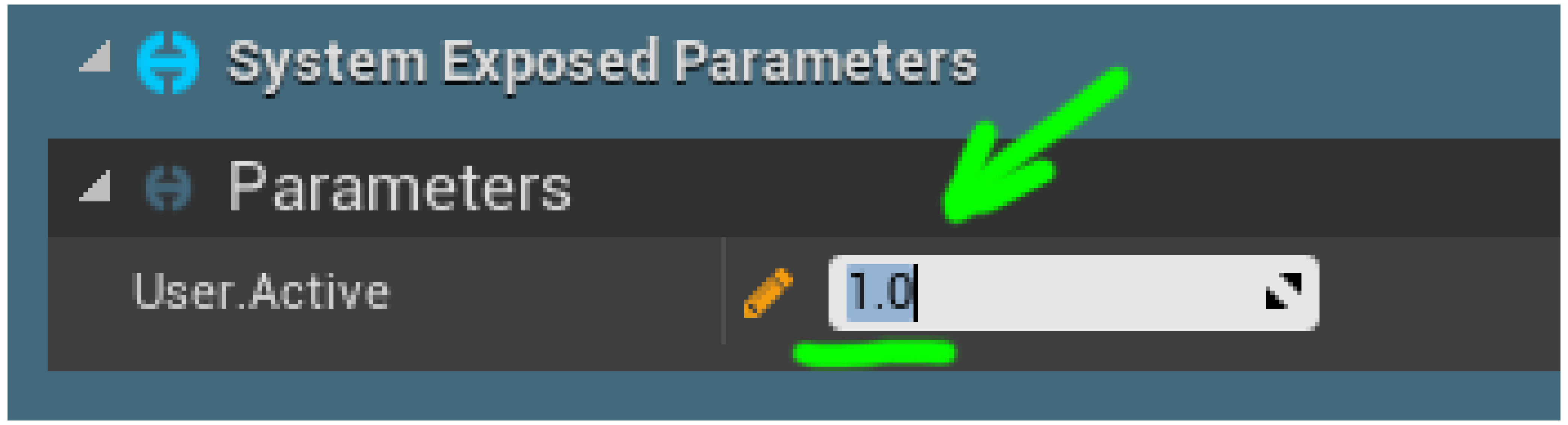
Target

In Variable Name User.Active

In Value

EXAMPLE

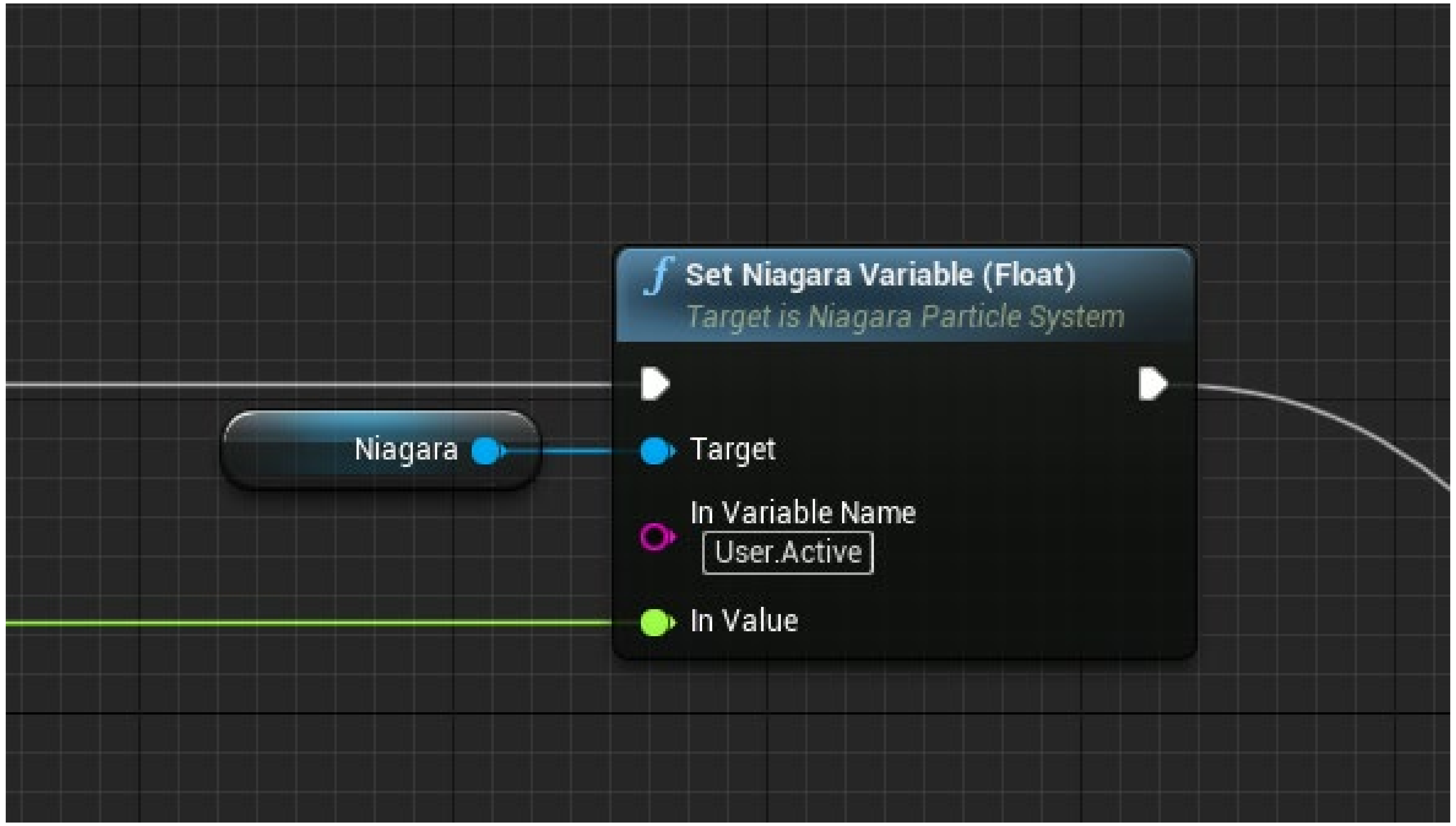




System Exposed Parameters

Parameters

User.Active 1.0



Set Niagara Variable (Float)

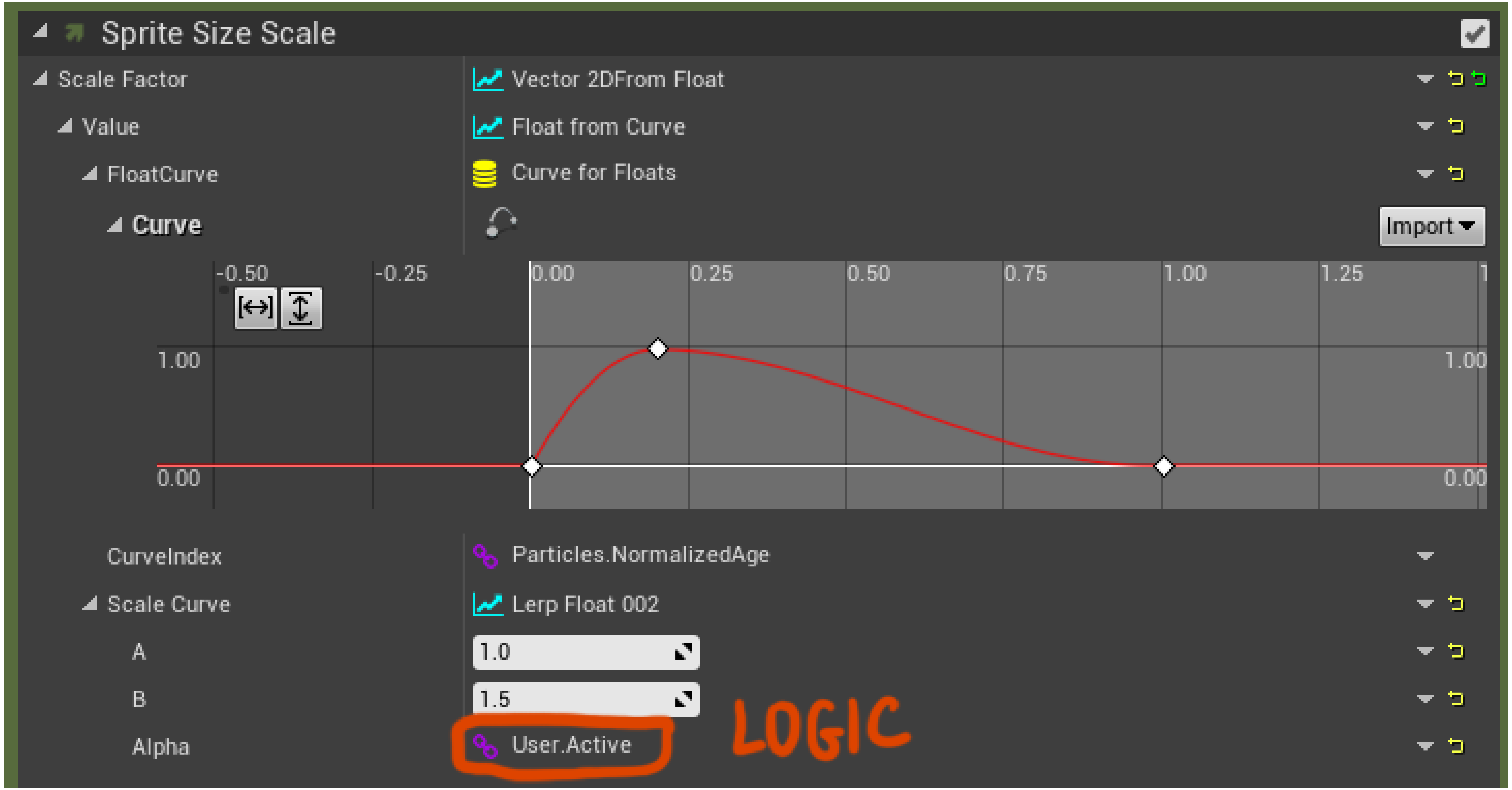
Target is Niagara Particle System

Niagara

Target

In Variable Name: User.Active

In Value



Sprite Size Scale

Scale Factor: Vector 2DFrom Float

Value: Float from Curve

FloatCurve: Curve for Floats

Curve

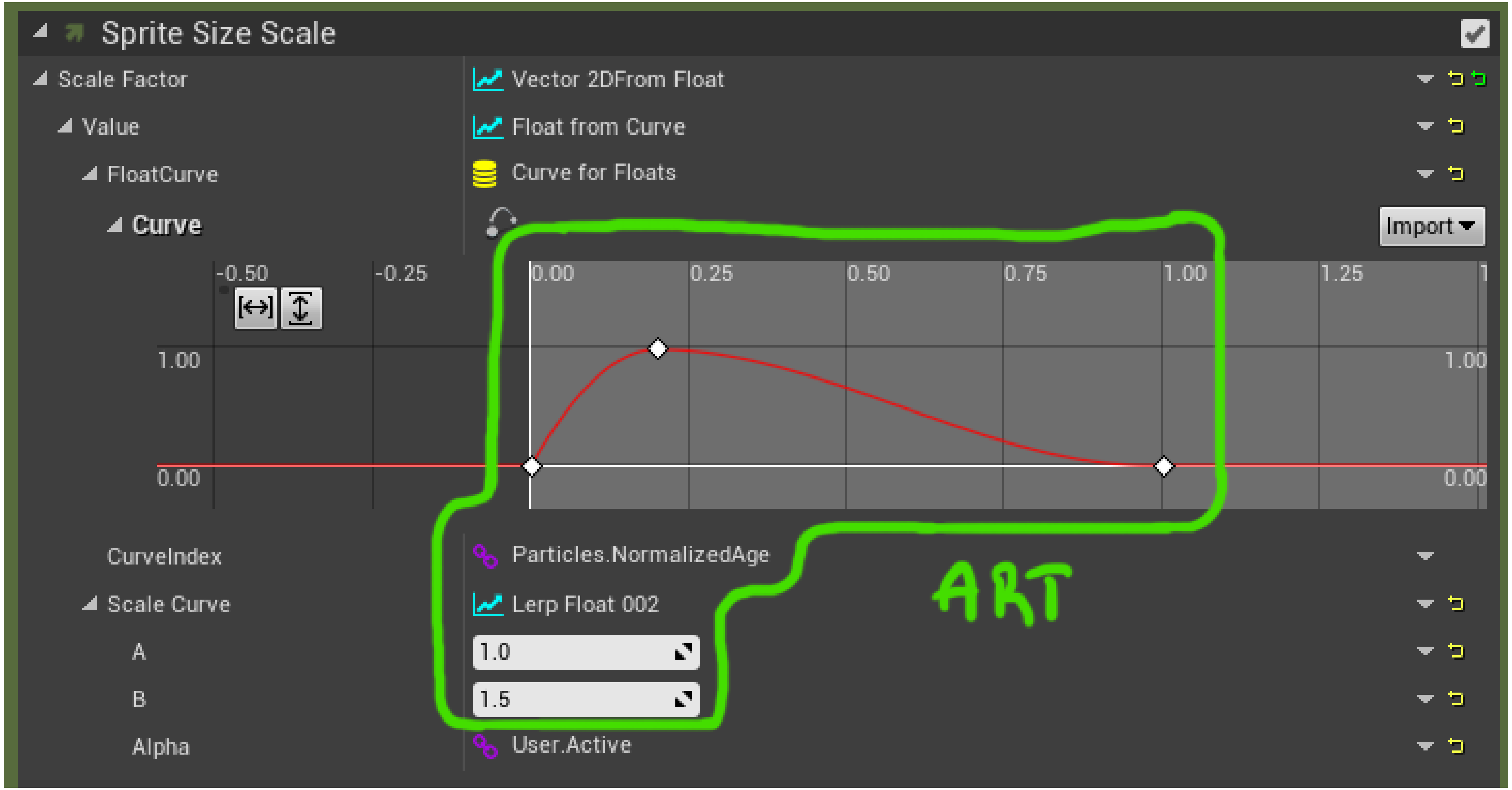
CurveIndex: Particles.NormalizedAge

Scale Curve: Lerp Float 002

A: 1.0

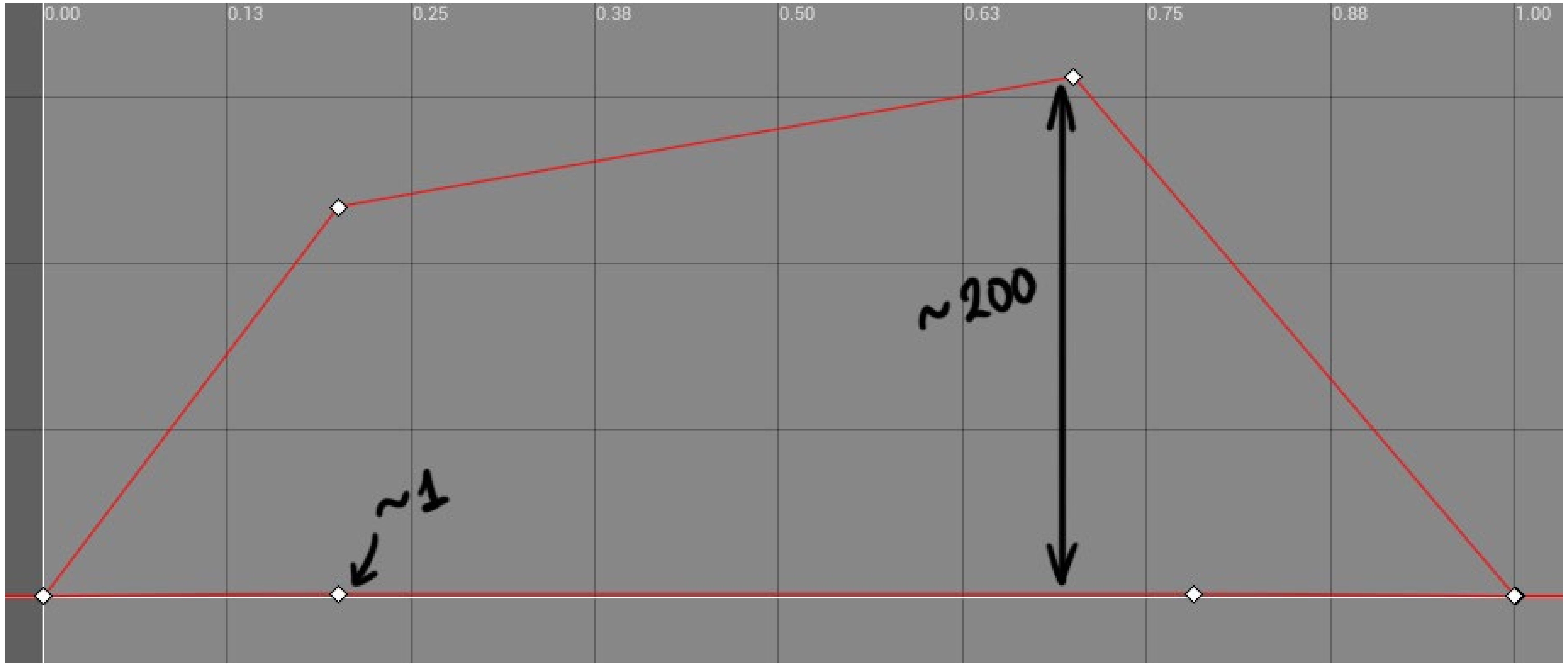
B: 1.5

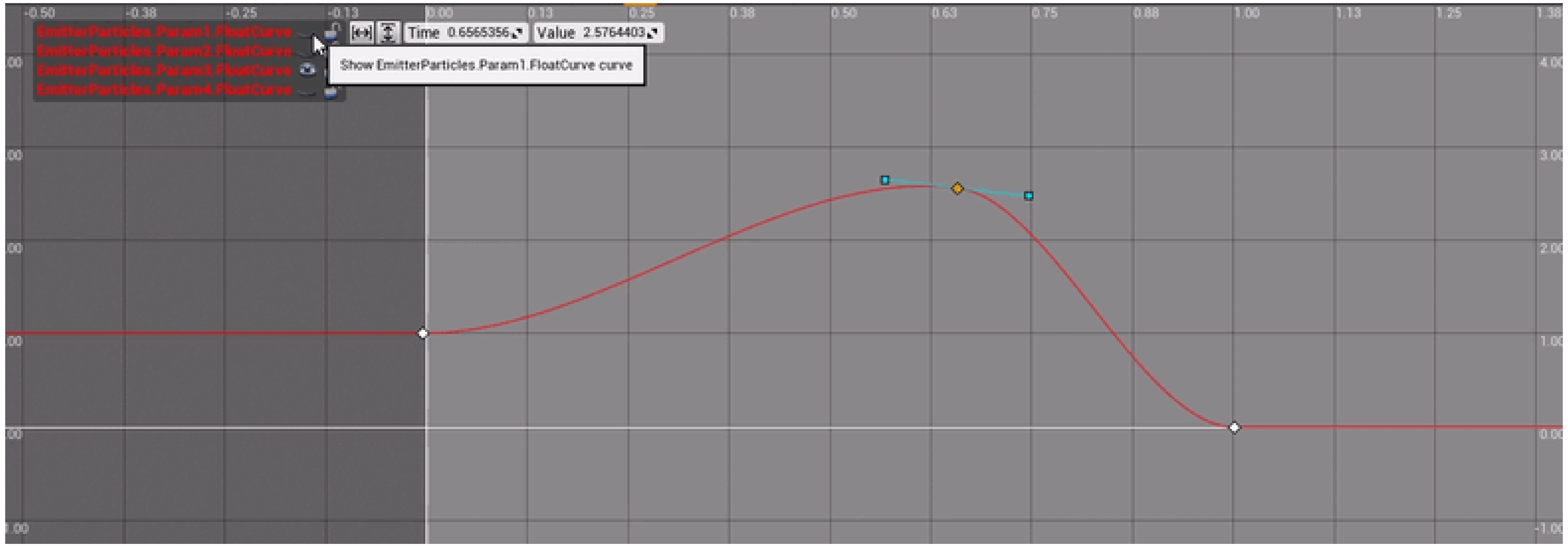
Alpha: User.Active



Sprite Size Scale

Scale Factor	Vector 2DFrom Float
Value	Float from Curve
FloatCurve	Curve for Floats
Curve	Import
CurveIndex	Particles.NormalizedAge
Scale Curve	Lerp Float 002
A	1.0
B	1.5
Alpha	User.Active

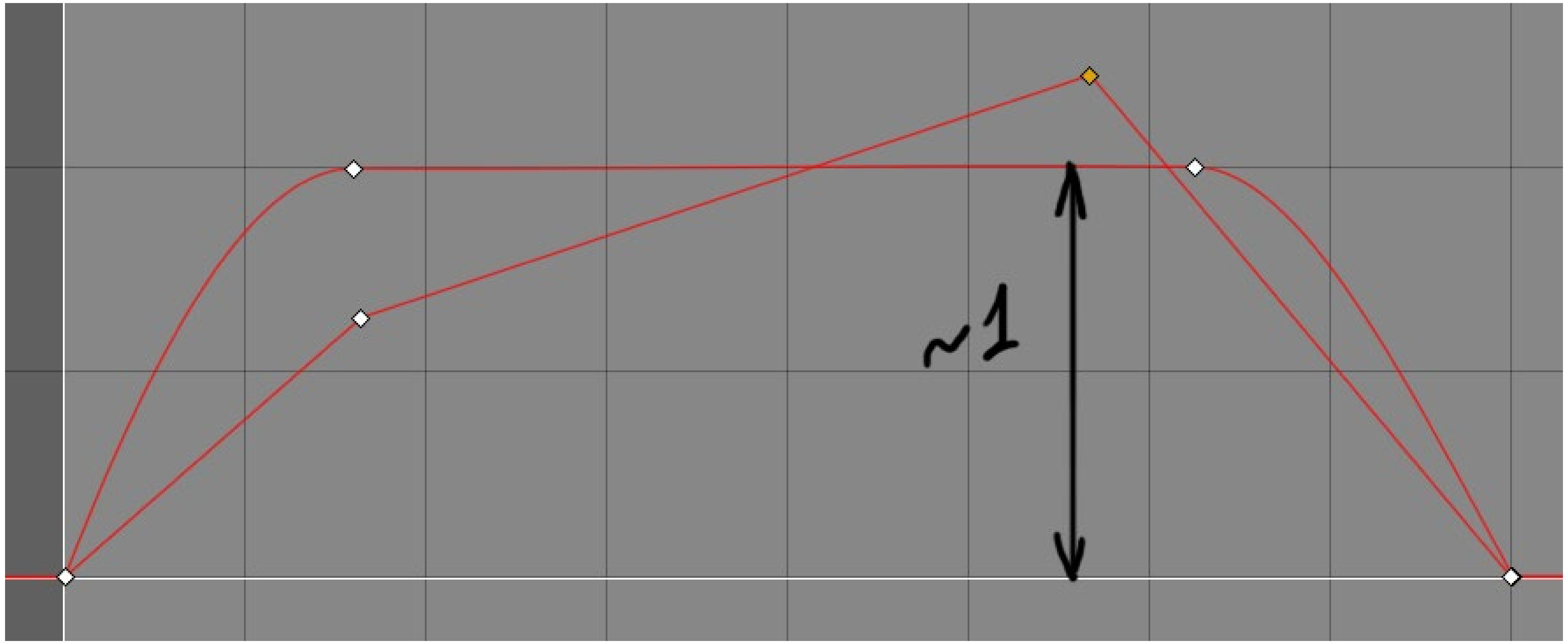


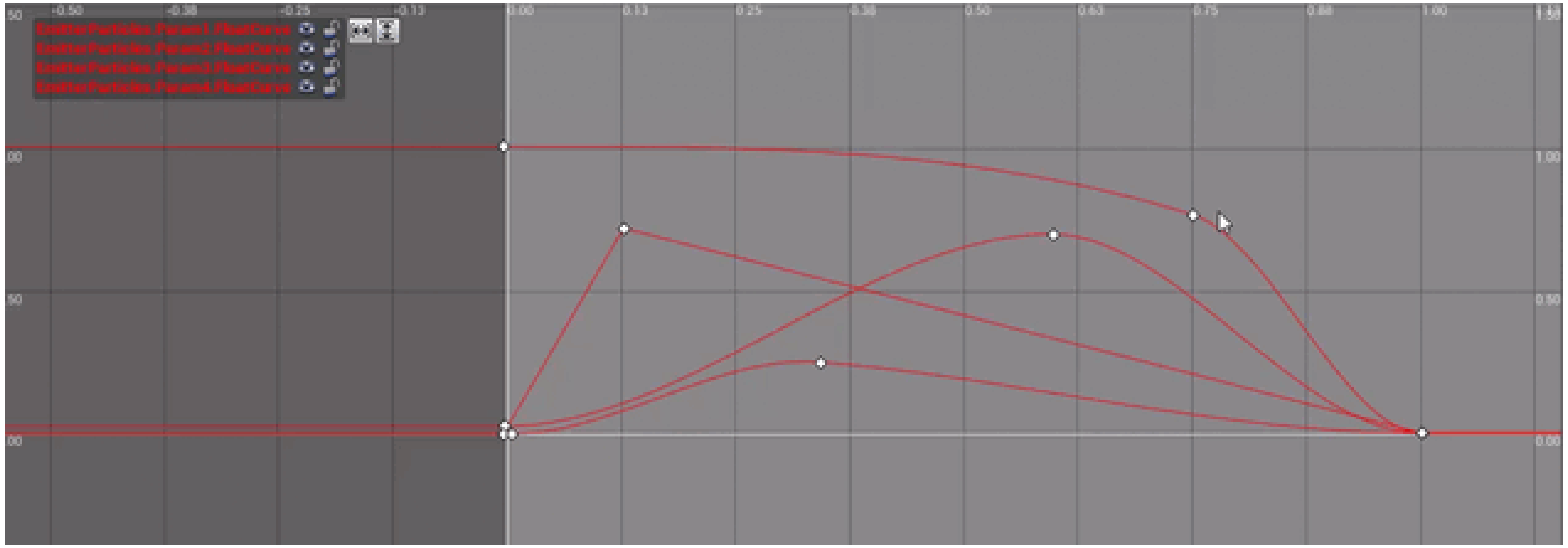

EmitterParticles.Param1.FloatCurve
EmitterParticles.Param2.FloatCurve
EmitterParticles.Param3.FloatCurve
EmitterParticles.Param4.FloatCurve

Time 0.6565356

Value 2.5764403

Show EmitterParticles.Param1.FloatCurve curve

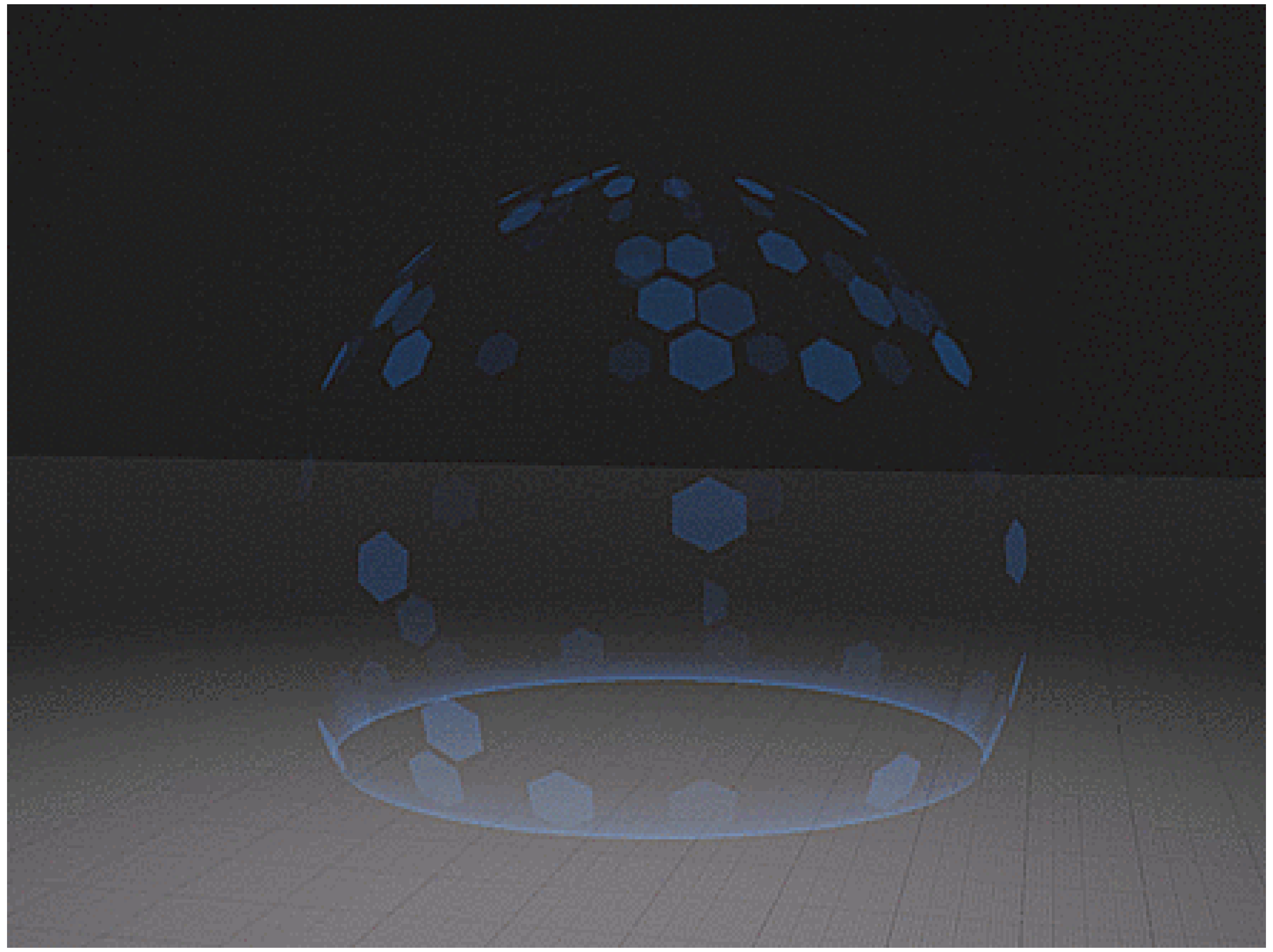

~1


EmitterParticles.Param1.FloatCurve
EmitterParticles.Param2.FloatCurve
EmitterParticles.Param3.FloatCurve
EmitterParticles.Param4.FloatCurve
-0.50
-0.38
-0.25
-0.13
0.00
0.13
0.25
0.38
0.50
0.63
0.75
0.88
1.00
1.50
1.00
0.50
0.00

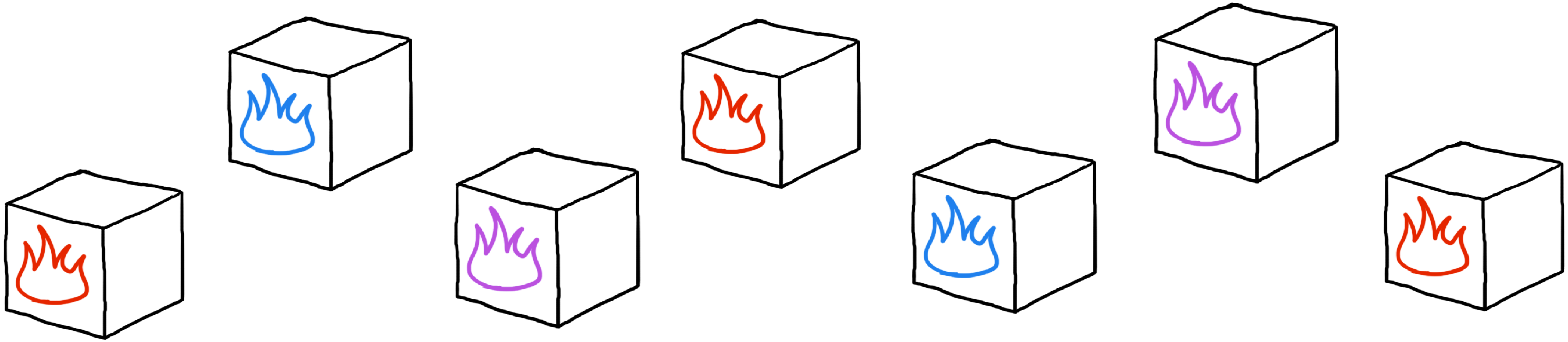
ABSTRACTION

- Separate abstract properties (logic) and concrete details of implementation (art)
- Concentrate on high level logic first, worry about art later
- Try to keep values between 0 and 1

ABSTRACTION



DISCLAIMER



DISCLAIMER

- How long is going to take to implement this?
- Is it going to be easy to tweak?
- Can we add/remove/replace components later if needed?

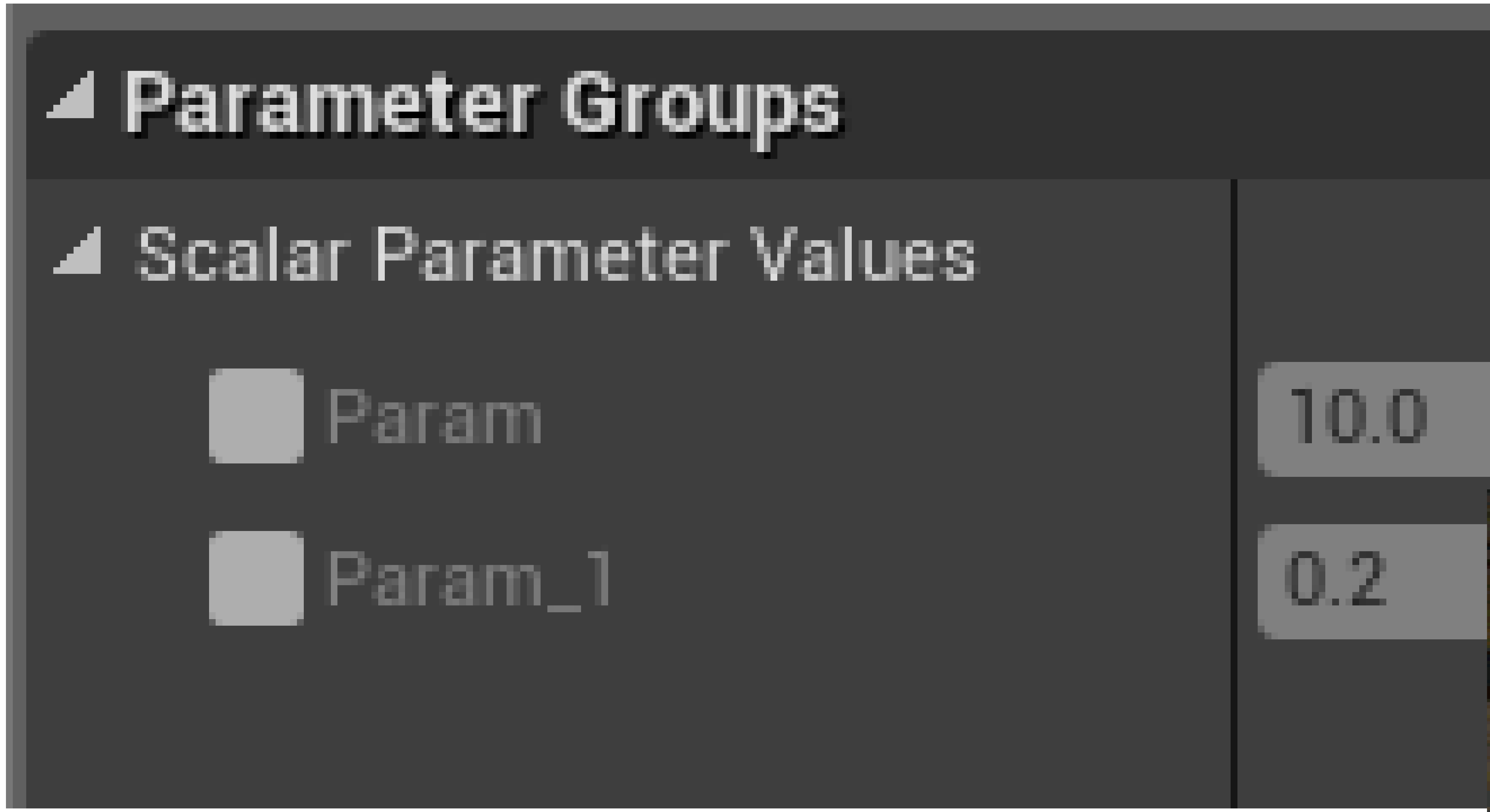
SOMETHING ISN’T WORKING

(help)

READ THE INSTRUCTIONS

(before all else fails)

UNDERSTAND THE SYSTEM

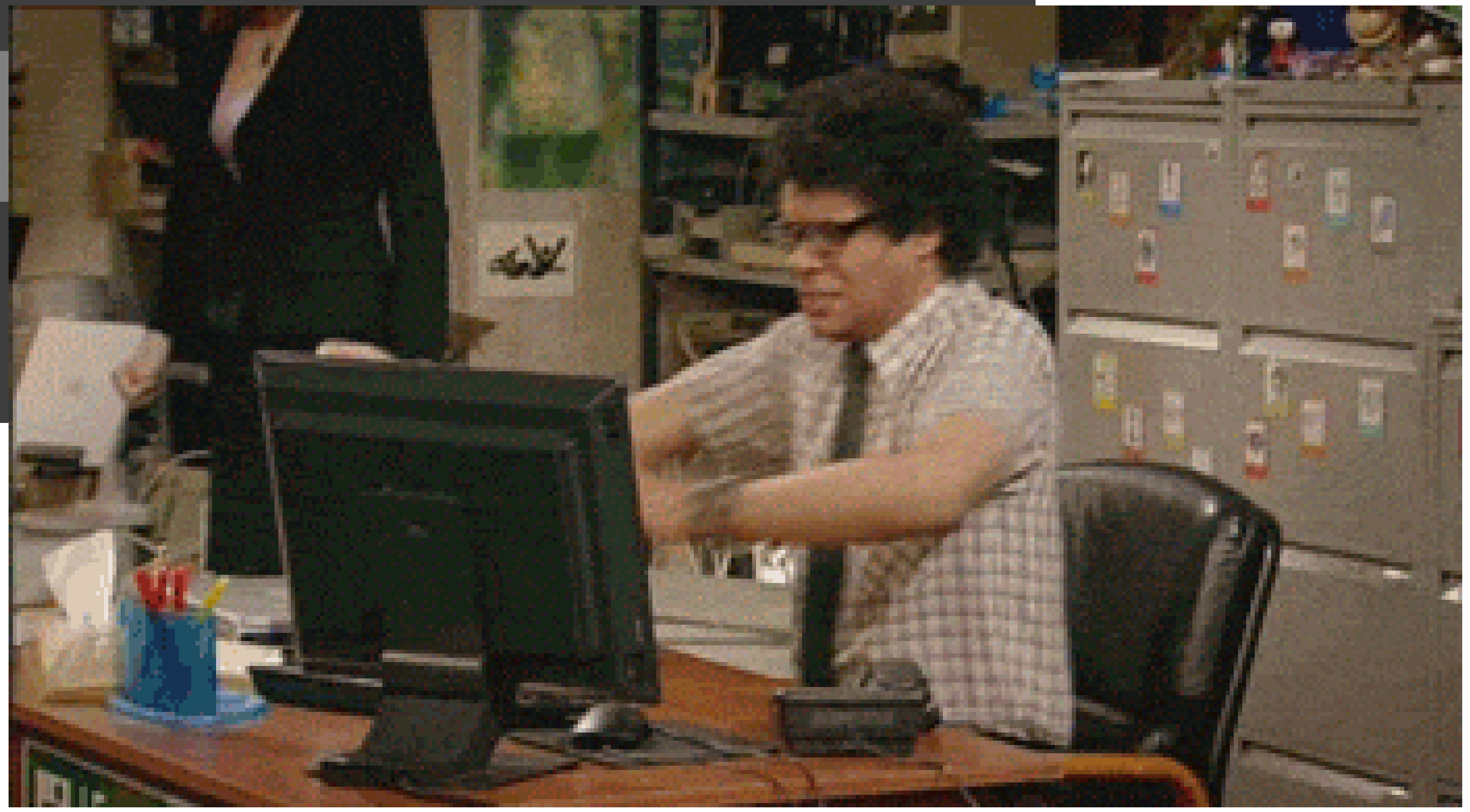


Parameter Groups

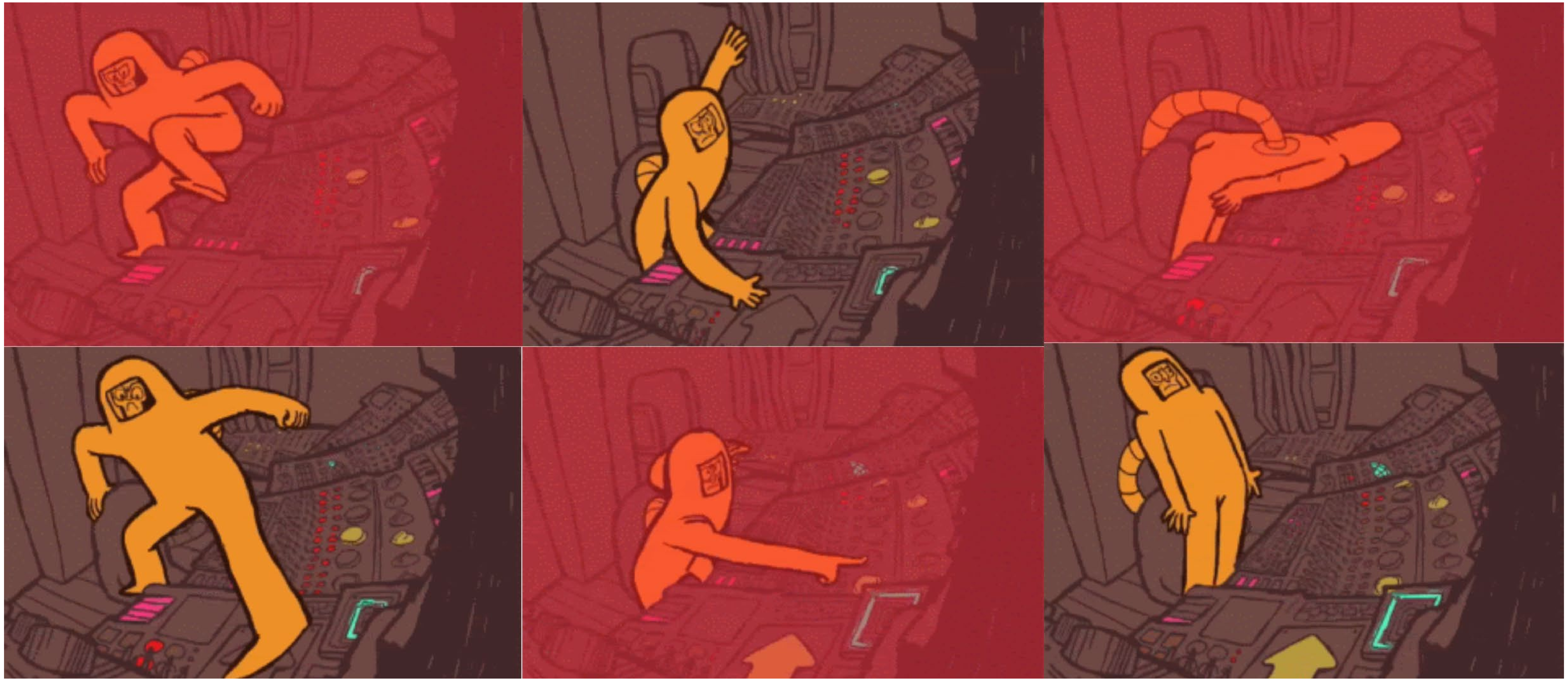
Scalar Parameter Values

□ Param 10.0

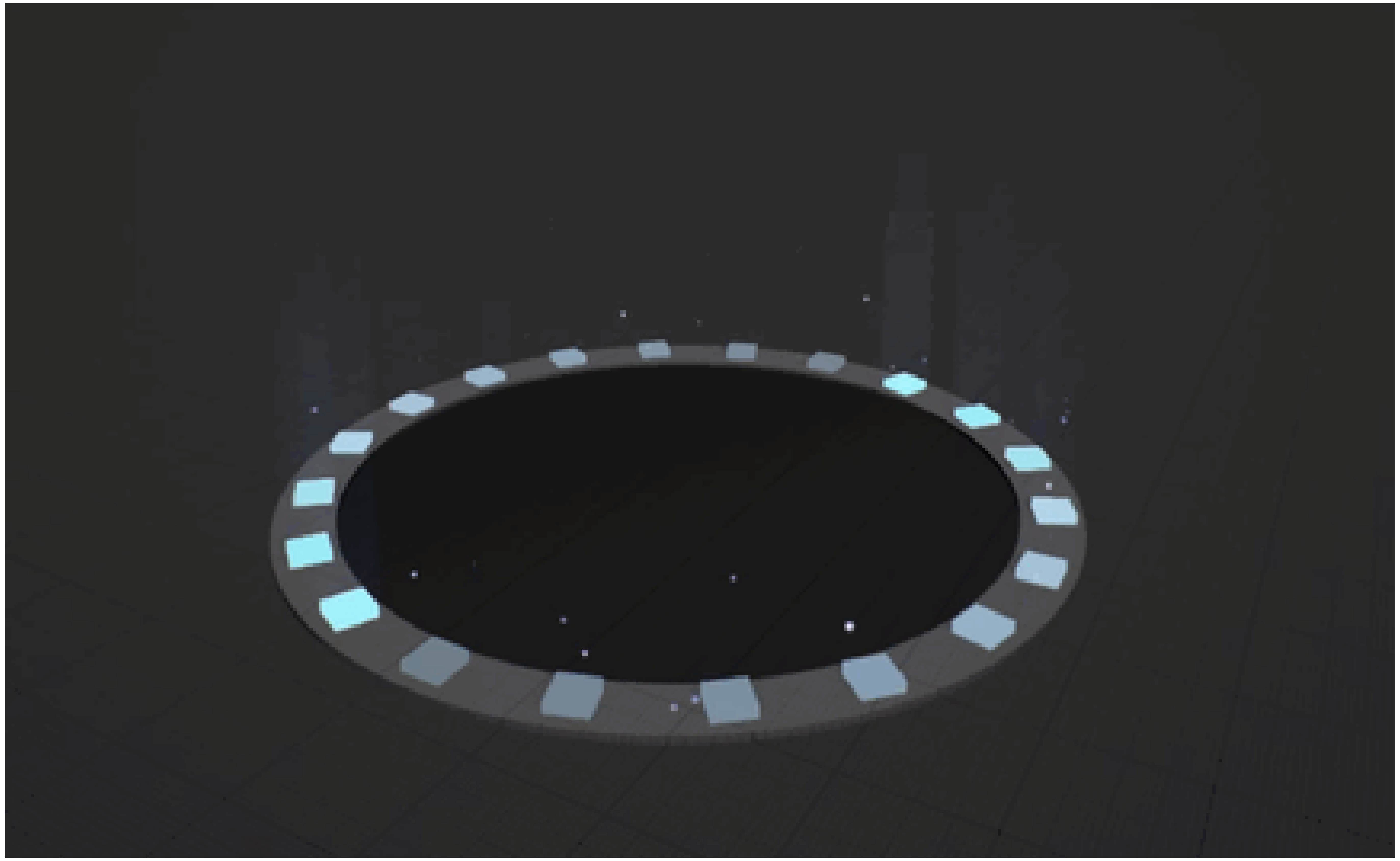
□ Param_1 0.2



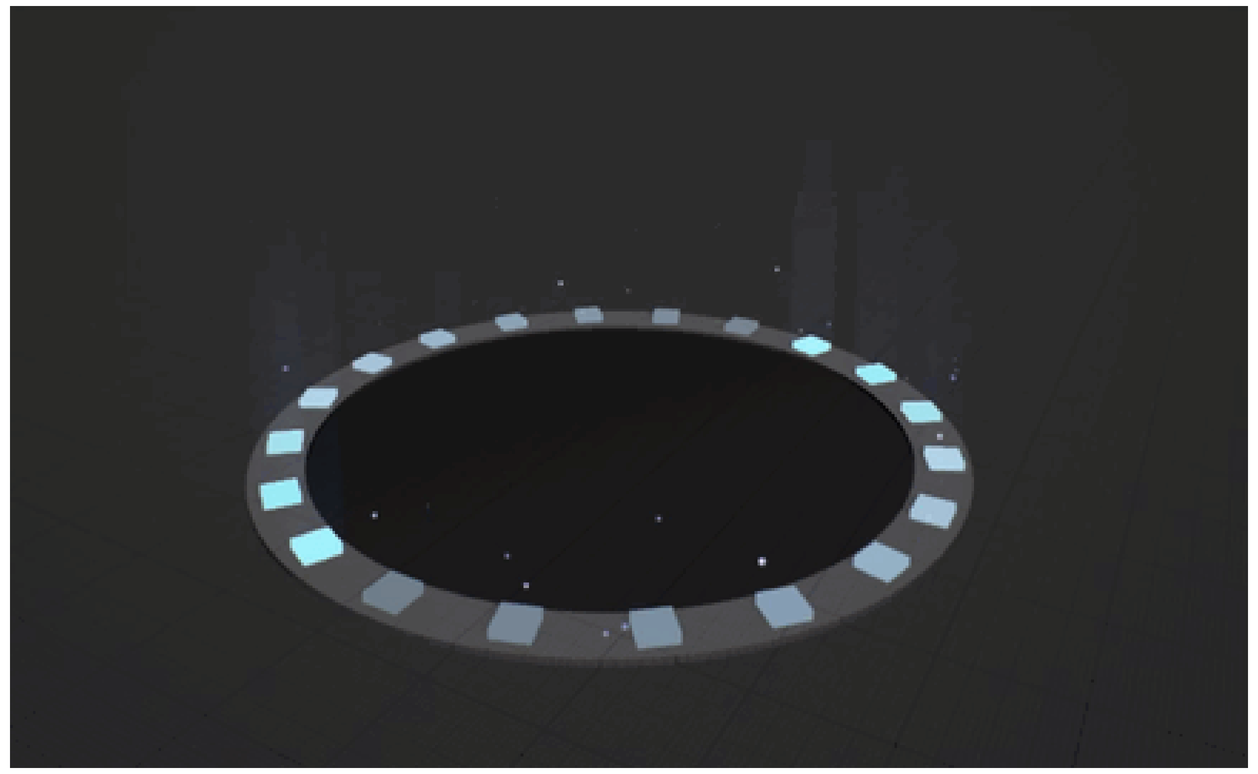
UNDERSTAND THE SYSTEM

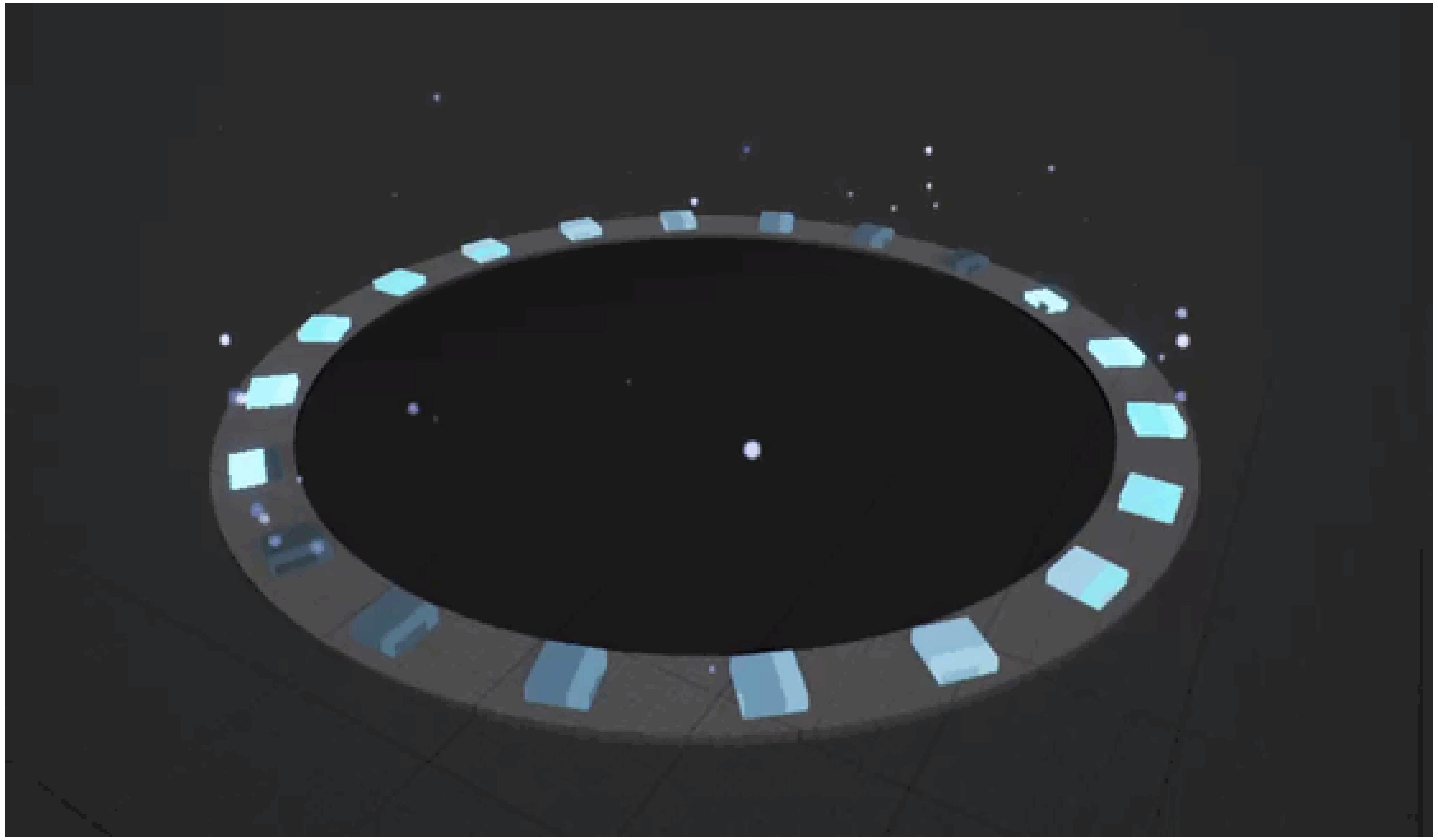


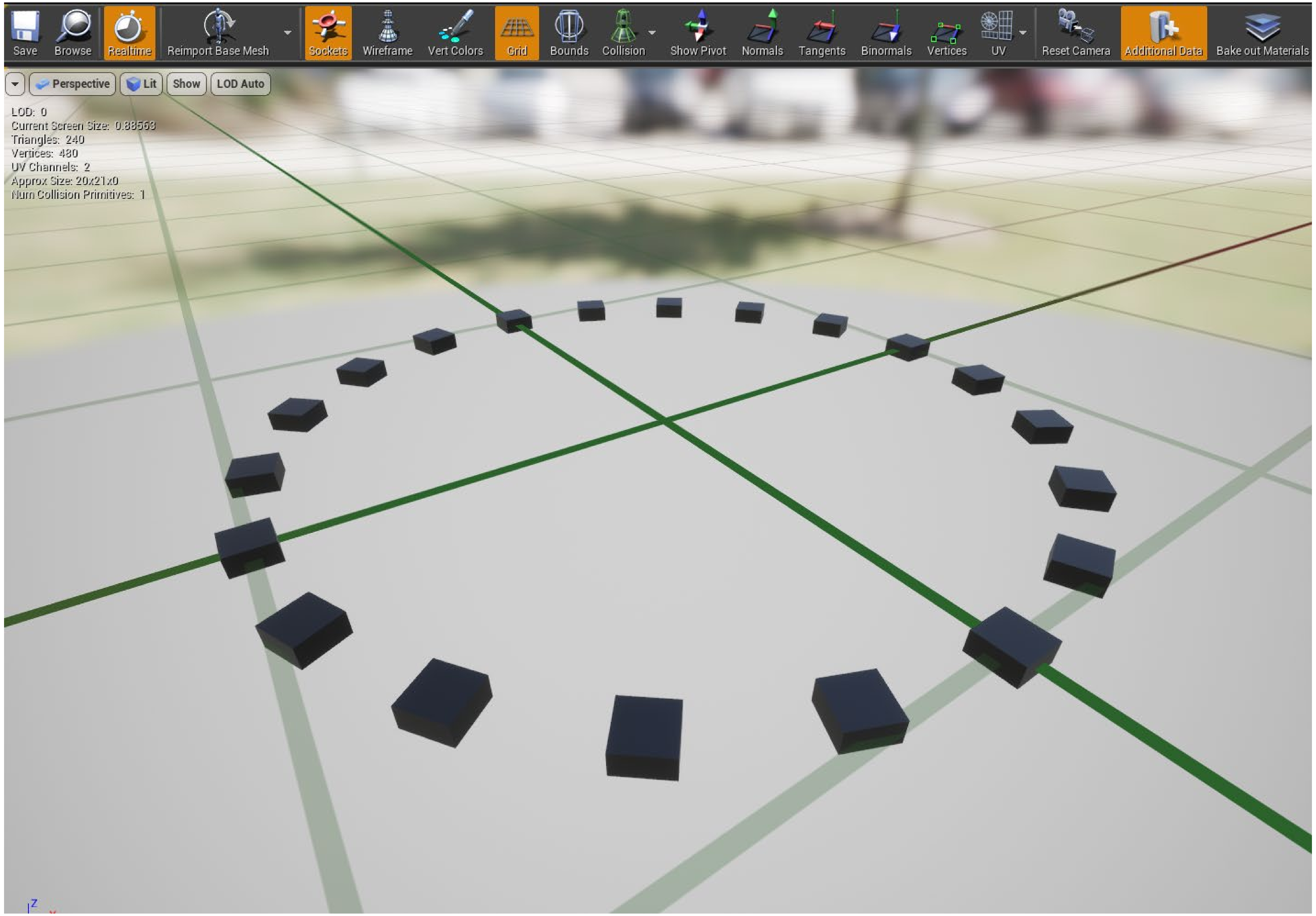
UNDERSTAND THE SYSTEM



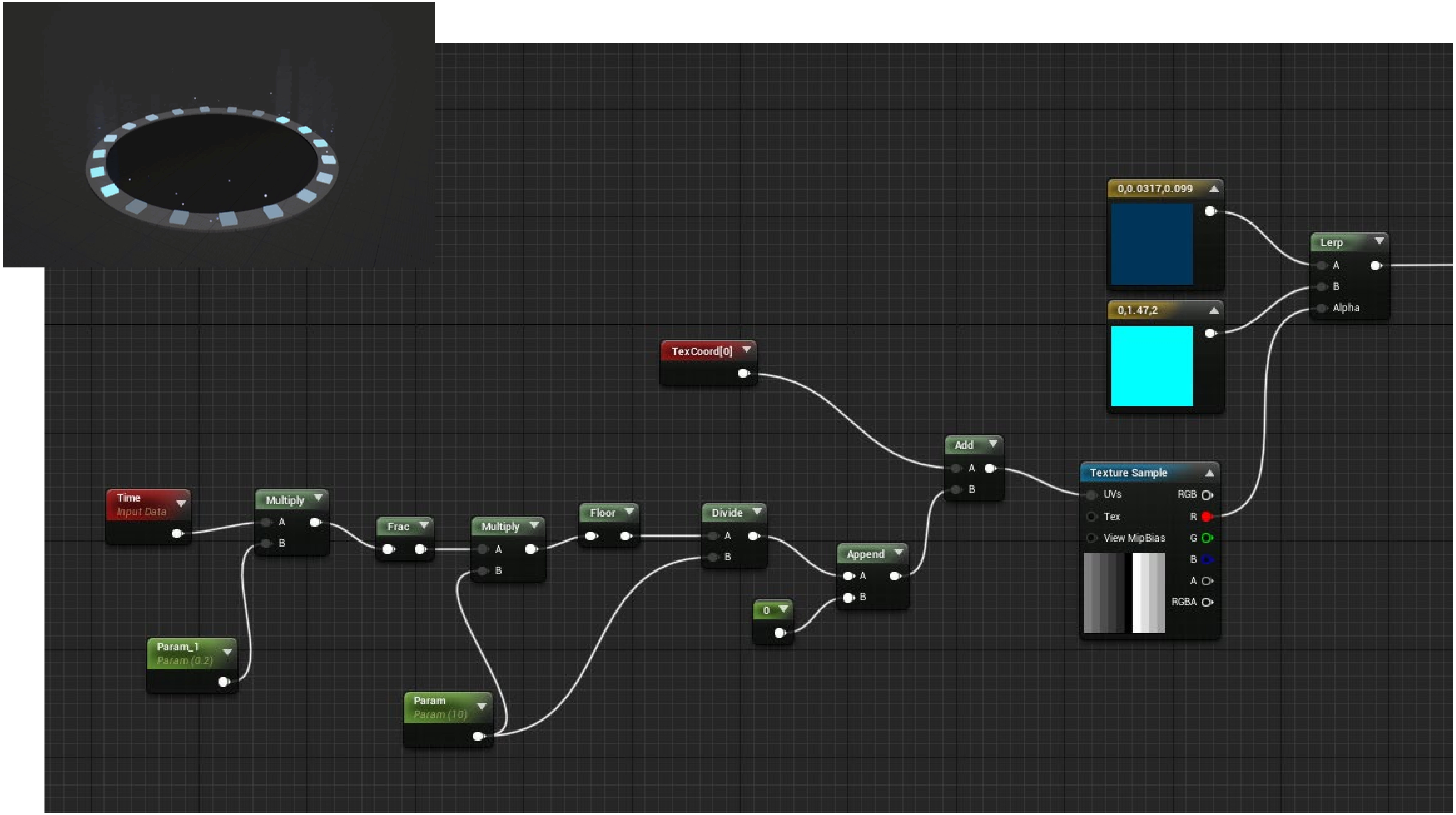
UNDERSTAND THE SYSTEM

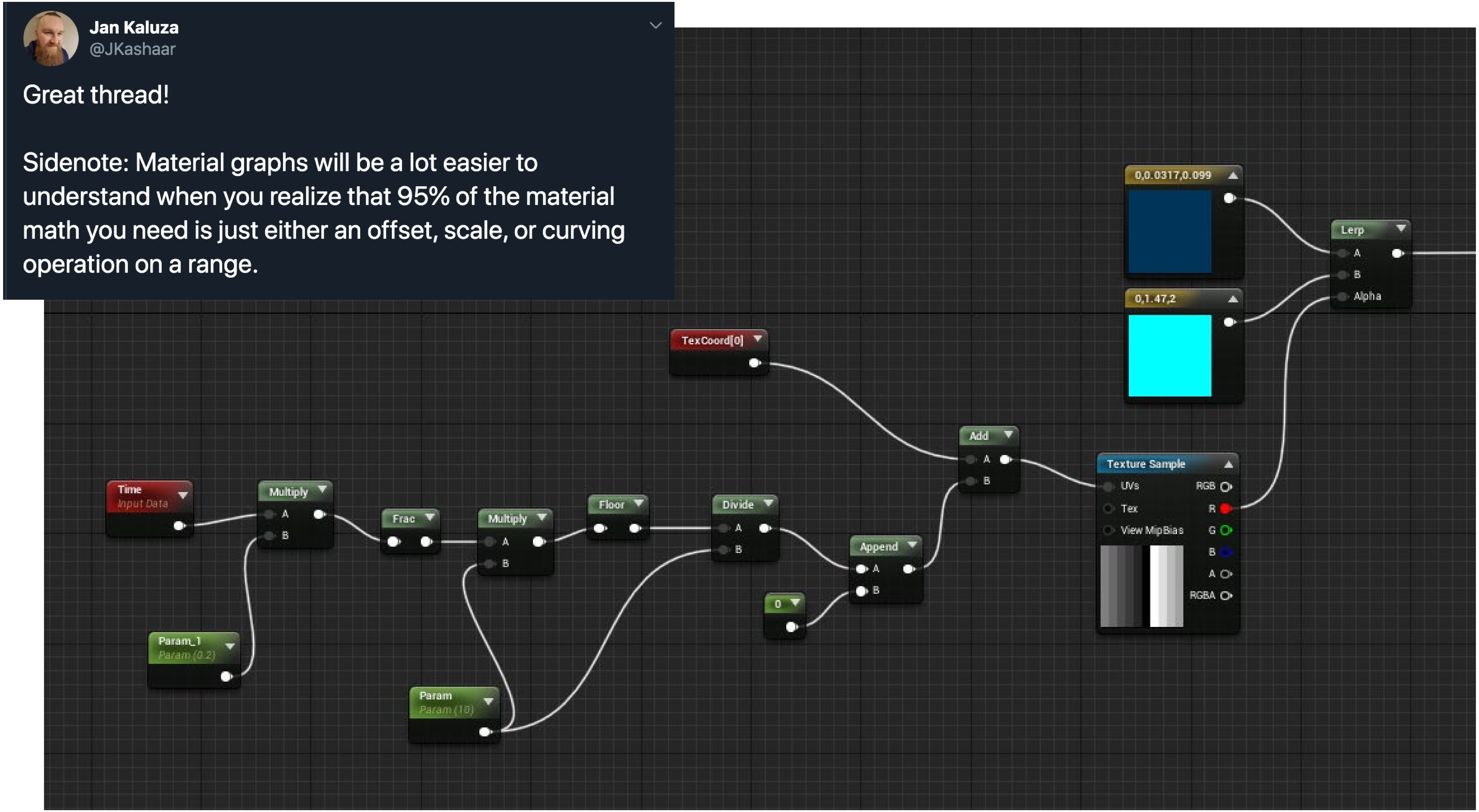






```
LOD:  0
Current Screen Size:  0.88563
Triangles:  240
Vertices:  480
UV Channels:  2
Approx Size: 20x21x0
Num Collision Primitives:  1
```

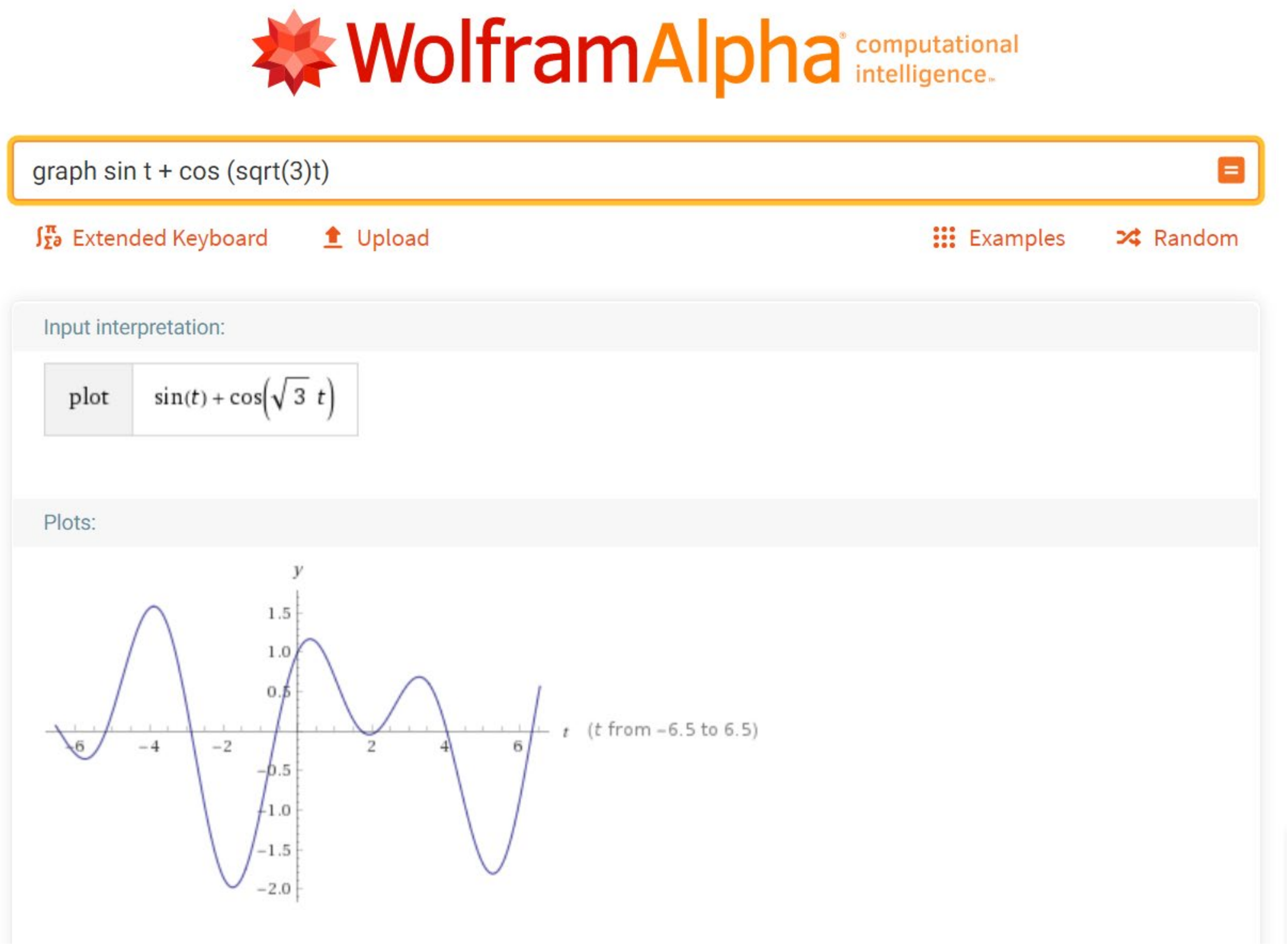

0,0.0317,0.099
0,1.47,2
Lerp
A
B
Alpha
TexCoord[0]
Add
A
B
Texture Sample
UVs
Tex
View MipBias
RGB
R
G
B
A
RGBA
Time
Input Data
Multiply
A
B
Frac
Multiply
A
B
Floor
Divide
A
B
Append
A
B
0
Param_1
Param (0.2)
Param
Param (10)



Jan Kaluza
@JKashaar

Great thread!

Sidenote: Material graphs will be a lot easier to understand when you realize that 95% of the material math you need is just either an offset, scale, or curving operation on a range.



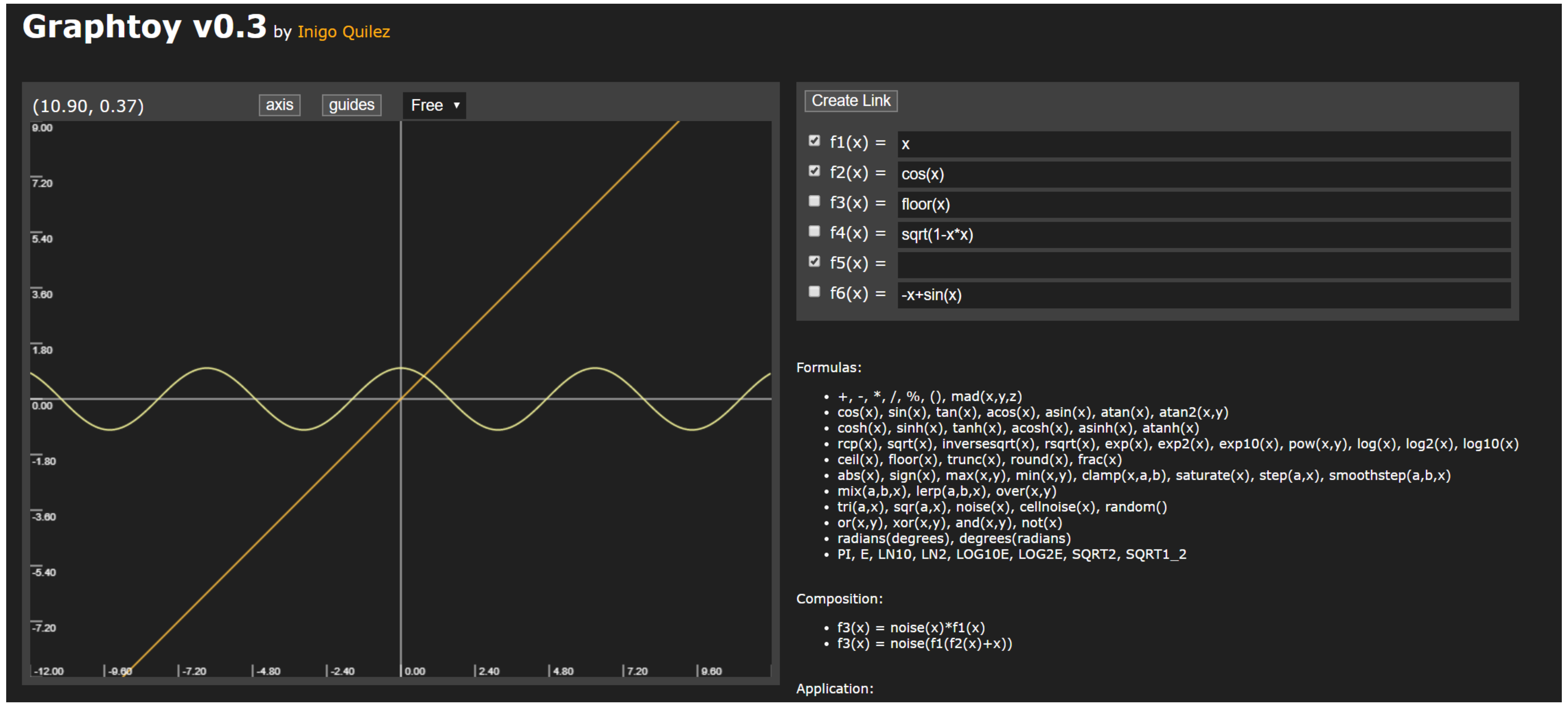
graph sin t + cos (sqrt(3)t)

Input interpretation:

plot $\sin(t) + \cos\left(\sqrt{3}\ t\right)$

Plots:

(t from −6.5 to 6.5)



Graphtoy v0.3 by Inigo Quilez

(10.90, 0.37)

☑ f1(x) = x
☑ f2(x) = cos(x)
□ f3(x) = floor(x)
□ f4(x) = sqrt(1-x*x)
☑ f5(x) =
□ f6(x) = -x+sin(x)

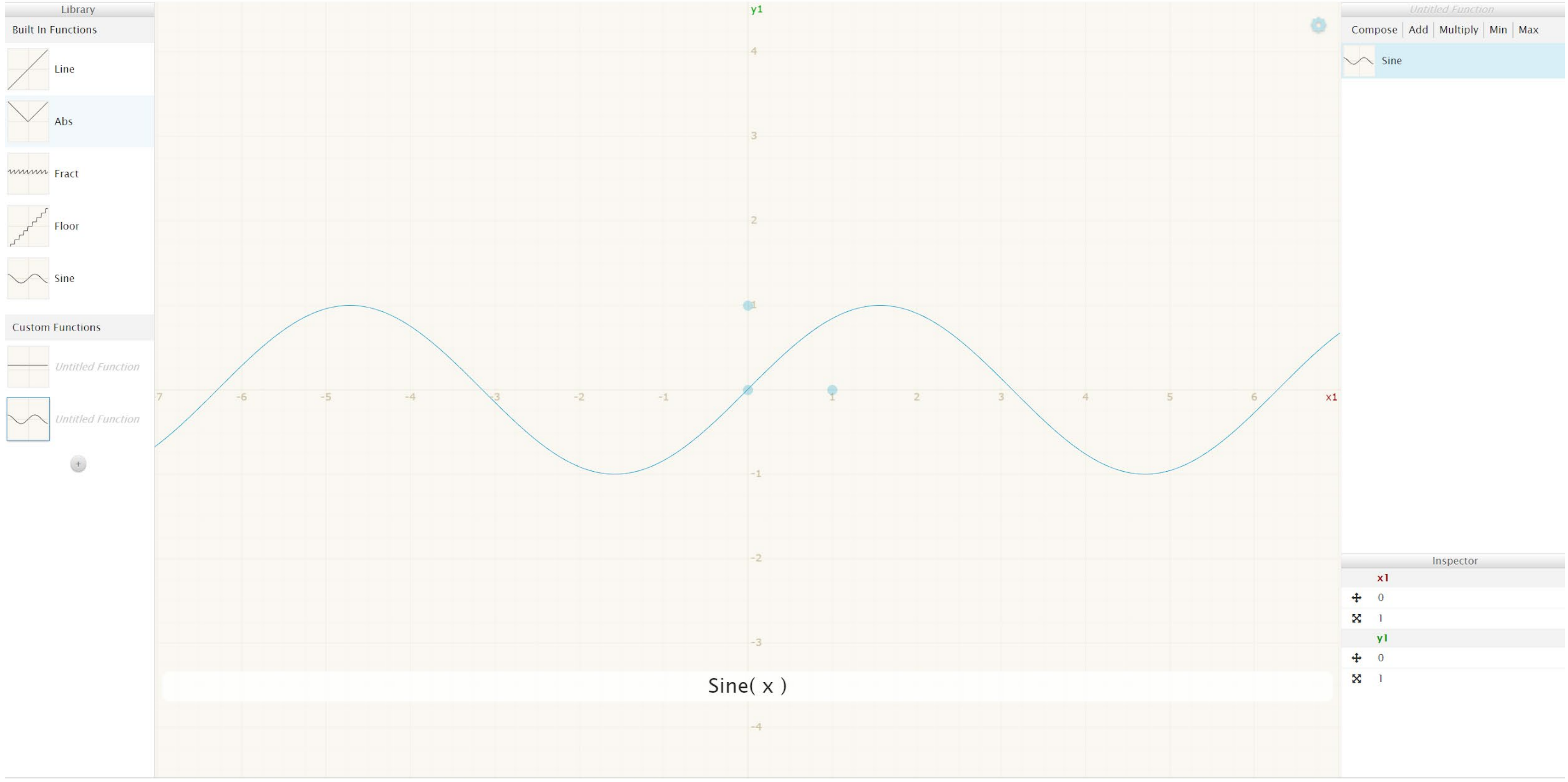
Formulas:

- +, -, *, /, %, (), mad(x,y,z)
- cos(x), sin(x), tan(x), acos(x), asin(x), atan(x), atan2(x,y)
- cosh(x), sinh(x), tanh(x), acosh(x), asinh(x), atanh(x)
- rcp(x), sqrt(x), inversesqrt(x), rsqrt(x), exp(x), exp2(x), exp10(x), pow(x,y), log(x), log2(x), log10(x)
- ceil(x), floor(x), trunc(x), round(x), frac(x)
- abs(x), sign(x), max(x,y), min(x,y), clamp(x,a,b), saturate(x), step(a,x), smoothstep(a,b,x)
- mix(a,b,x), lerp(a,b,x), over(x,y)
- tri(a,x), sqr(a,x), noise(x), cellnoise(x), random()
- or(x,y), xor(x,y), and(x,y), not(x)
- radians(degrees), degrees(radians)
- PI, E, LN10, LN2, LOG10E, LOG2E, SQRT2, SQRT1_2

Composition:

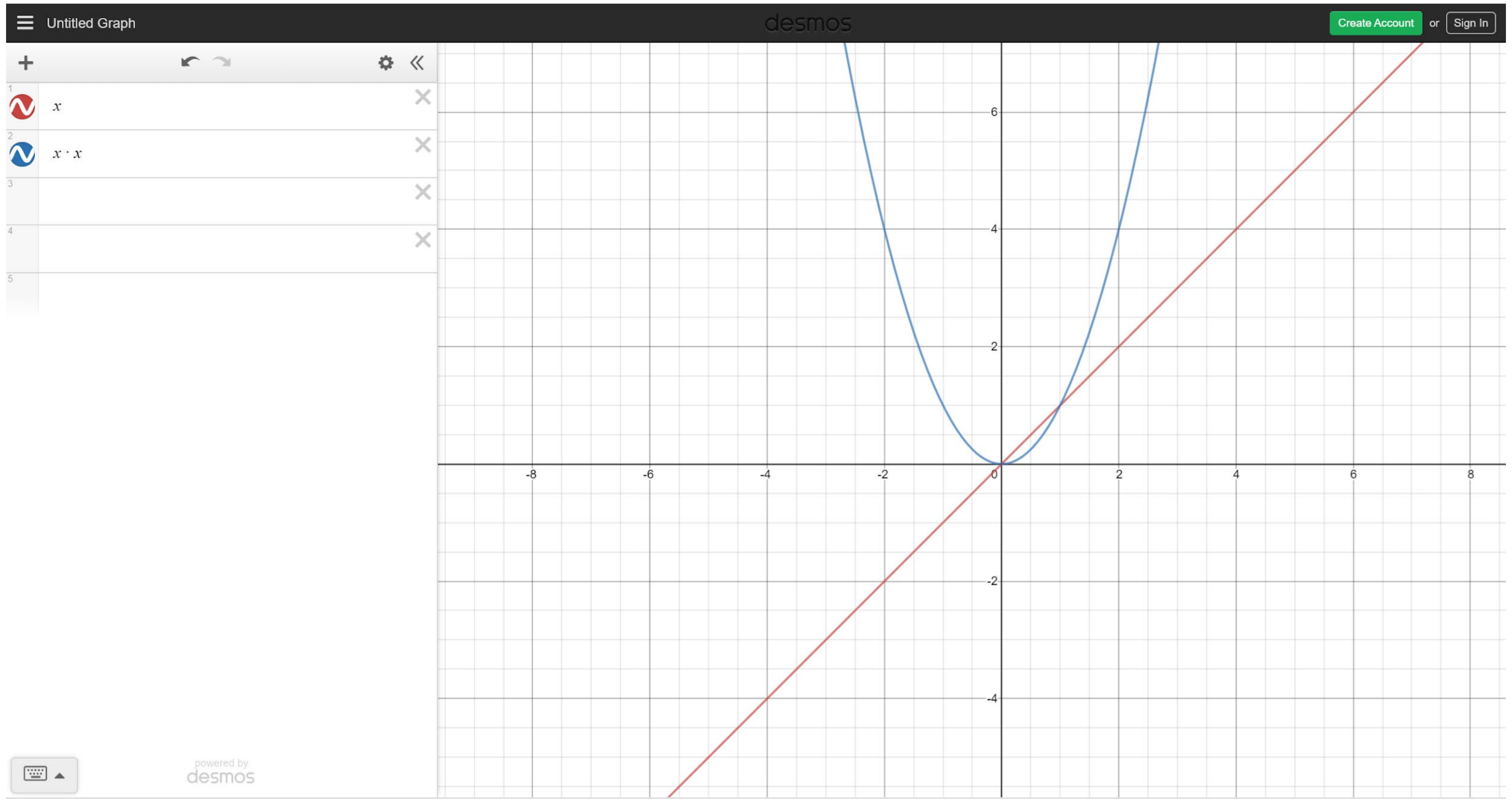
- f3(x) = noise(x)*f1(x)
- f3(x) = noise(f1(f2(x)+x))

Application:



x1
✥ 0
⤧ 1
y1
✥ 0
⤧ 1

Sine(x)



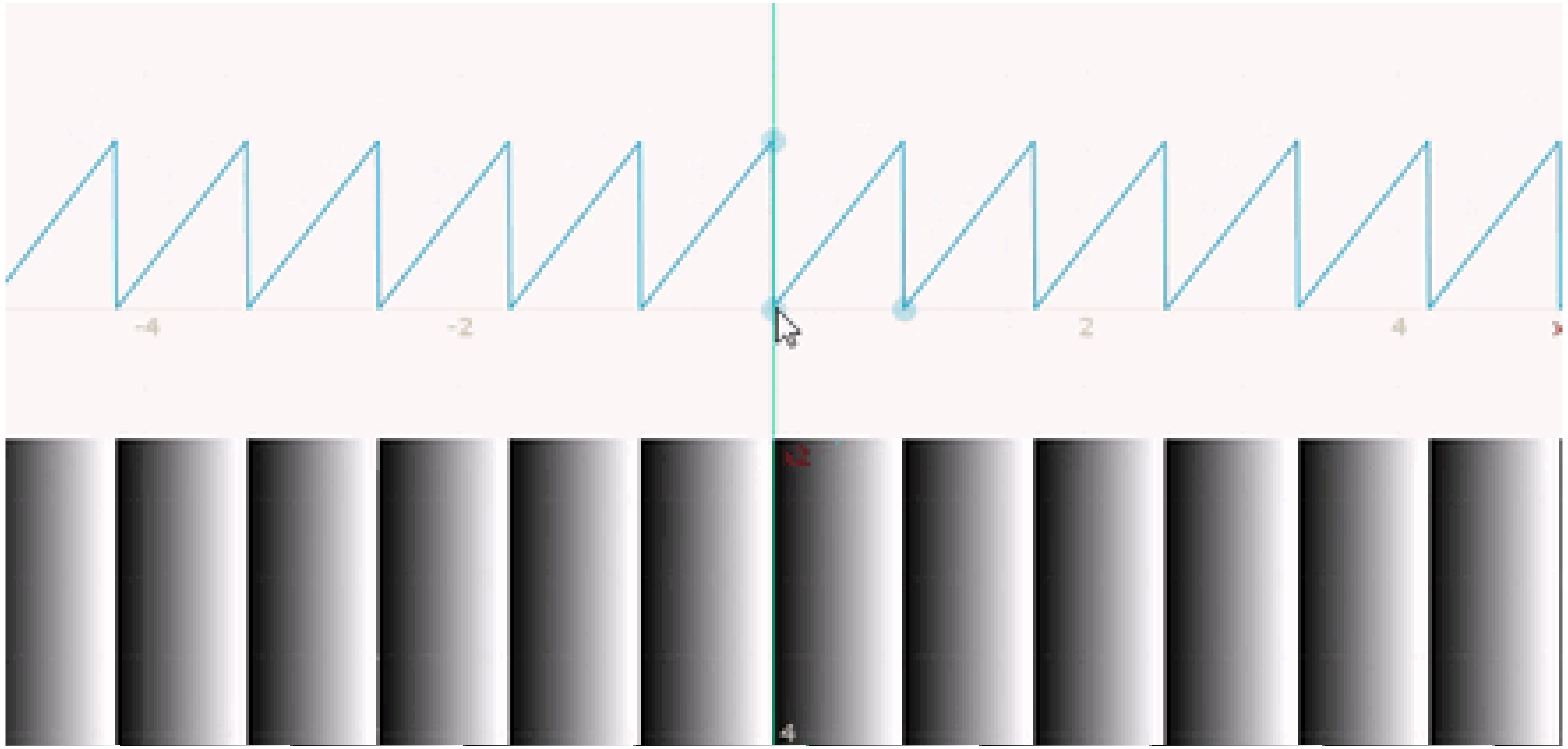
1 x

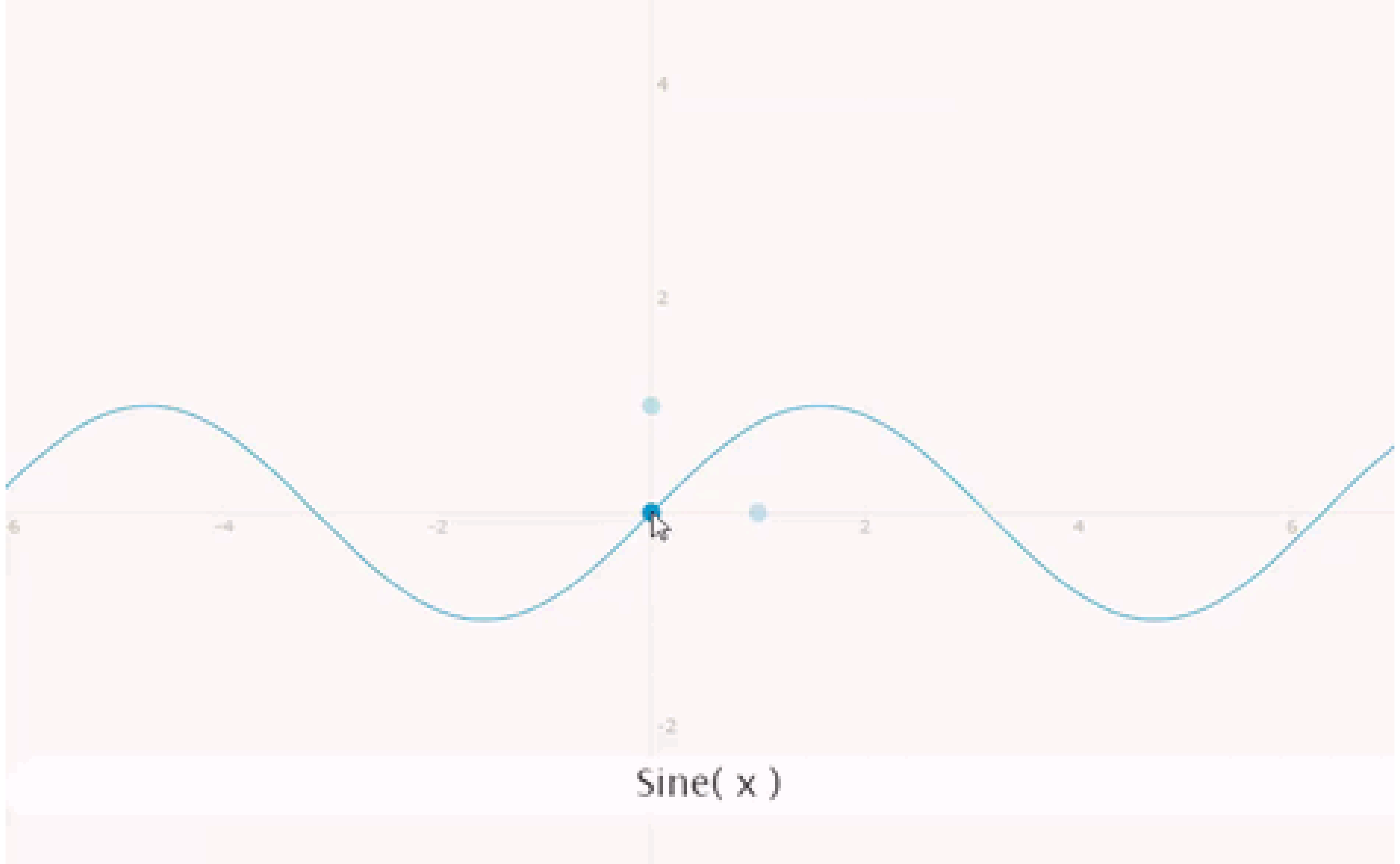
2 $x \cdot x$

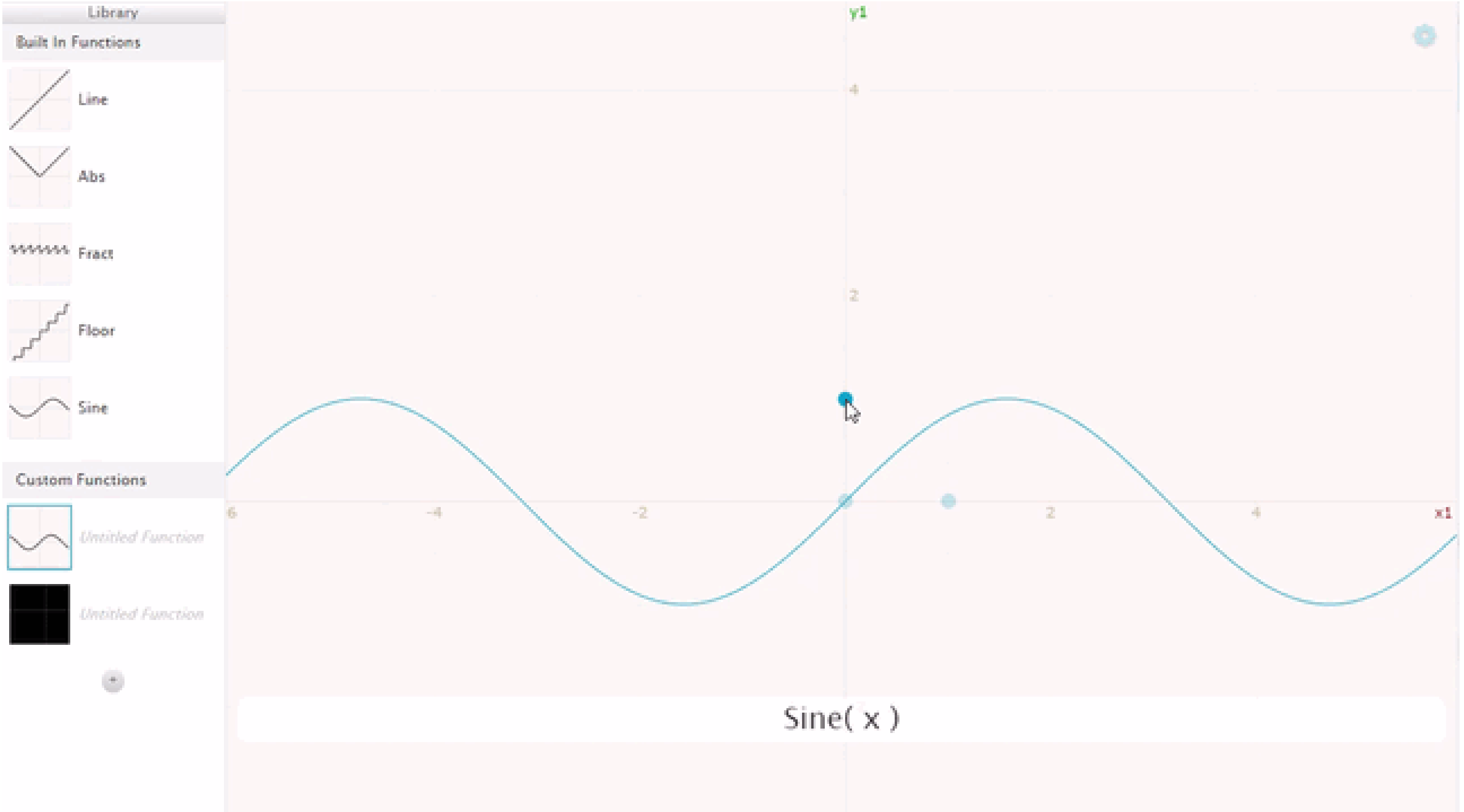
3

4

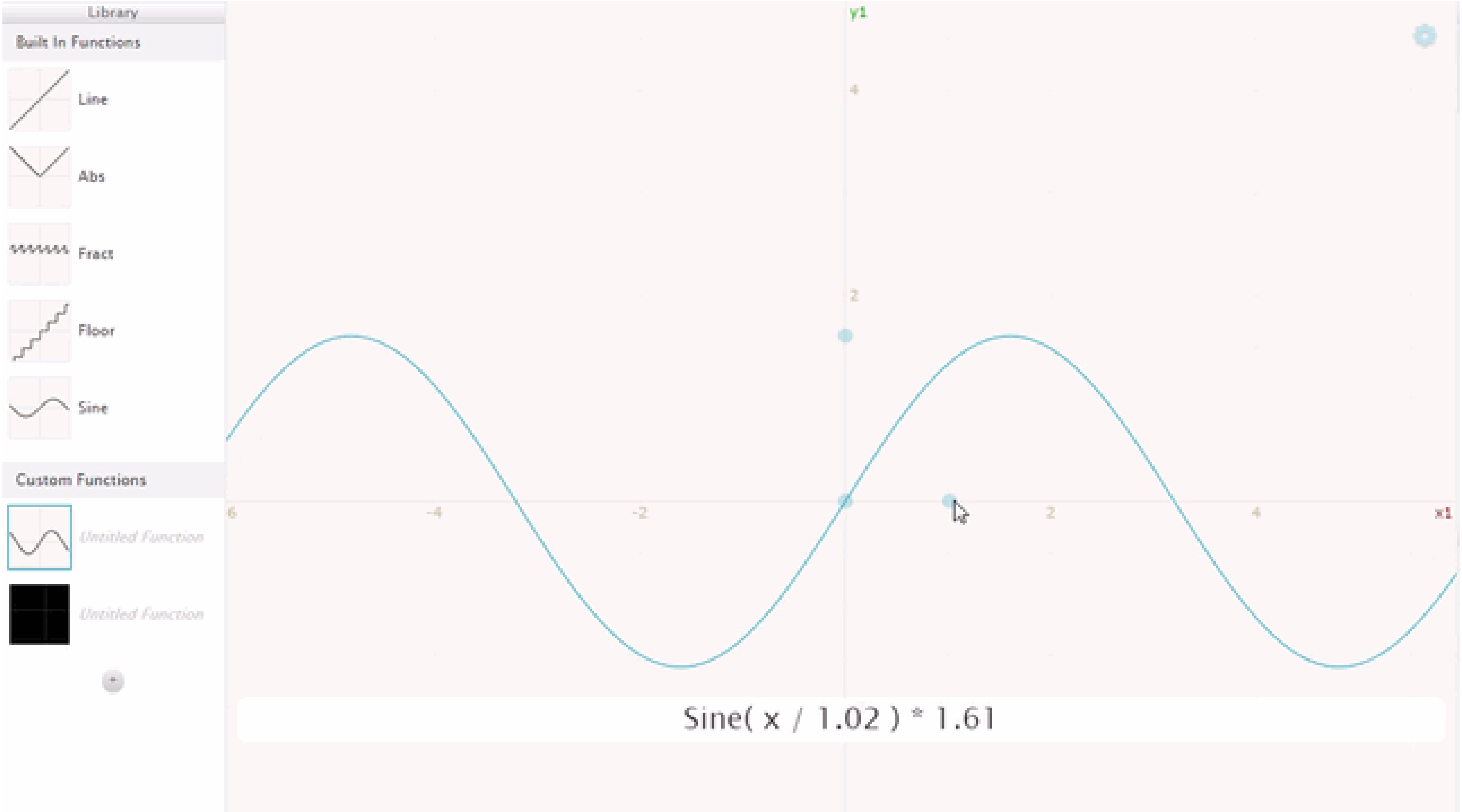
5


-4
-2
2
4
x
4

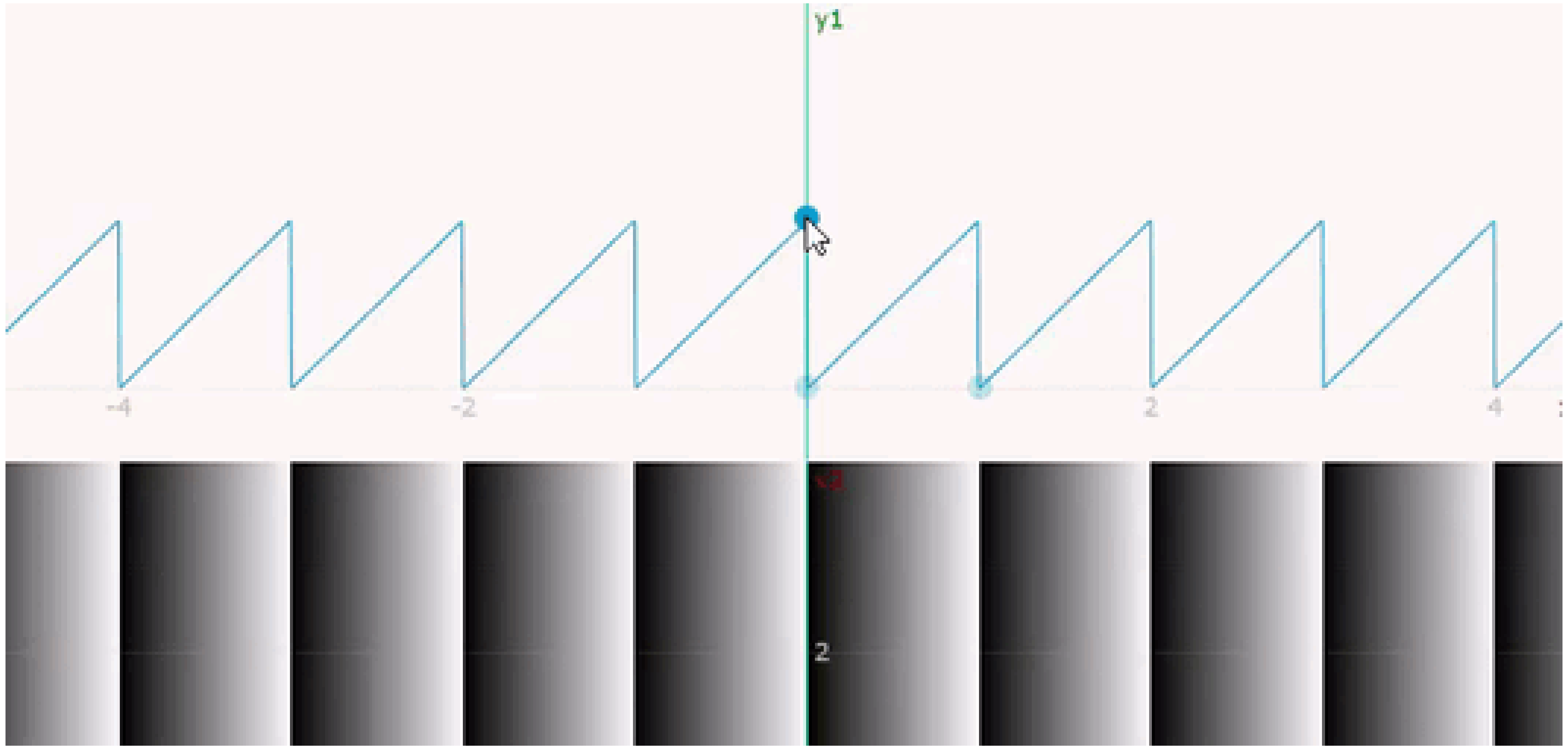

4
2
-2
-6
-4
-2
2
4
6
Sine(x)

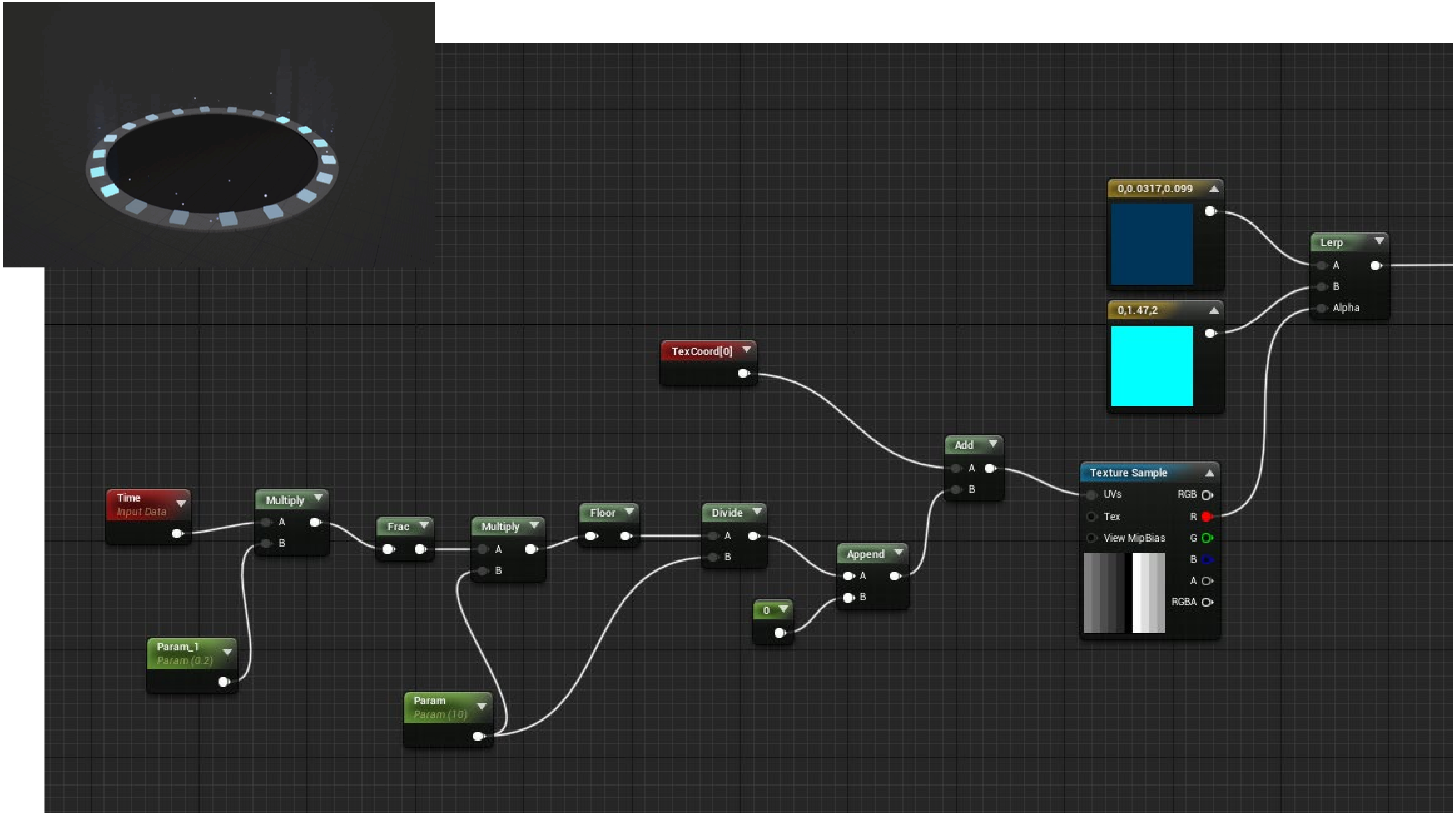


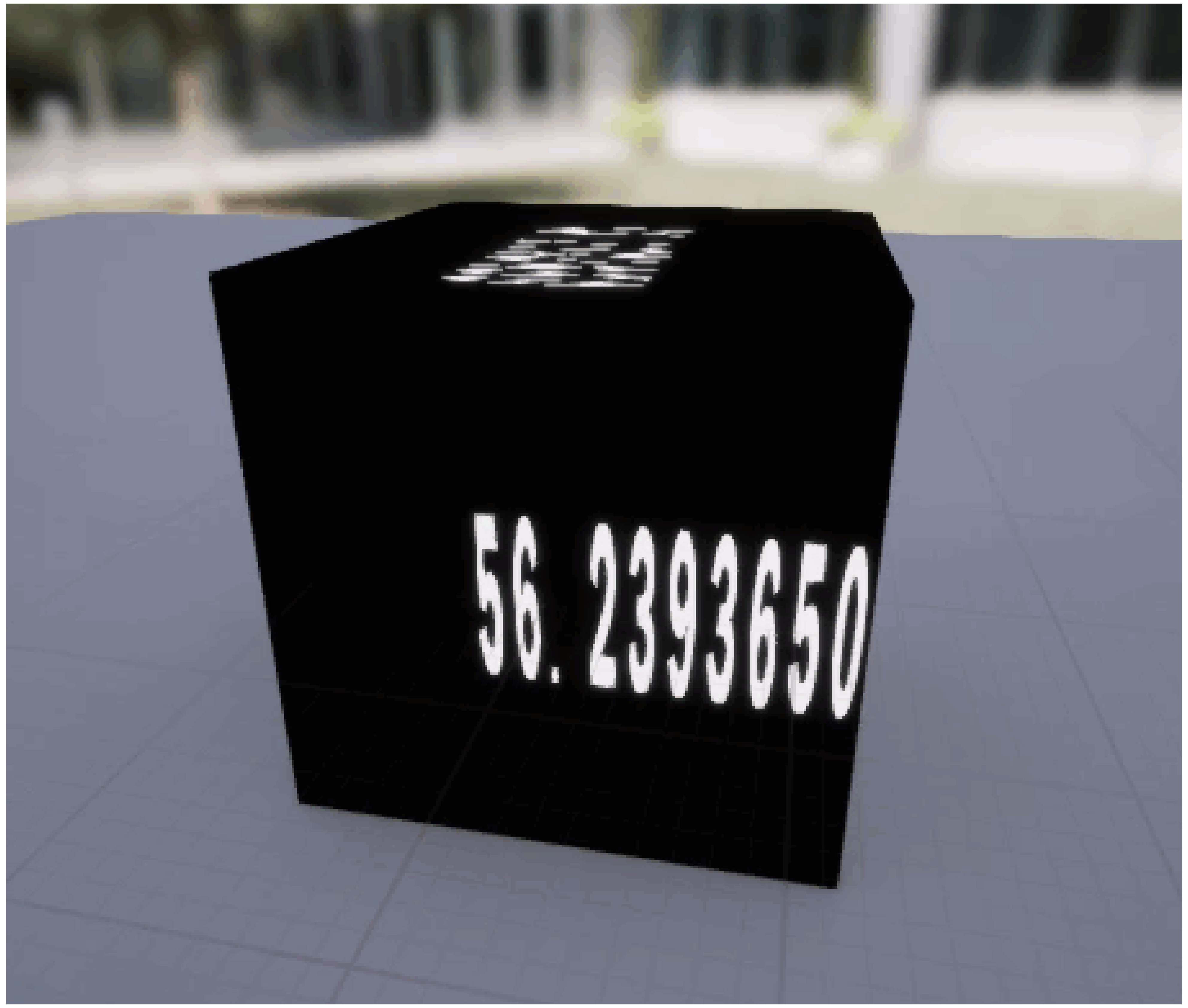
Sine(x)

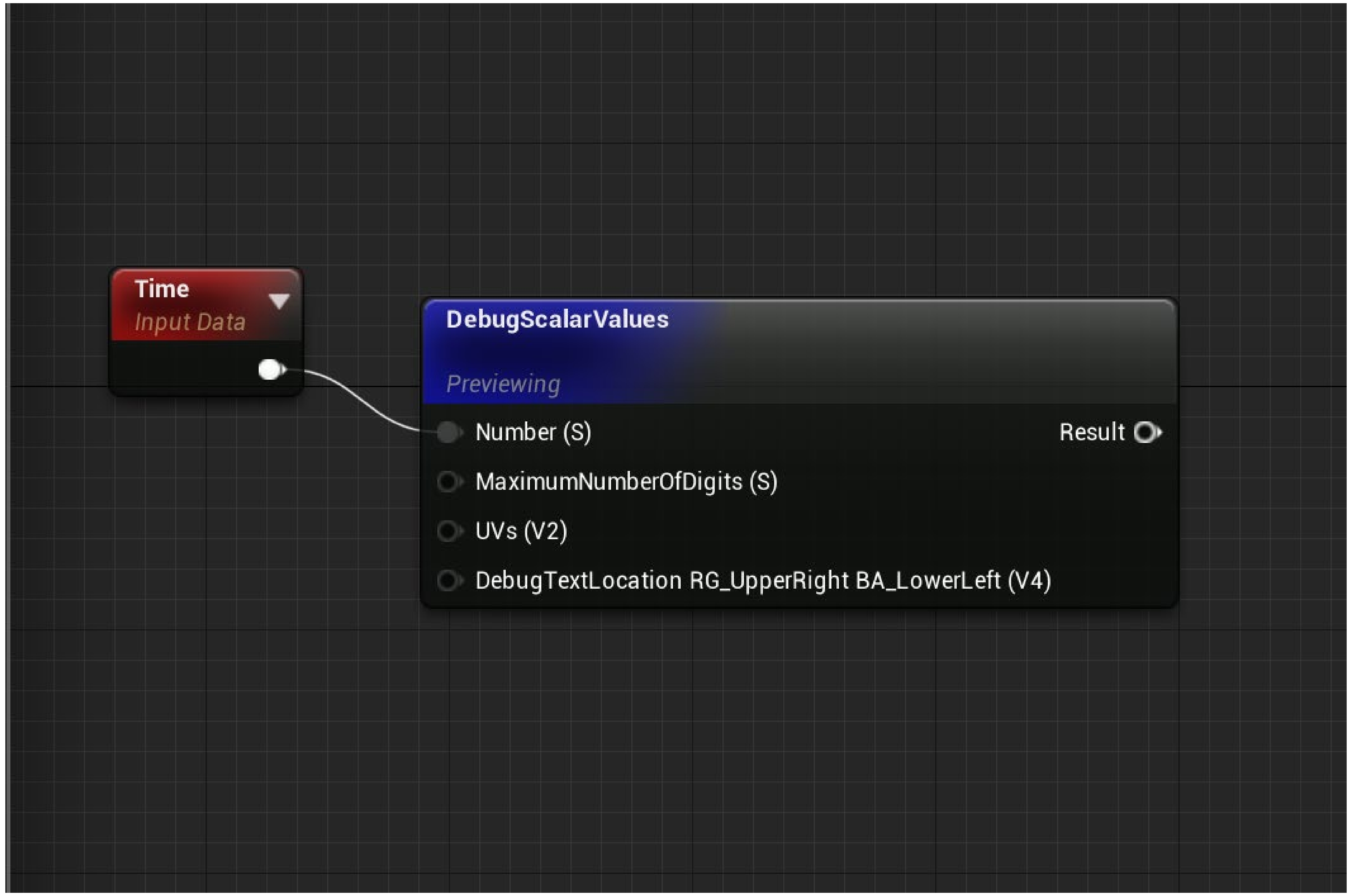


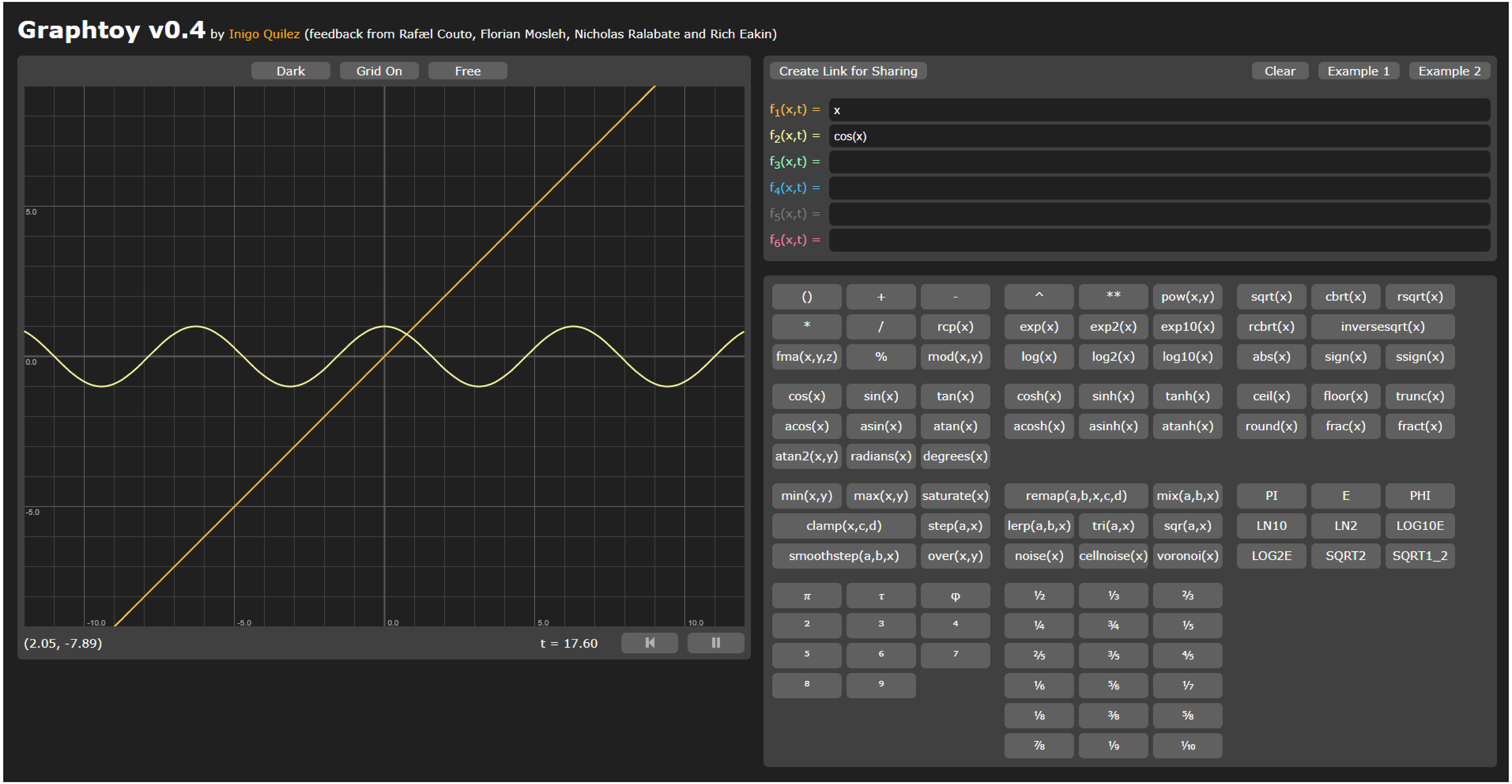
Sine(x / 1.02) * 1.61


y1
-4
-2
2
4
2


0,0.0317,0.099
0,1.47,2
Lerp
A
B
Alpha
TexCoord[0]
Add
A
B
Texture Sample
UVs
Tex
View MipBias
RGB
R
G
B
A
RGBA
Time
Input Data
Multiply
A
B
Frac
Multiply
A
B
Floor
Divide
A
B
Append
A
B
0
Param_1
Param (0.2)
Param
Param (10)


56.2393650


Time
Input Data
DebugScalarValues
Previewing
Number (S)
Result
MaximumNumberOfDigits (S)
UVs (V2)
DebugTextLocation RG_UpperRight BA_LowerLeft (V4)



Graphtoy v0.4 by Inigo Quilez (feedback from Rafæl Couto, Florian Mosleh, Nicholas Ralabate and Rich Eakin)

$f_1(x,t)$ = x

$f_2(x,t)$ = cos(x)

$f_3(x,t)$ =

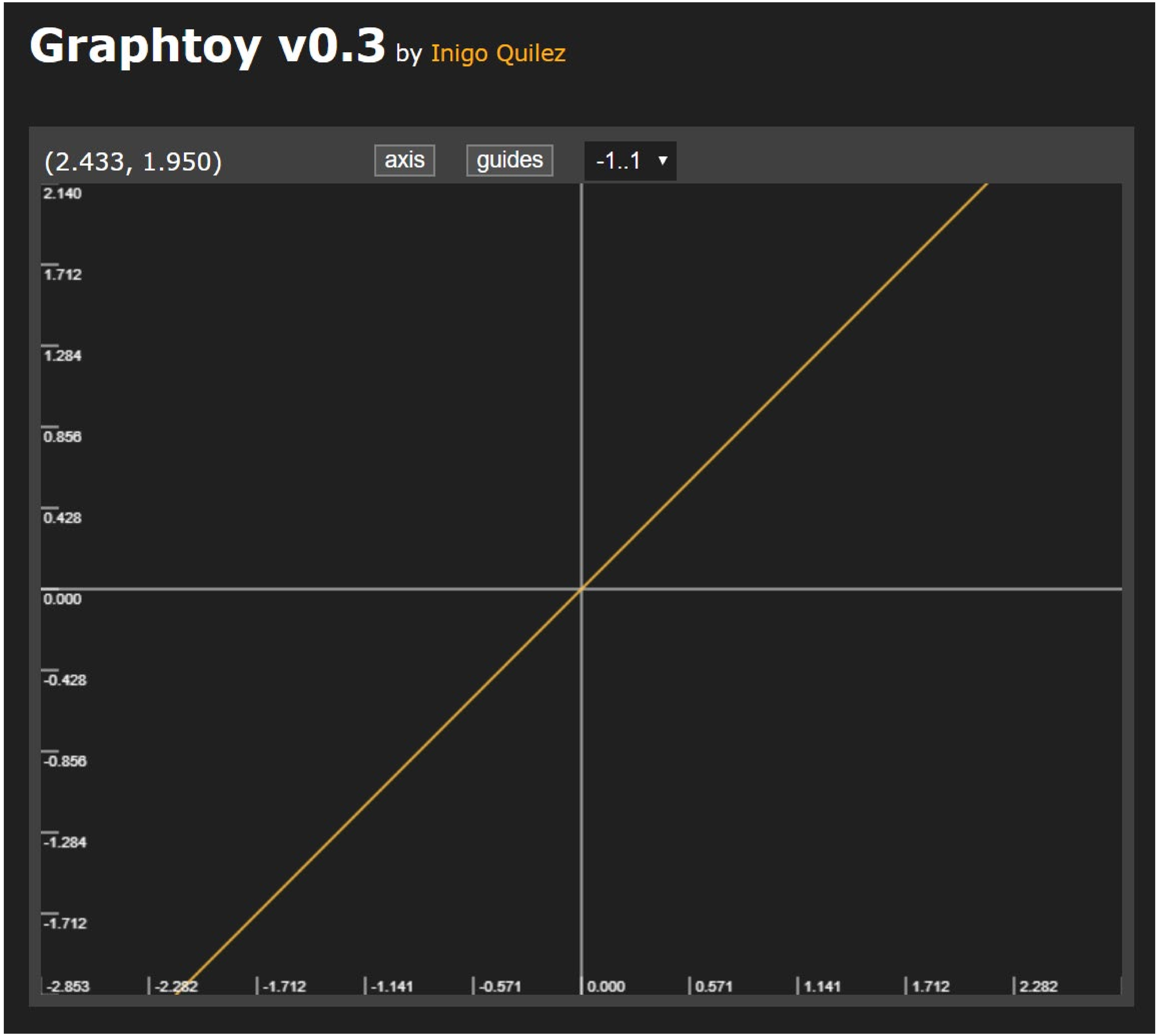
$f_4(x,t)$ =

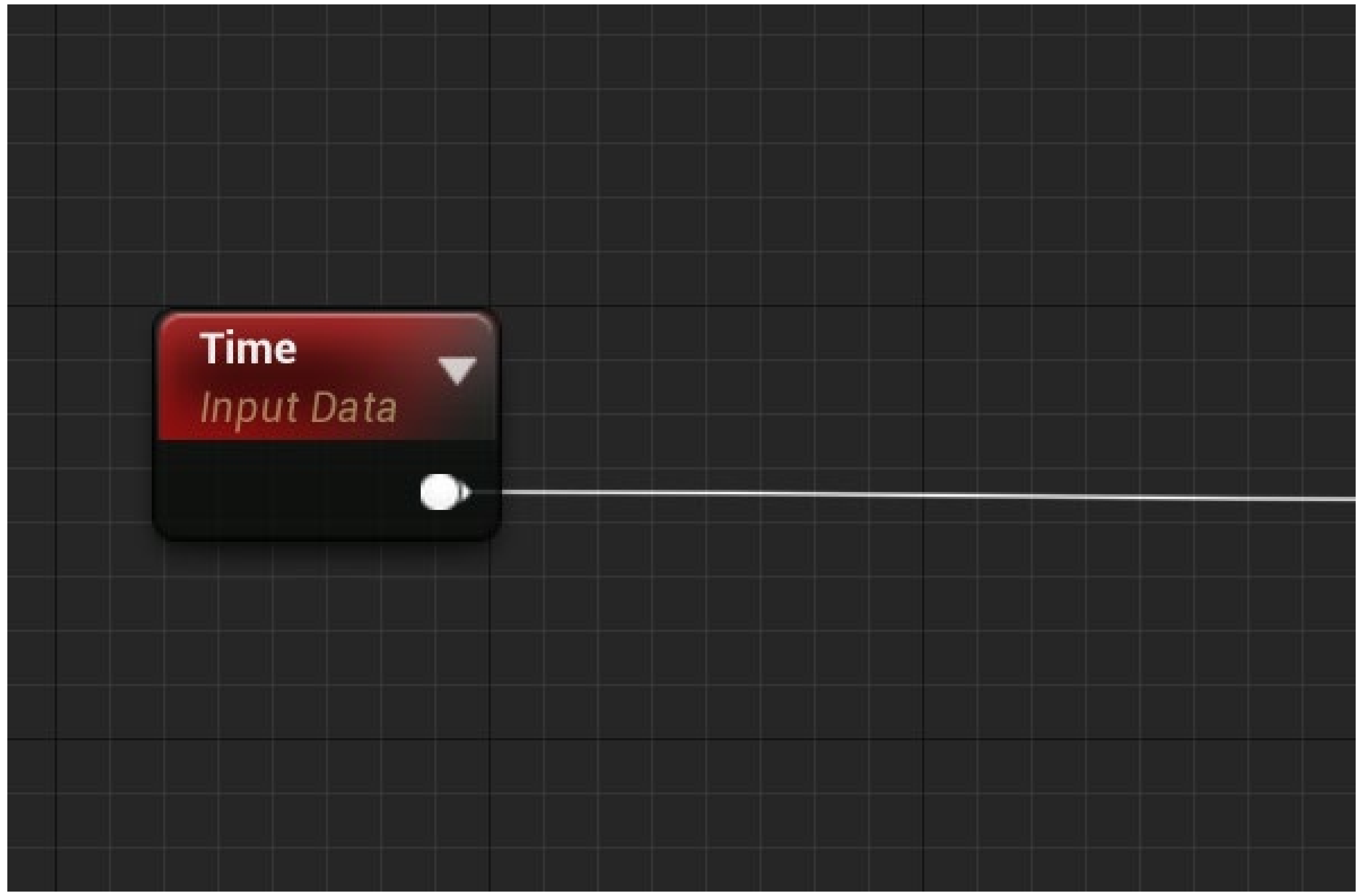
$f_5(x,t)$ =

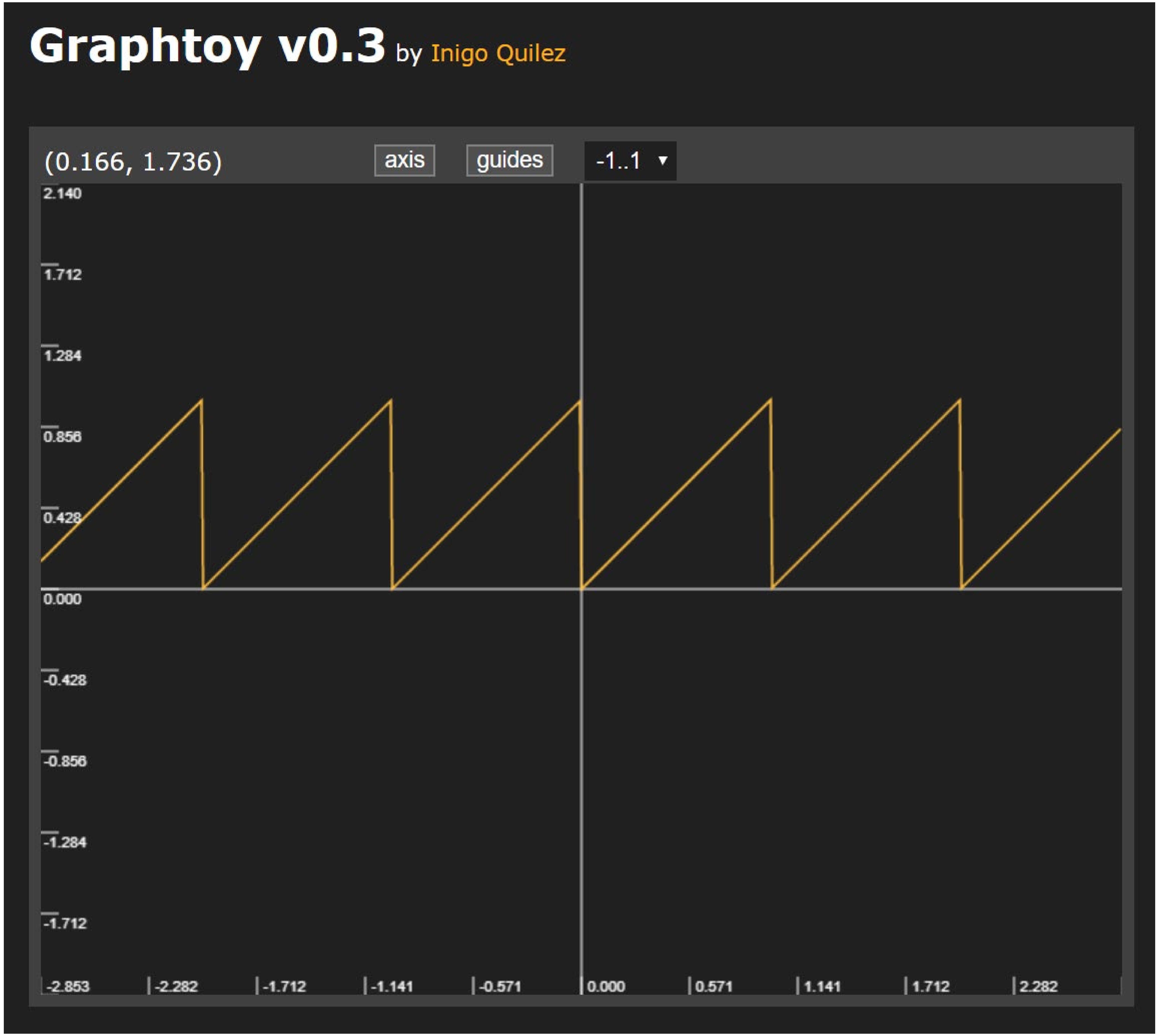
$f_6(x,t)$ =

(2.05, -7.89)

t = 17.60

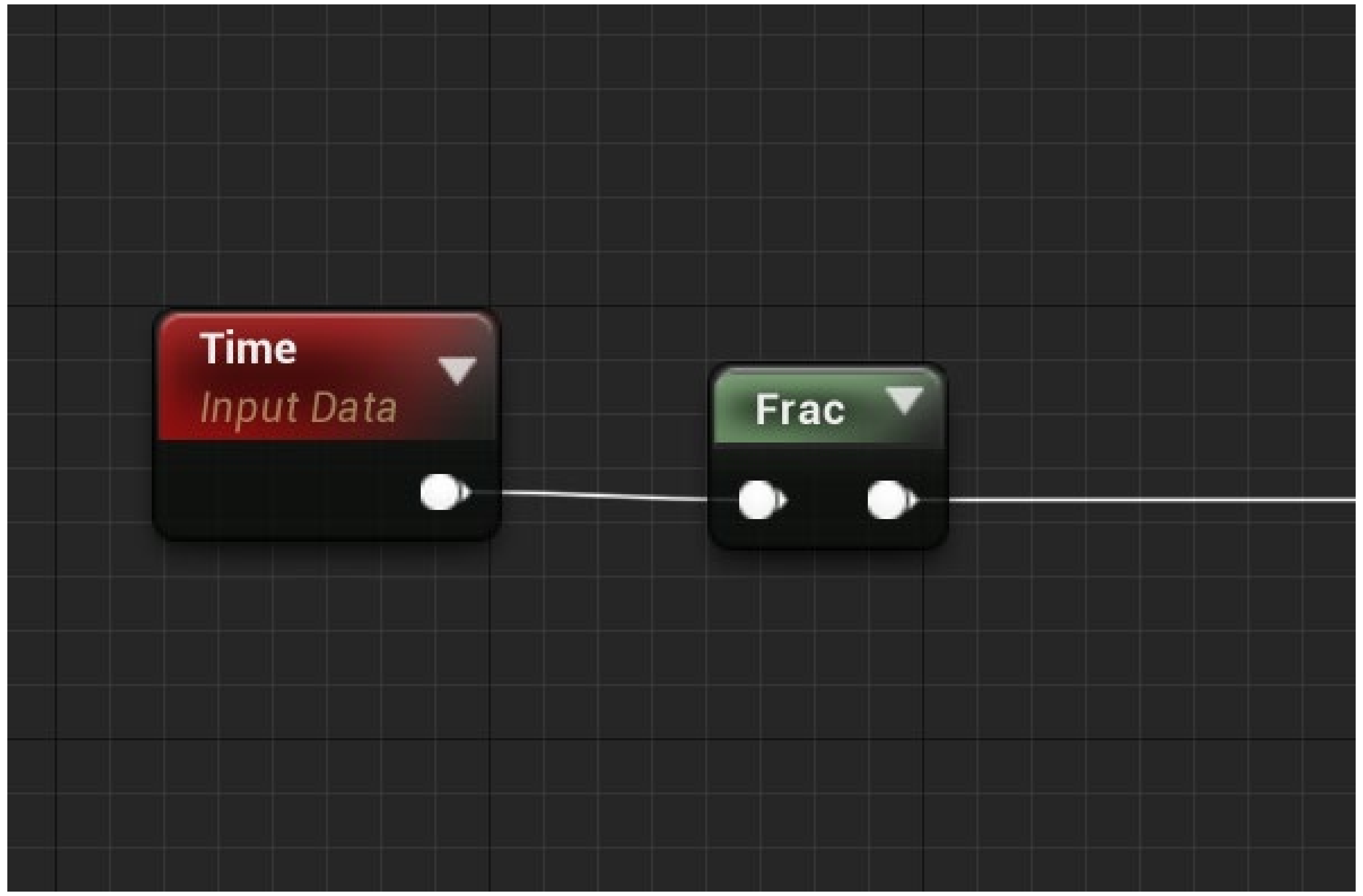

Graphtoy v0.3 by Inigo Quilez
(2.433, 1.950)
axis
guides
-1..1
2.140
1.712
1.284
0.856
0.428
0.000
-0.428
-0.856
-1.284
-1.712
-2.853
-2.282
-1.712
-1.141
-0.571
0.000
0.571
1.141
1.712
2.282

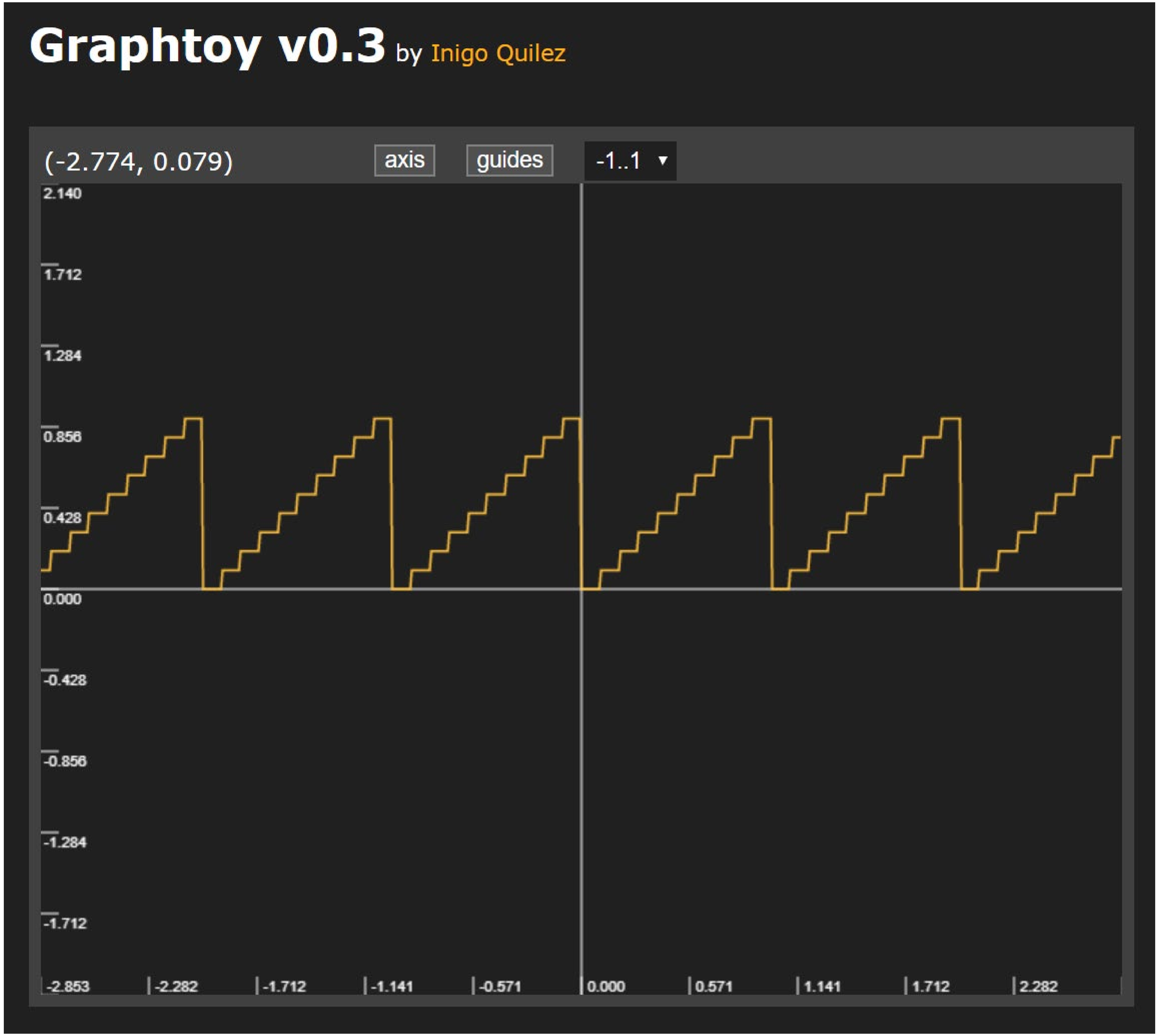

Time
Input Data



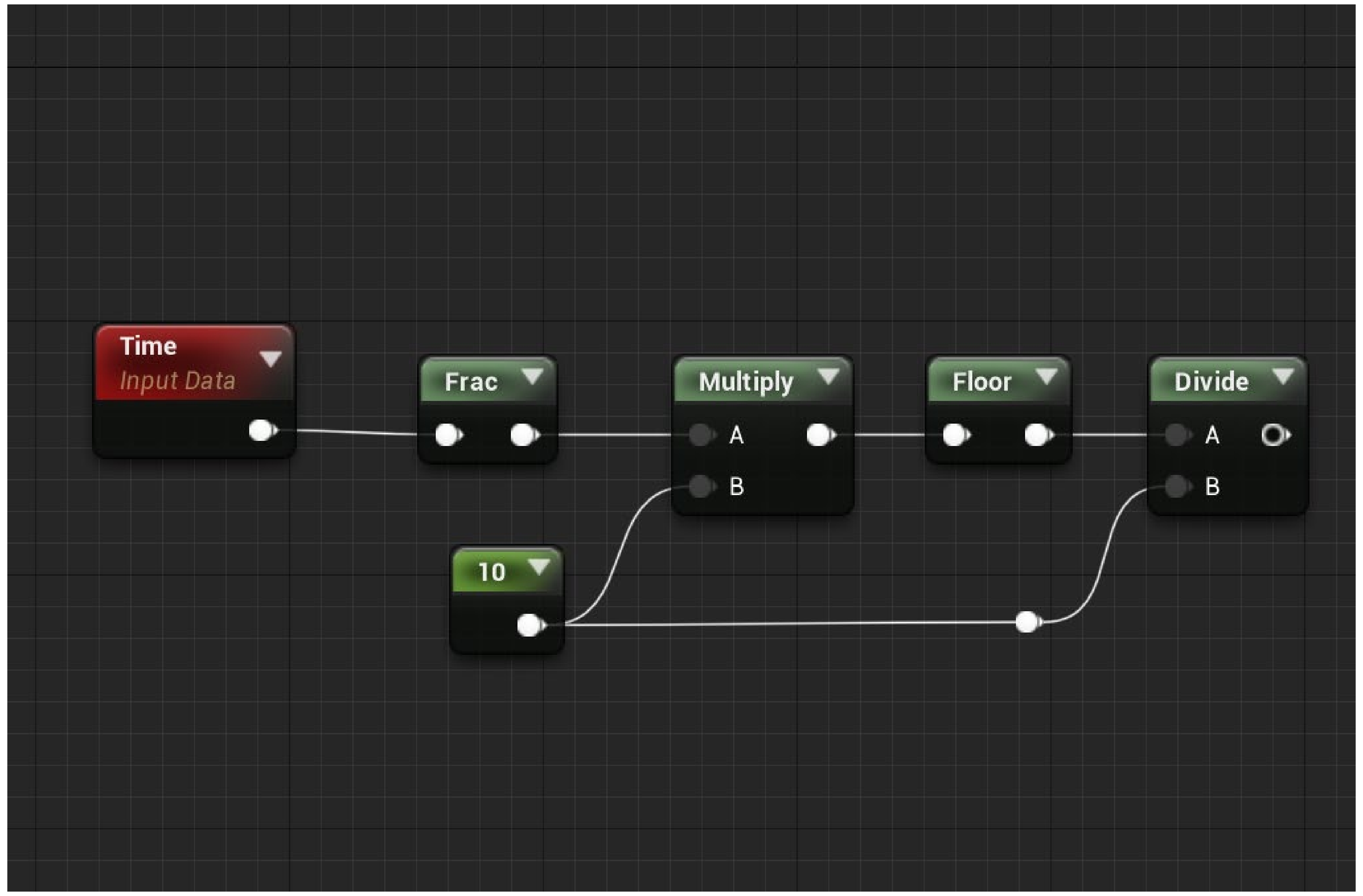
Graphtoy v0.3 by Inigo Quilez

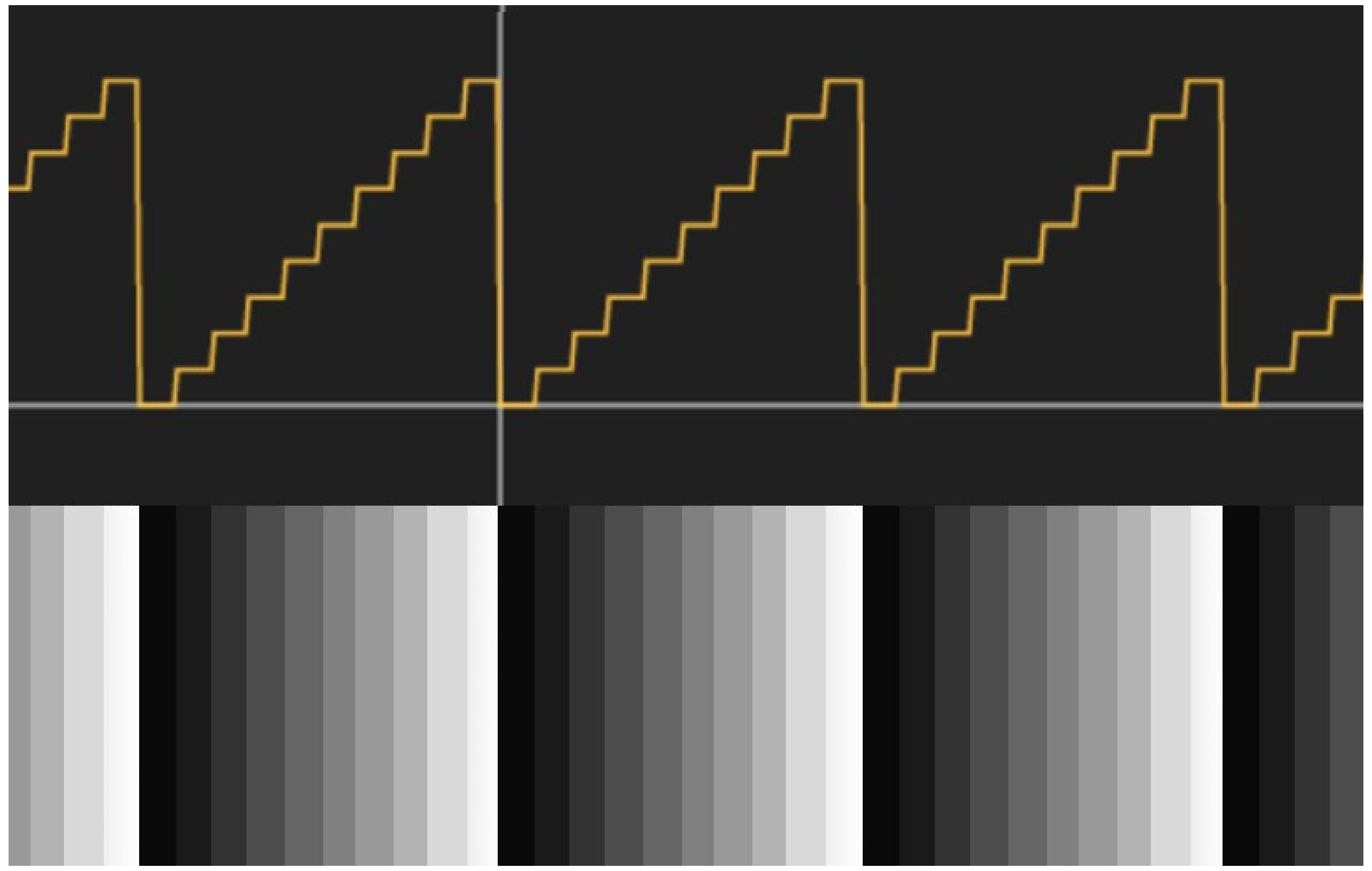
frac(time)

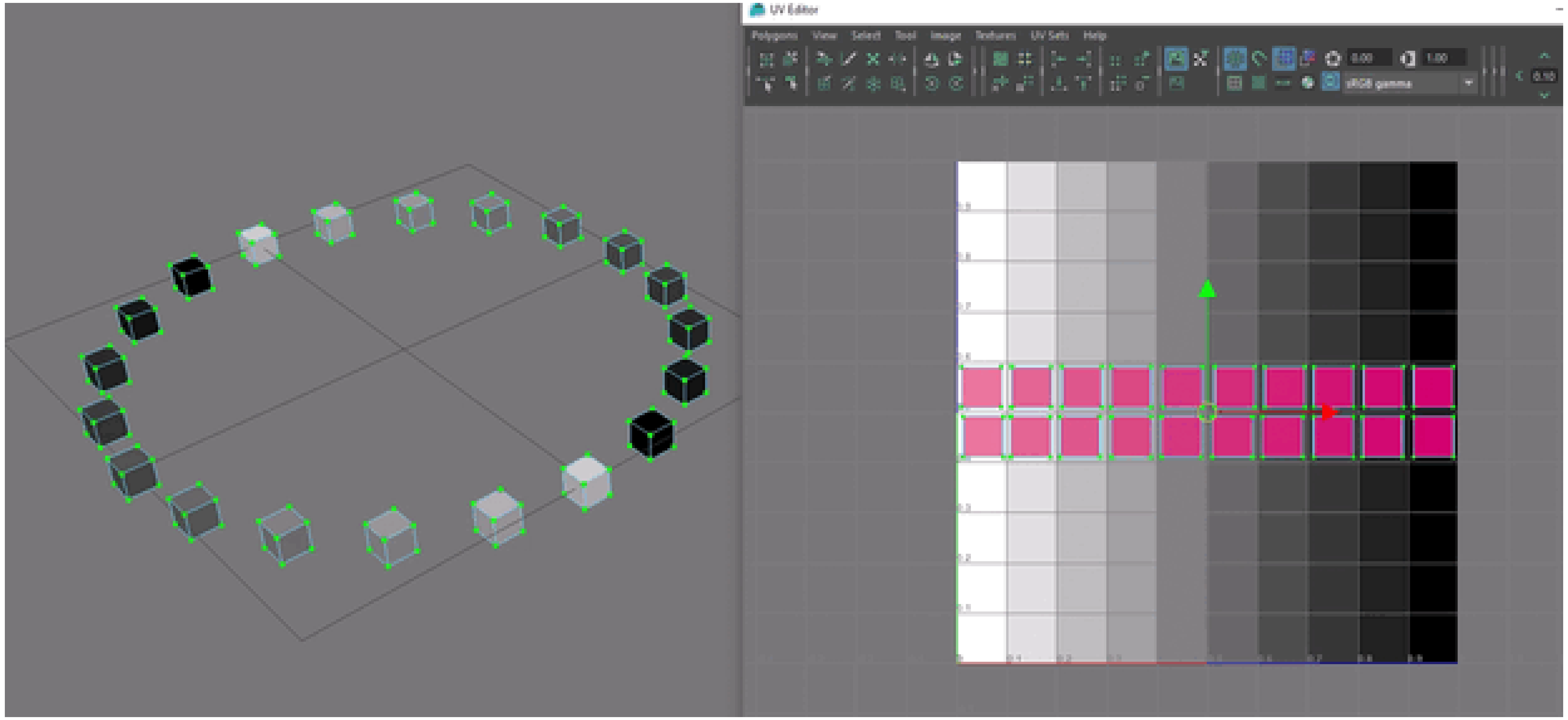


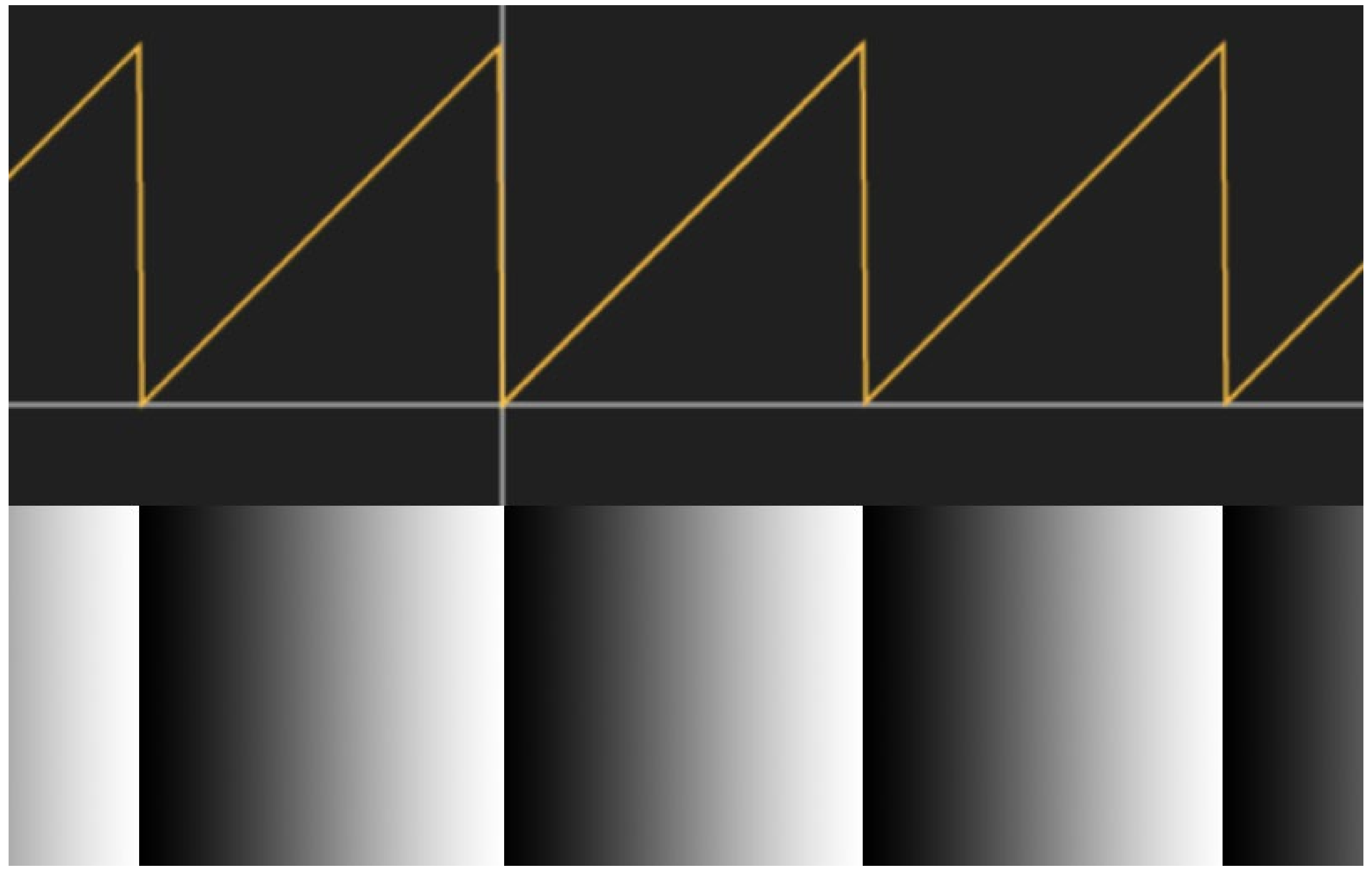


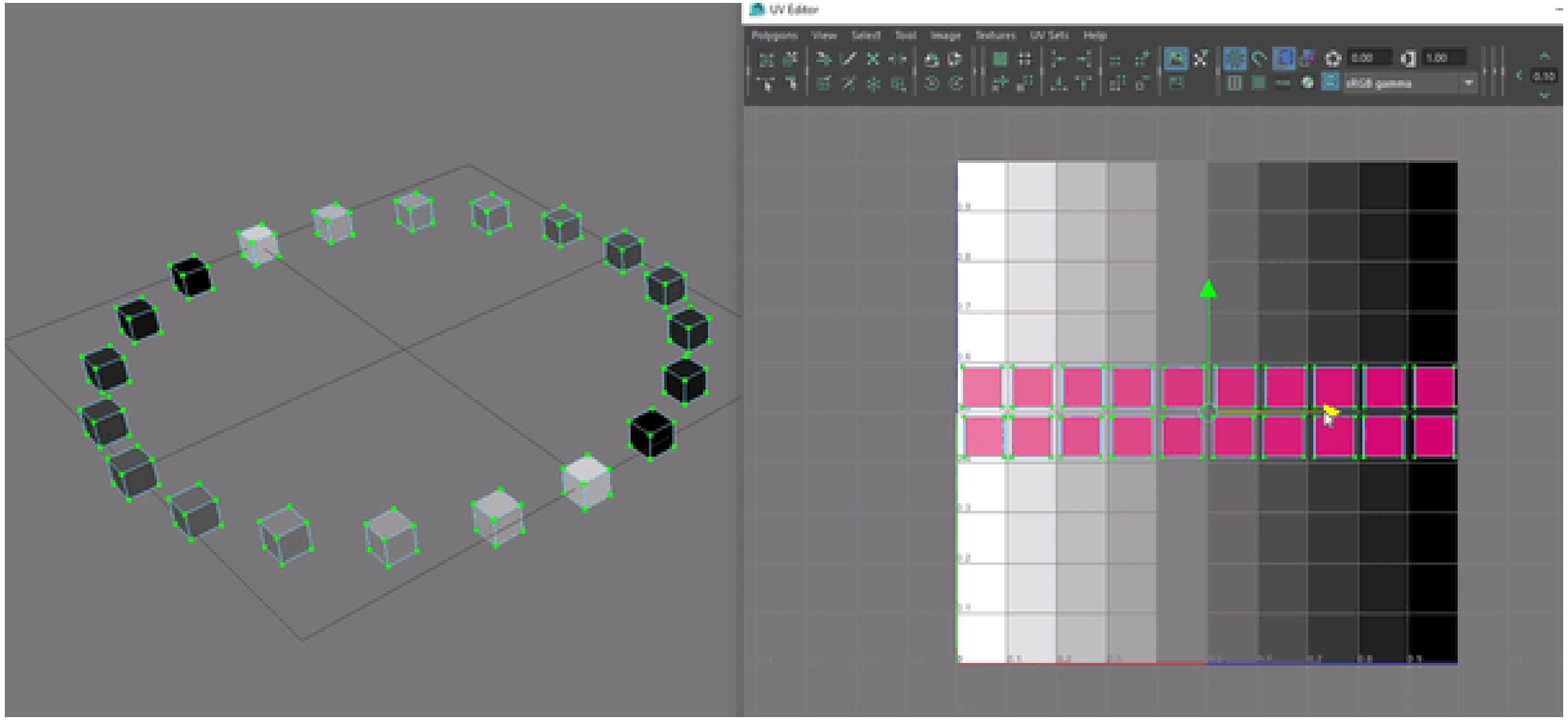
floor(frac(time)*10)/10

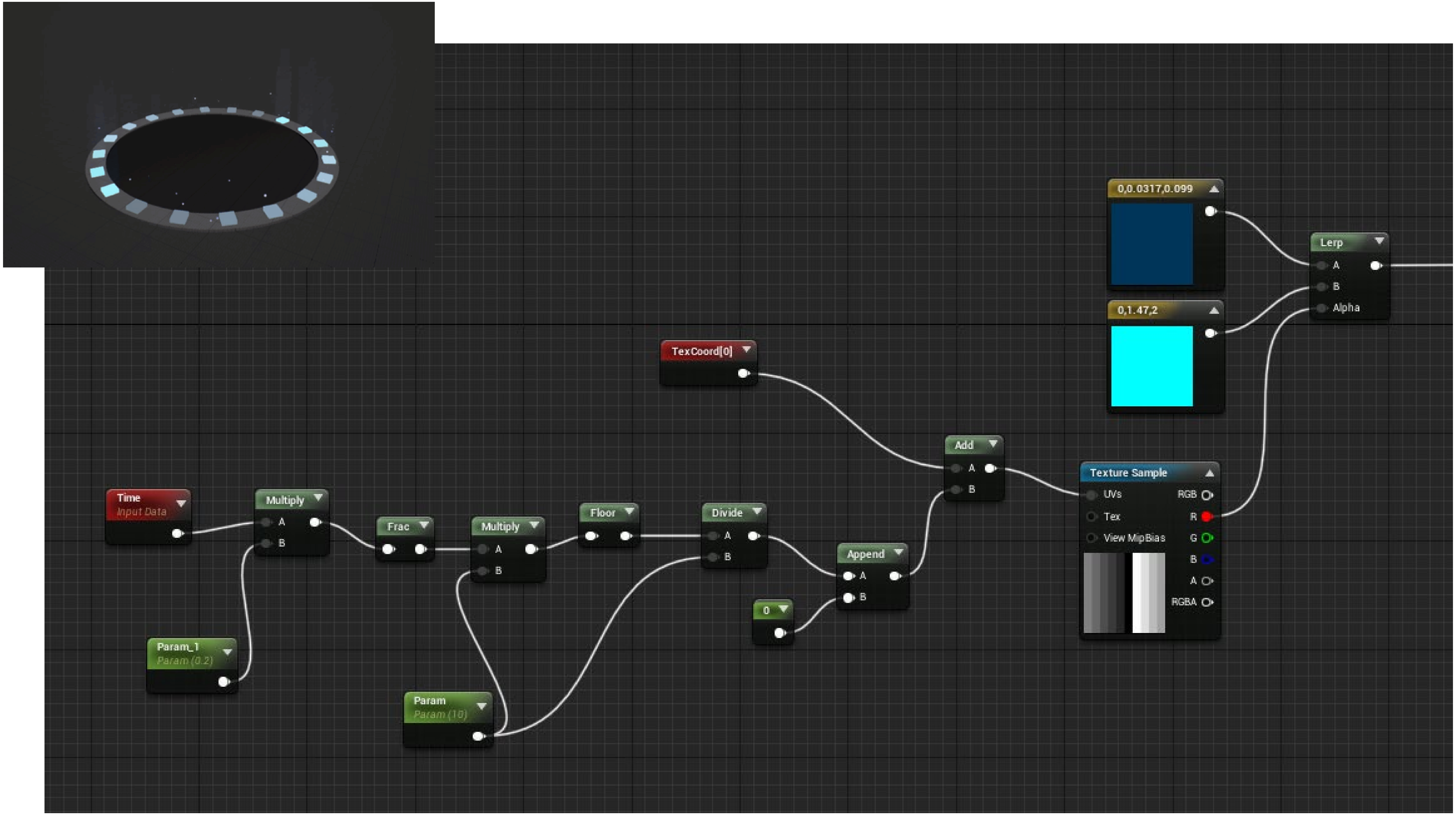


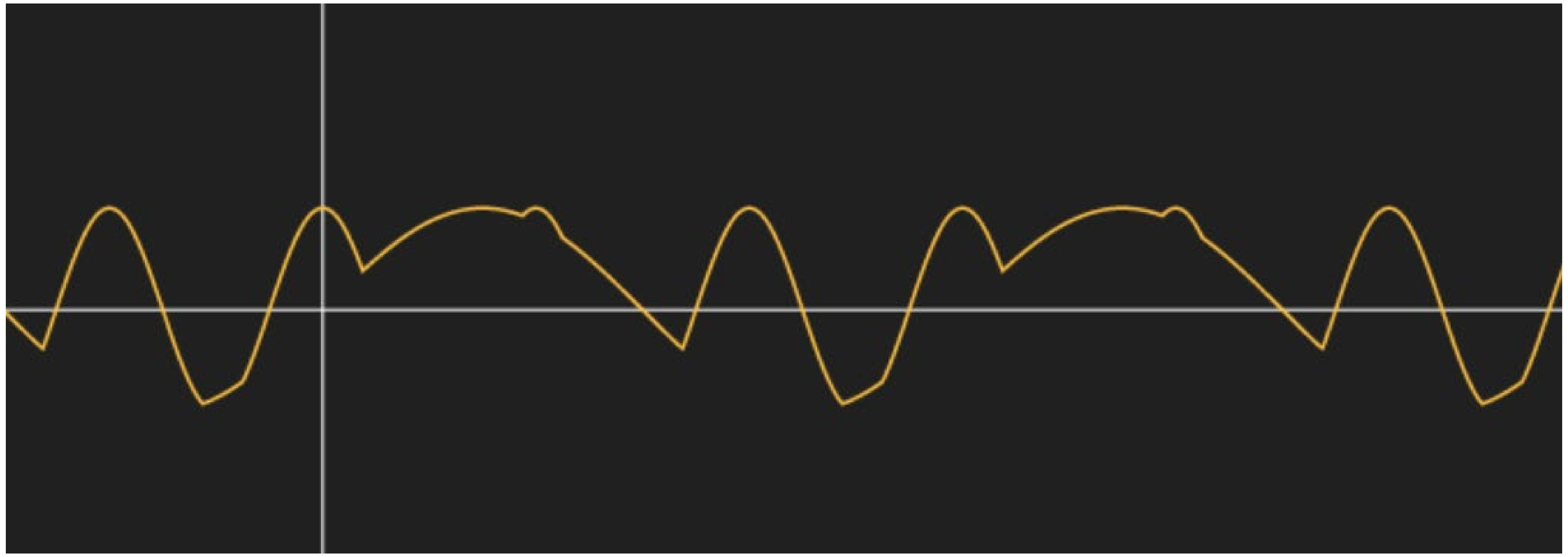



UV Editor
Polygons
View
Select
Tool
Image
Textures
UV Sets
Help
0.00
1.00
sRGB gamma
0.10

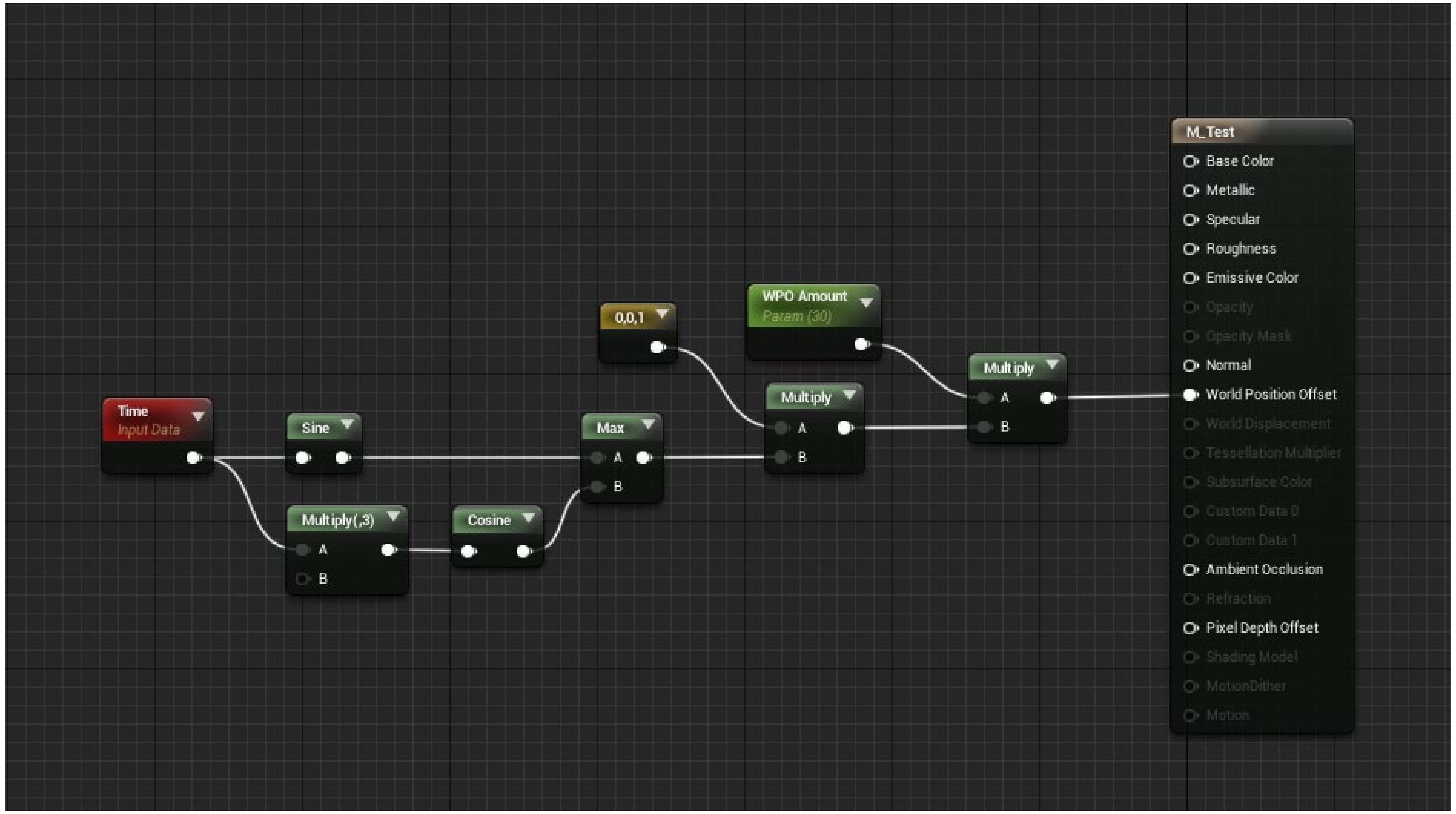


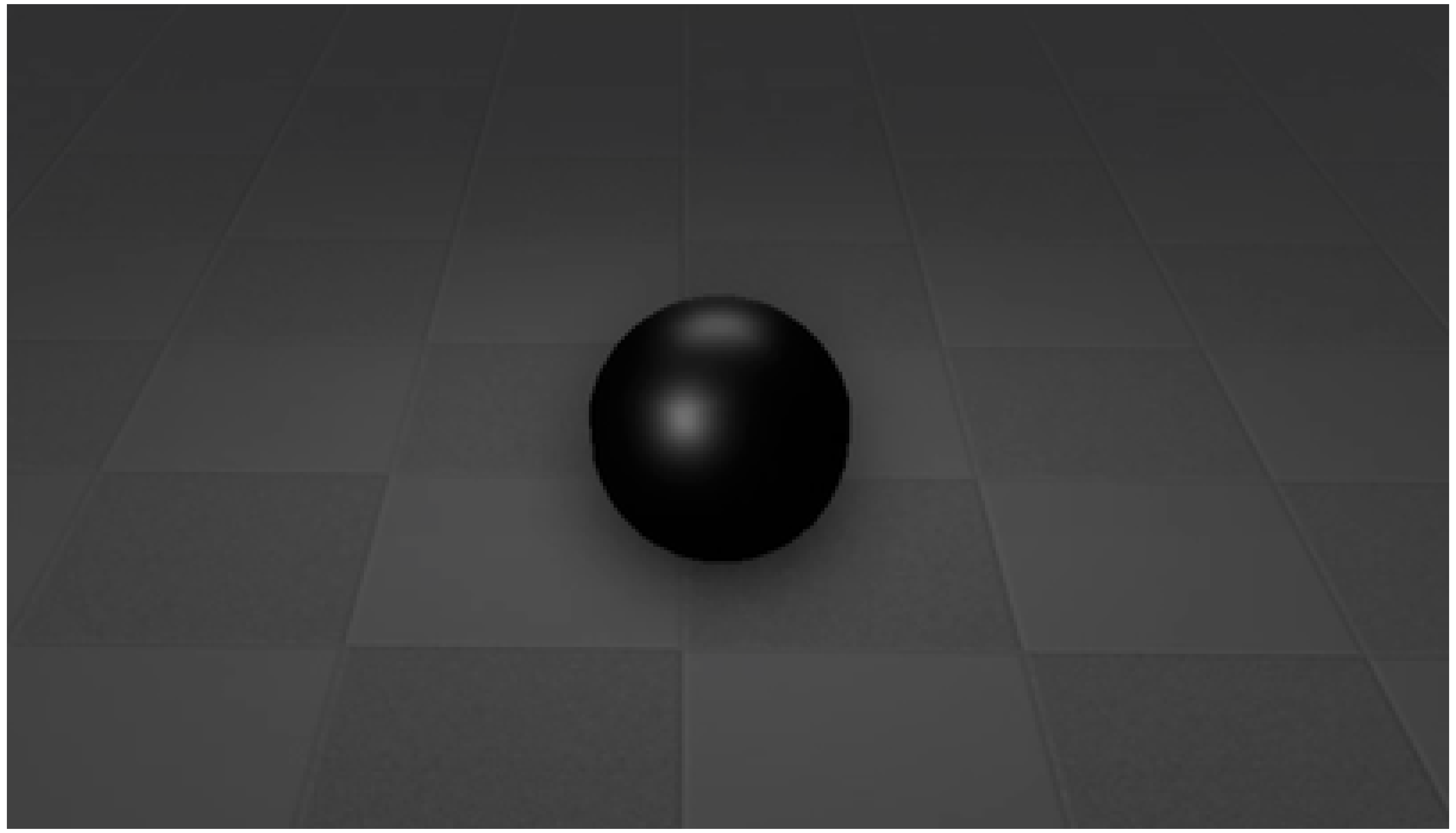

UV Editor
Polygons
View
Select
Tool
Image
Textures
UV Sets
Help
0.00
1.00
sRGB gamma


Time
Input Data
Multiply
A
B
Param_1
Param (0.2)
Frac
Multiply
A
B
Param
Param (10)
Floor
Divide
A
B
0
Append
A
B
TexCoord[0]
Add
A
B
Texture Sample
UVs
Tex
View MipBias
RGB
R
G
B
A
RGBA
0,0.0317,0.099
0,1.47,2
Lerp
A
B
Alpha



max(sin(time),cos(x*3))





LINKS

http://www.iquilezles.org/apps/graphtoy/

(or google ‘graphtoy’)

http://tobyschachman.com/Shadershop/editor/

(or google ‘shadershop’)

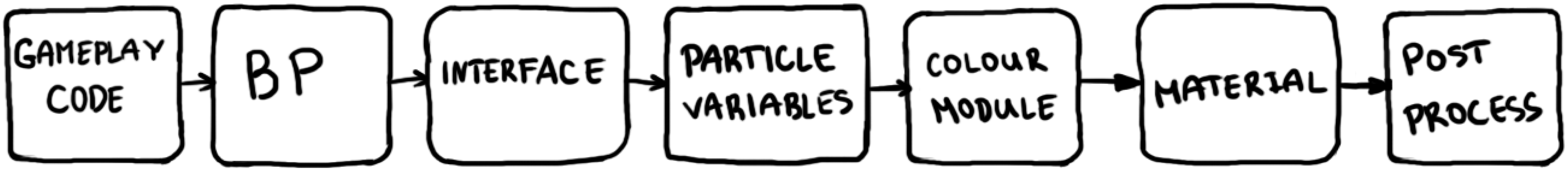
YouTube - tharlevfx

SOMETHING ISN’T WORKING

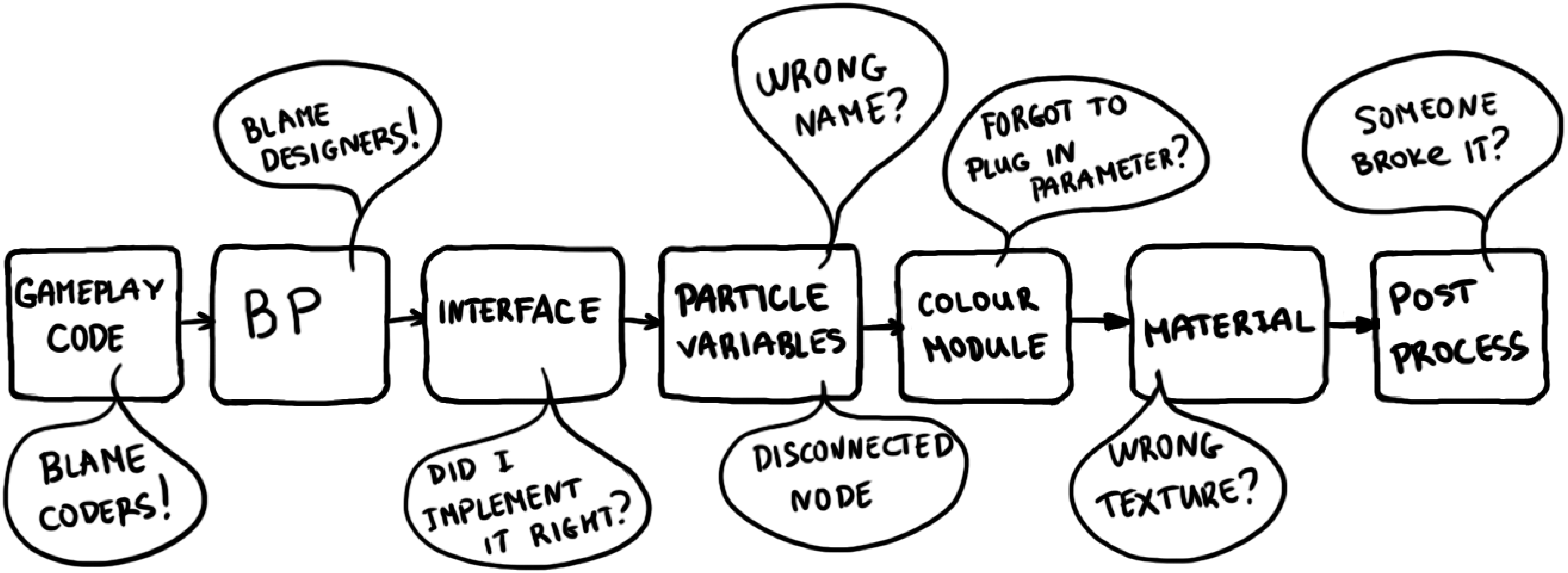
(help)

- Take a deep breath!
- Start debugging

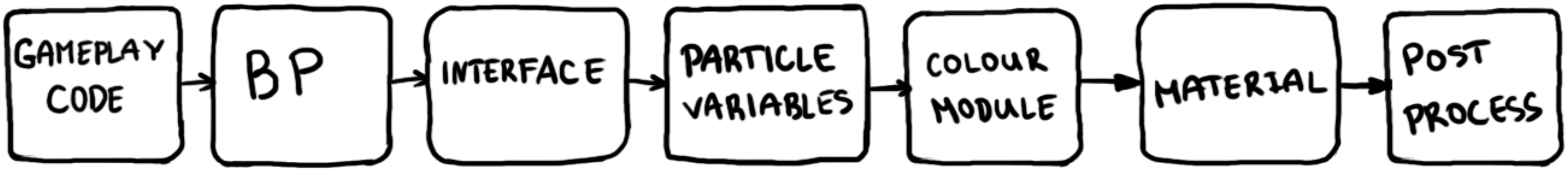
DEBUG


GAMEPLAY CODE
BP
INTERFACE
PARTICLE VARIABLES
COLOUR MODULE
MATERIAL
POST PROCESS

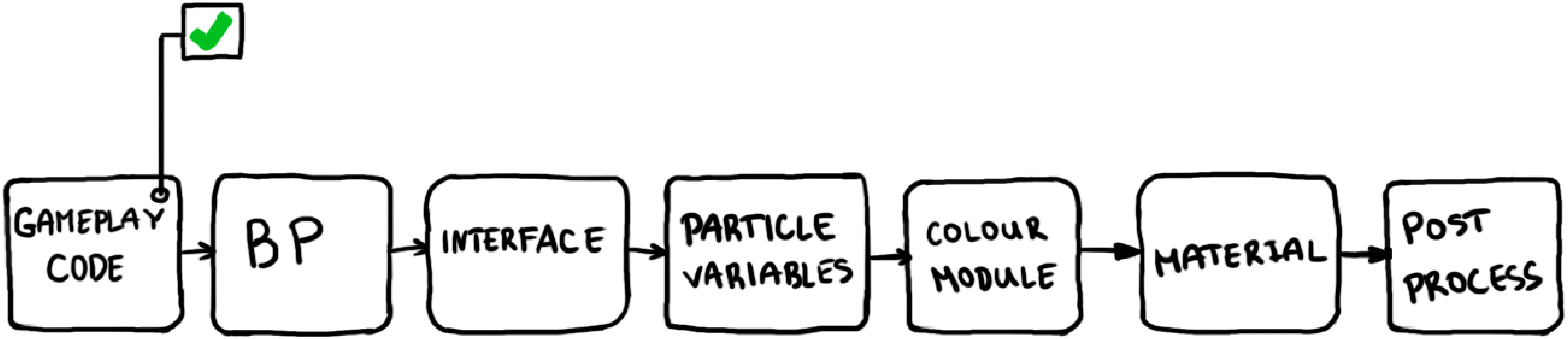
DON'T RUSH


GAMEPLAY CODE
BLAME CODERS!
BP
BLAME DESIGNERS!
INTERFACE
DID I IMPLEMENT IT RIGHT?
PARTICLE VARIABLES
WRONG NAME?
DISCONNECTED NODE
COLOUR MODULE
FORGOT TO PLUG IN PARAMETER?
MATERIAL
WRONG TEXTURE?
POST PROCESS
SOMEONE BROKE IT?

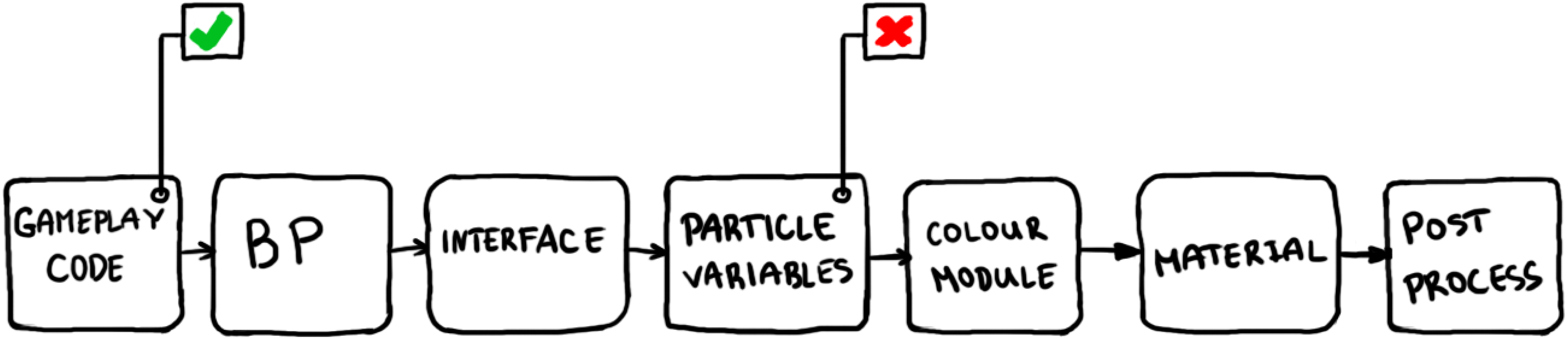
NARROW THE SEARCH



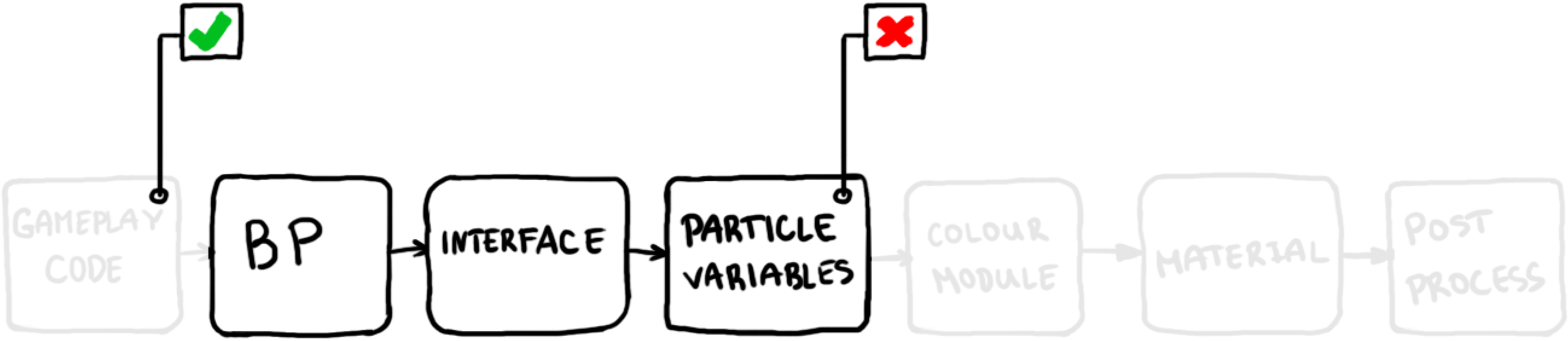
NARROW THE SEARCH



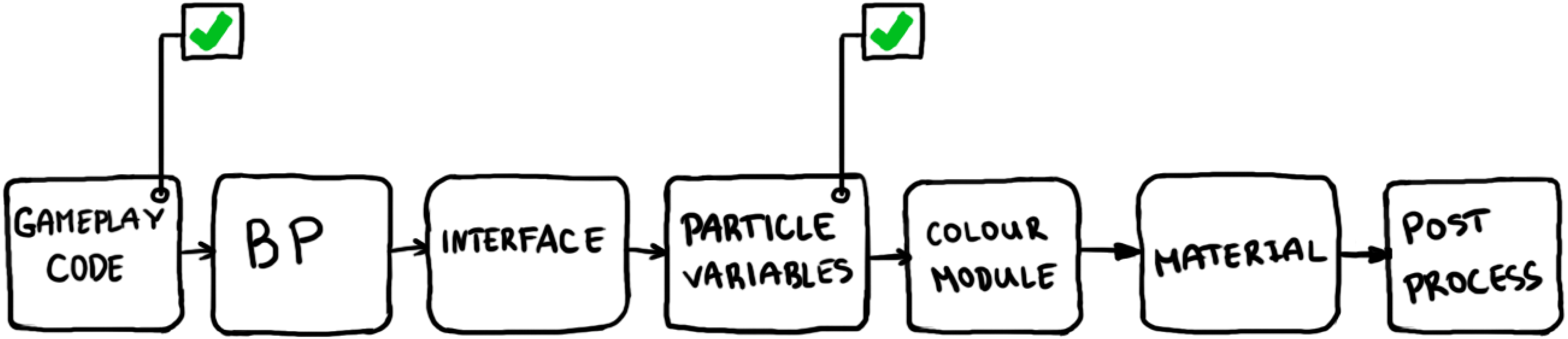
NARROW THE SEARCH



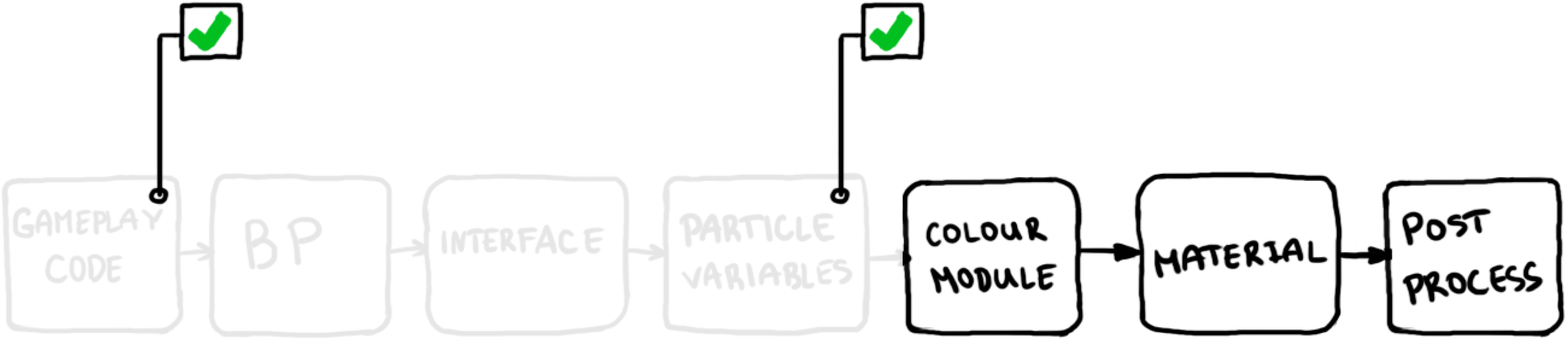
NARROW THE SEARCH



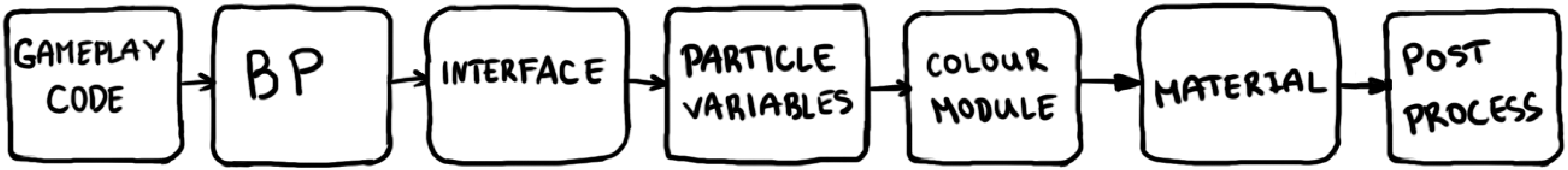
NARROW THE SEARCH



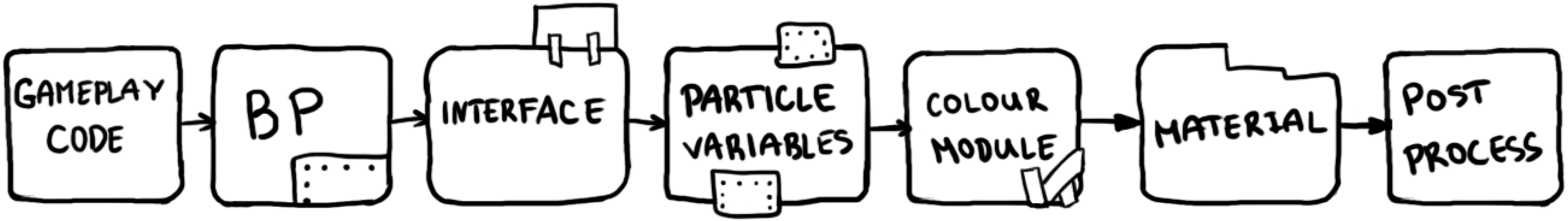
NARROW THE SEARCH



CHANGE ONE THING AT A TIME



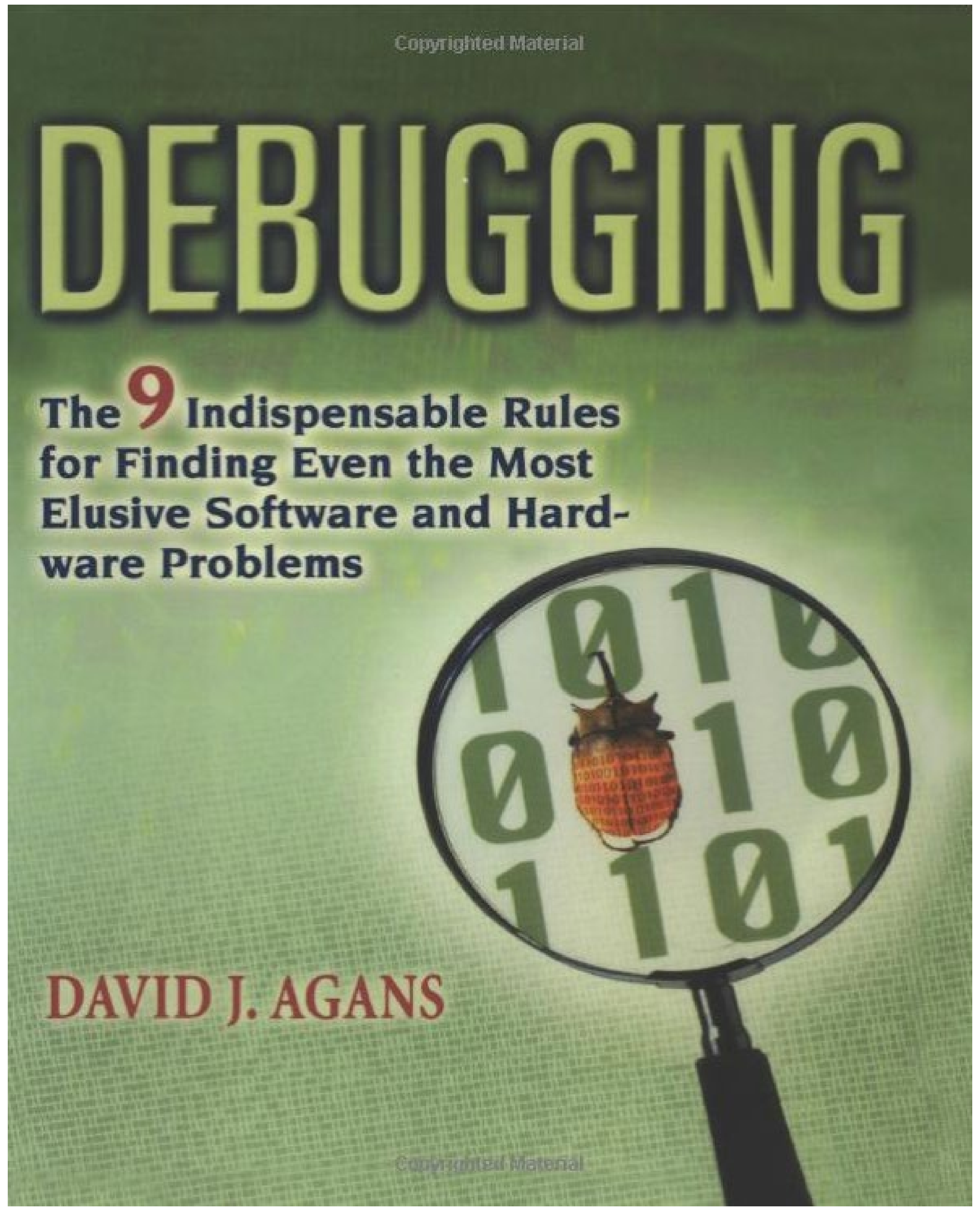
CHANGE ONE THING AT A TIME



DEBUG

- Understand the system
- Don’t rush
- Narrow the search
- Change one thing at a time

DEBUG


Copyrighted Material
DEBUGGING
The 9 Indispensable Rules for Finding Even the Most Elusive Software and Hardware Problems
DAVID J. AGANS
Copyrighted Material

David J.Agans

Debugging: The 9 Indispensable Rules for Finding Even the Most Elusive Software and Hardware Problems

WHAT’S NEXT?

EXTENSION AND SUPPORT

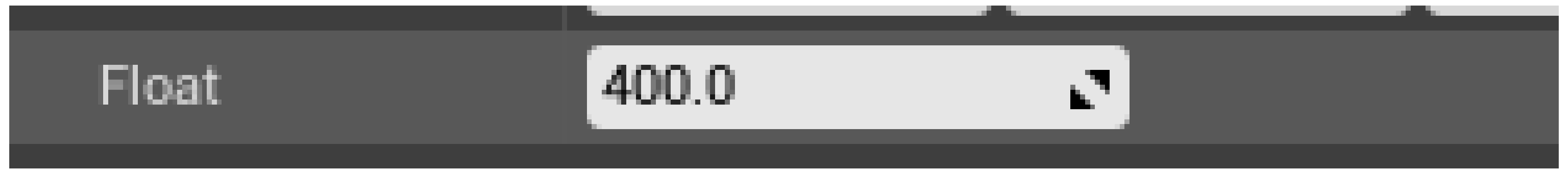
- Tricky details are easy to forget

EXTENSION AND SUPPORT

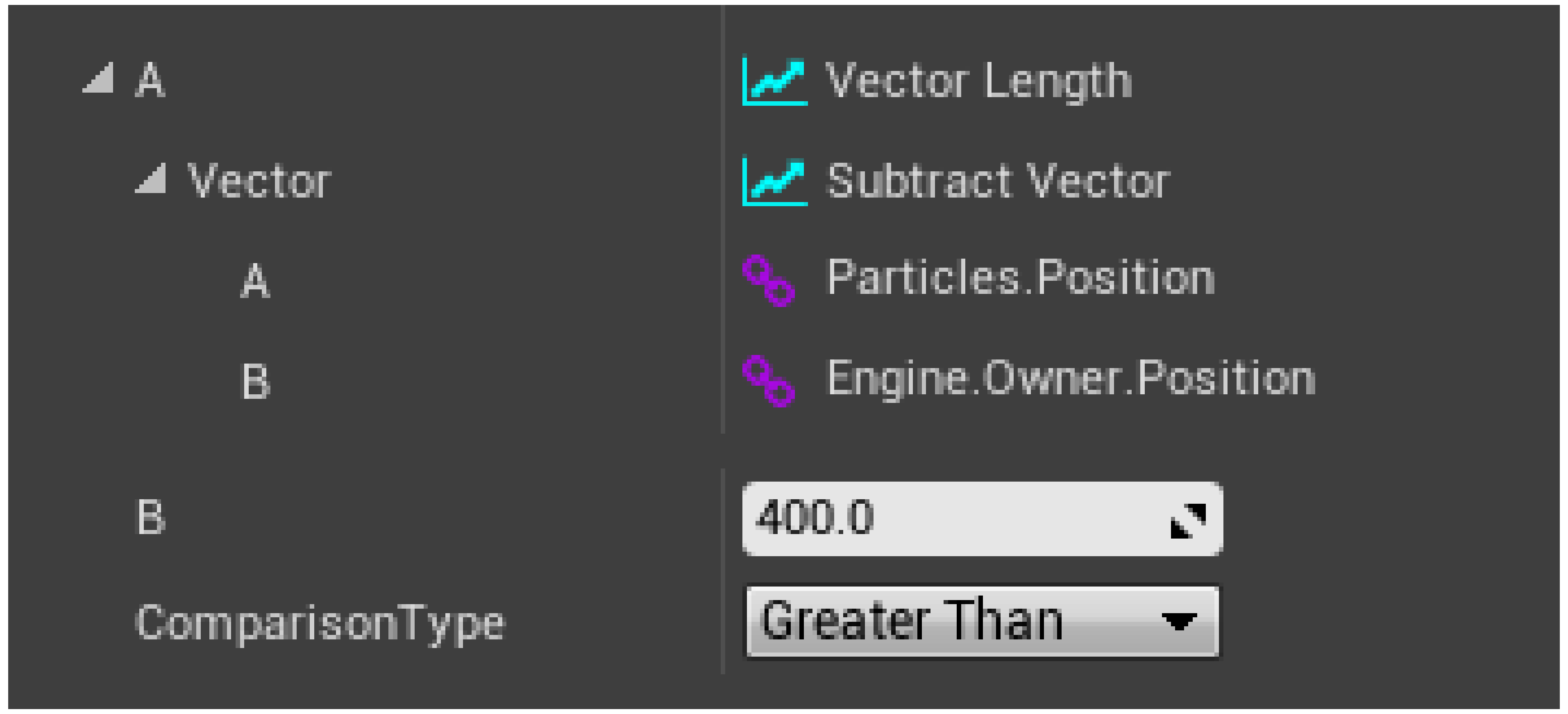
- Tricky details are easy to forget
- Someone else will most probably have to work with your assets and systems

EXTENSION AND SUPPORT

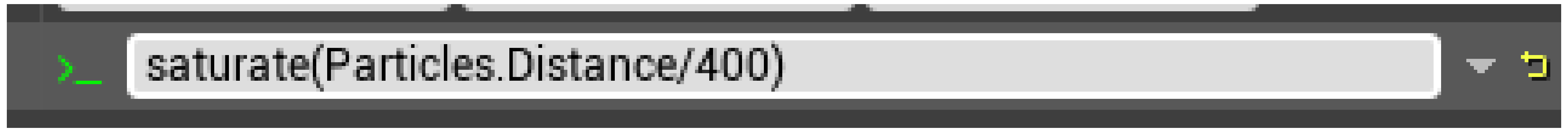
- Tricky details are easy to forget
- Someone else will most probably have to work with your assets and systems
- Variety of tools allow implementations that can be completely different and hard to understand



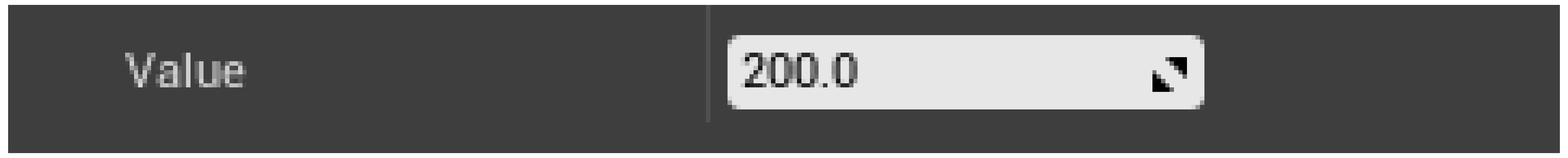
Float 400.0



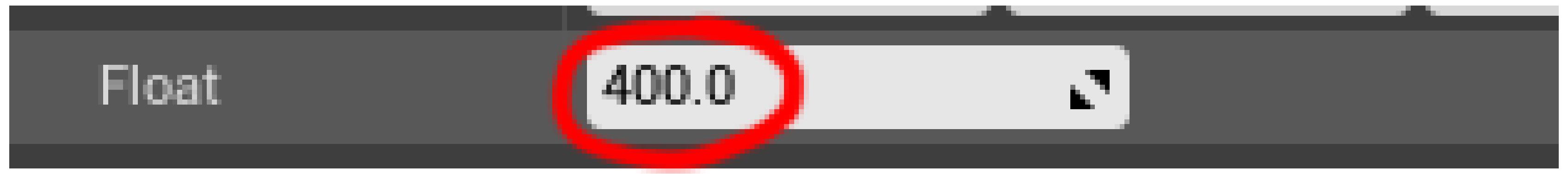
- A — Vector Length
  - Vector — Subtract Vector
    - A — Particles.Position
    - B — Engine.Owner.Position
- B — 400.0
- ComparisonType — Greater Than



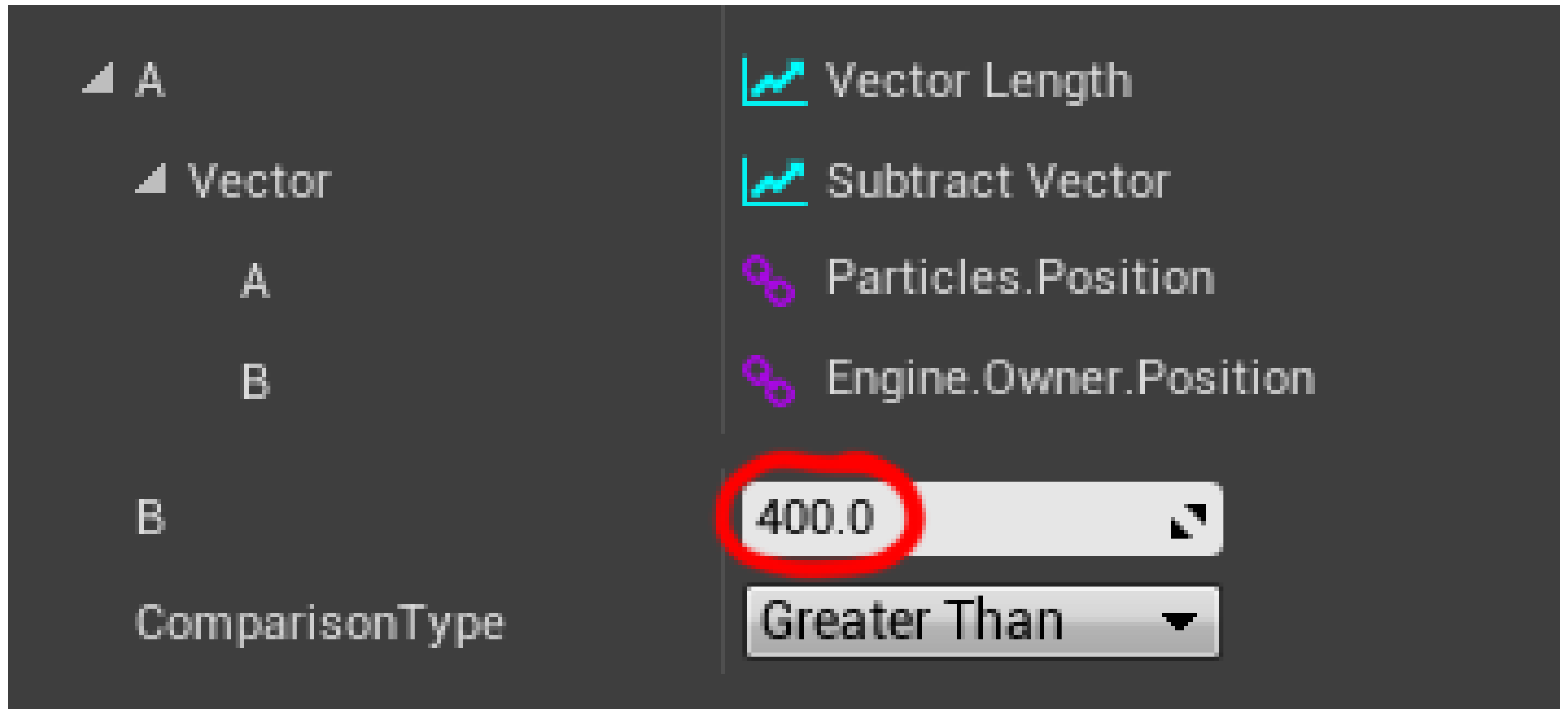
```
saturate(Particles.Distance/400)
```



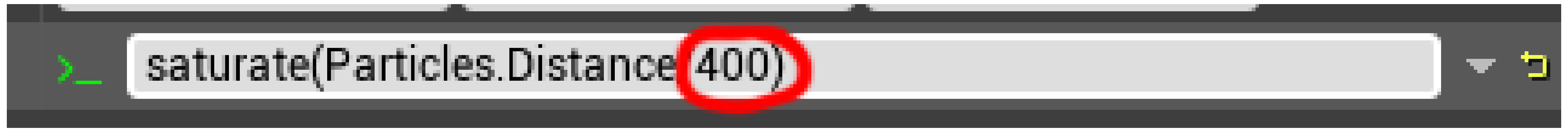
Value 200.0



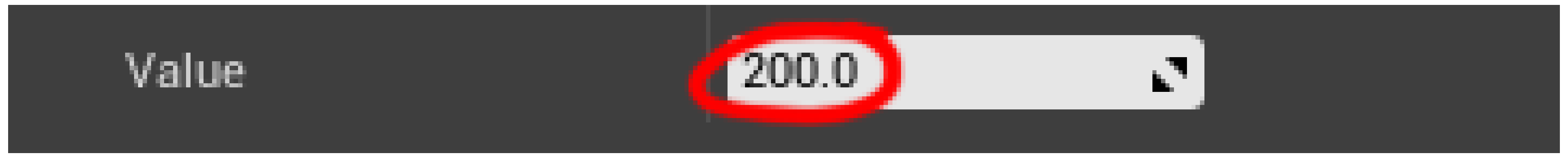
Float	400.0



◢ A	Vector Length
◢ Vector	Subtract Vector
A	Particles.Position
B	Engine.Owner.Position
B	400.0
ComparisonType	Greater Than



```
saturate(Particles.Distance 400)
```



Value	200.0

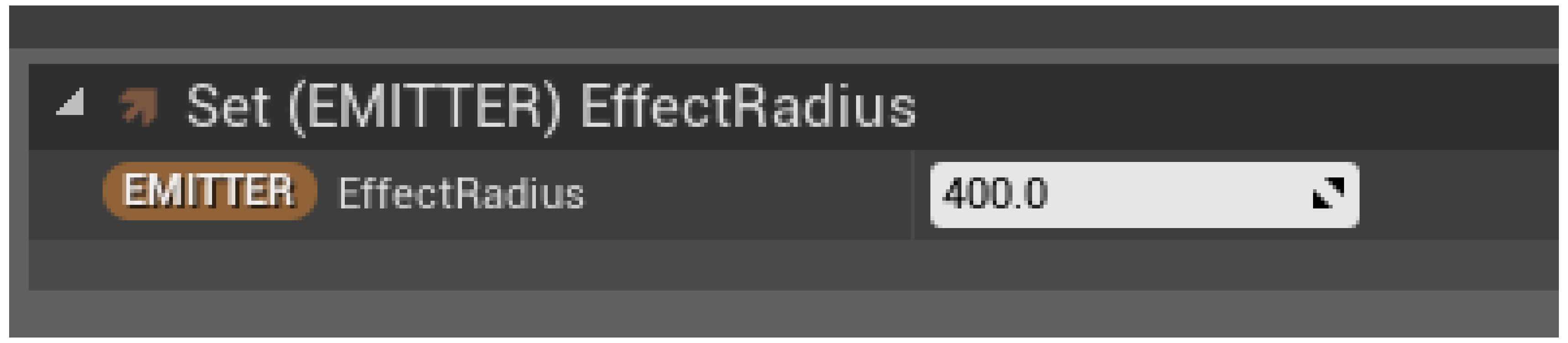
MAGIC NUMBERS

- Numbers that seem arbitrary and have no context or meaning
- Difficult to replace and maintain

MAGIC NUMBERS

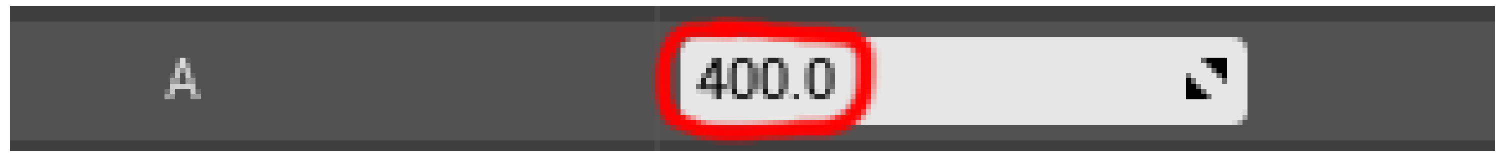
- Numbers that seem arbitrary and have no context or meaning
- Difficult to replace and maintain

400 -> EffectRadius

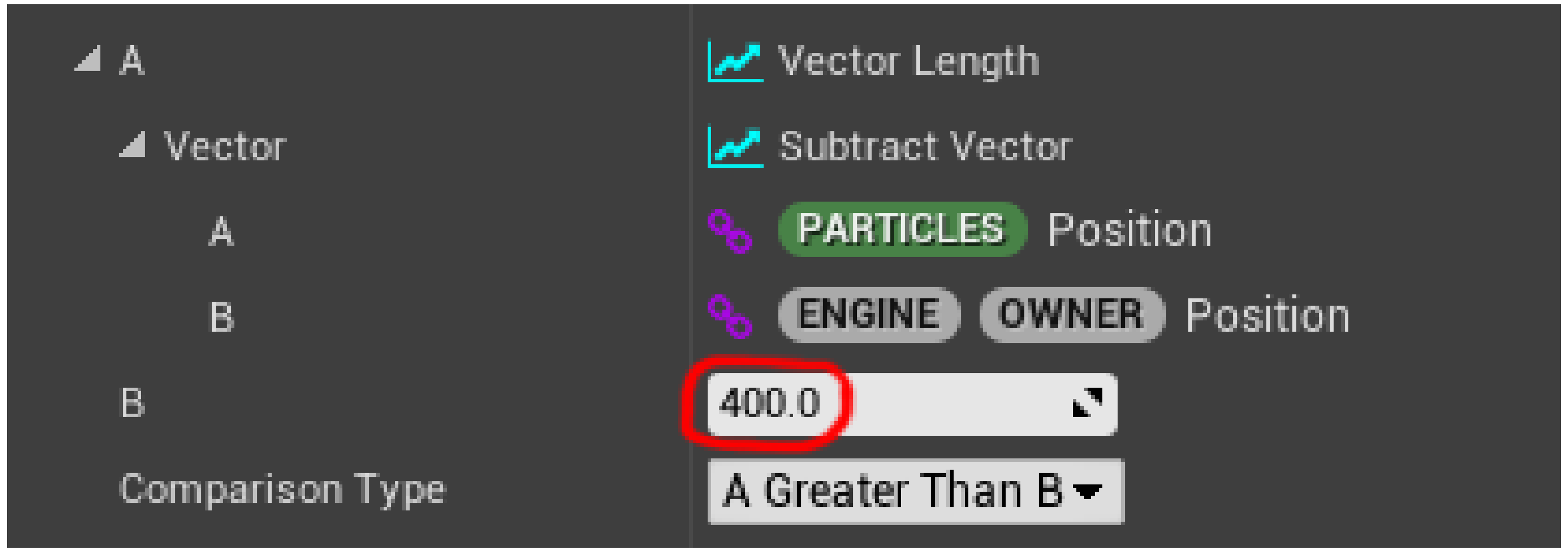


Set (EMITTER) EffectRadius

EMITTER EffectRadius 400.0



A 400.0



A Vector Length

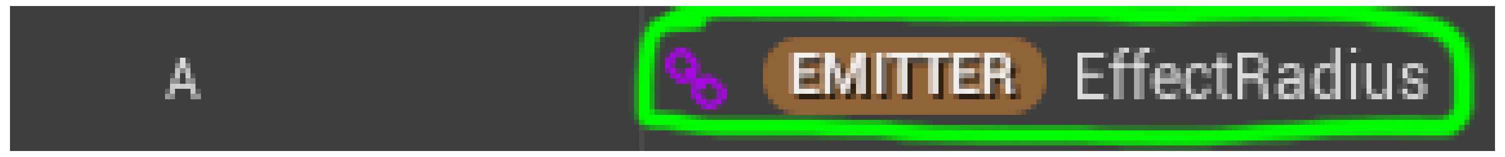
Vector Subtract Vector

A PARTICLES Position

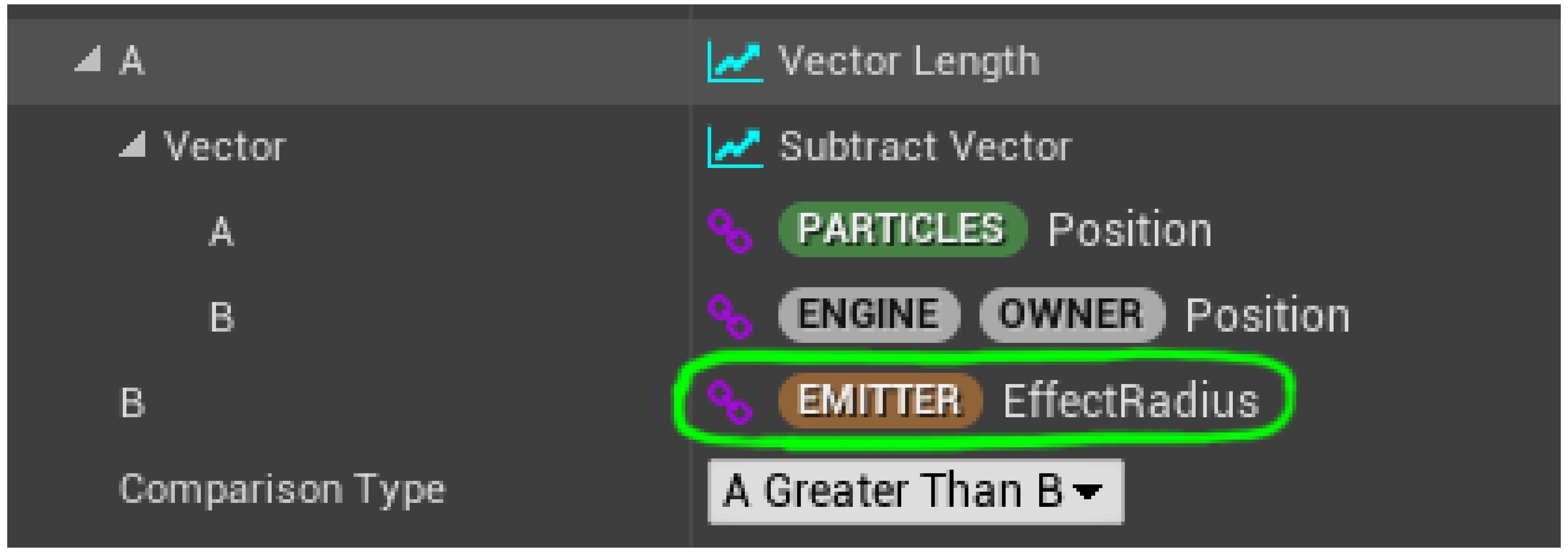
B ENGINE OWNER Position

B 400.0

Comparison Type A Greater Than B



A EMITTER EffectRadius



A Vector Length

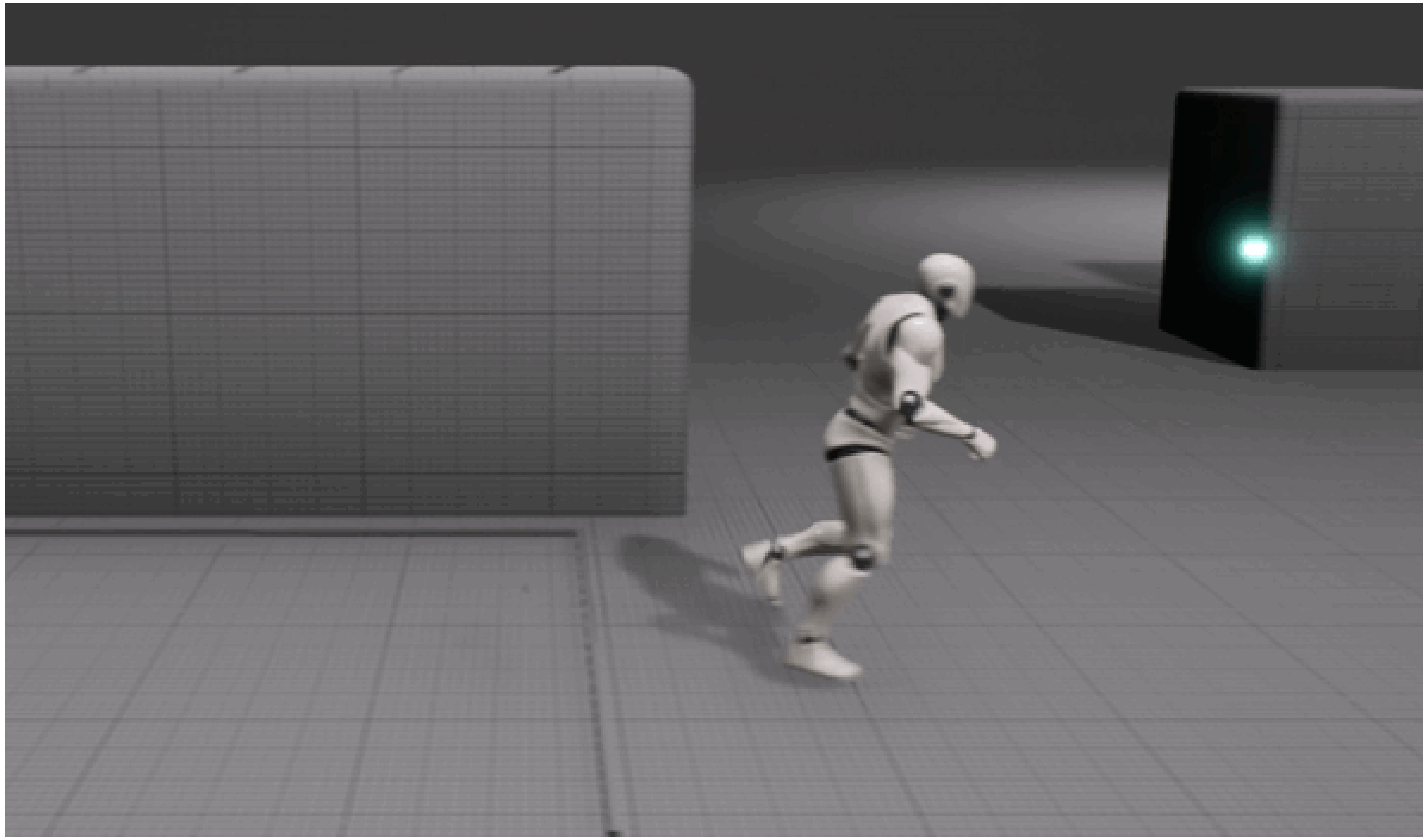
Vector Subtract Vector

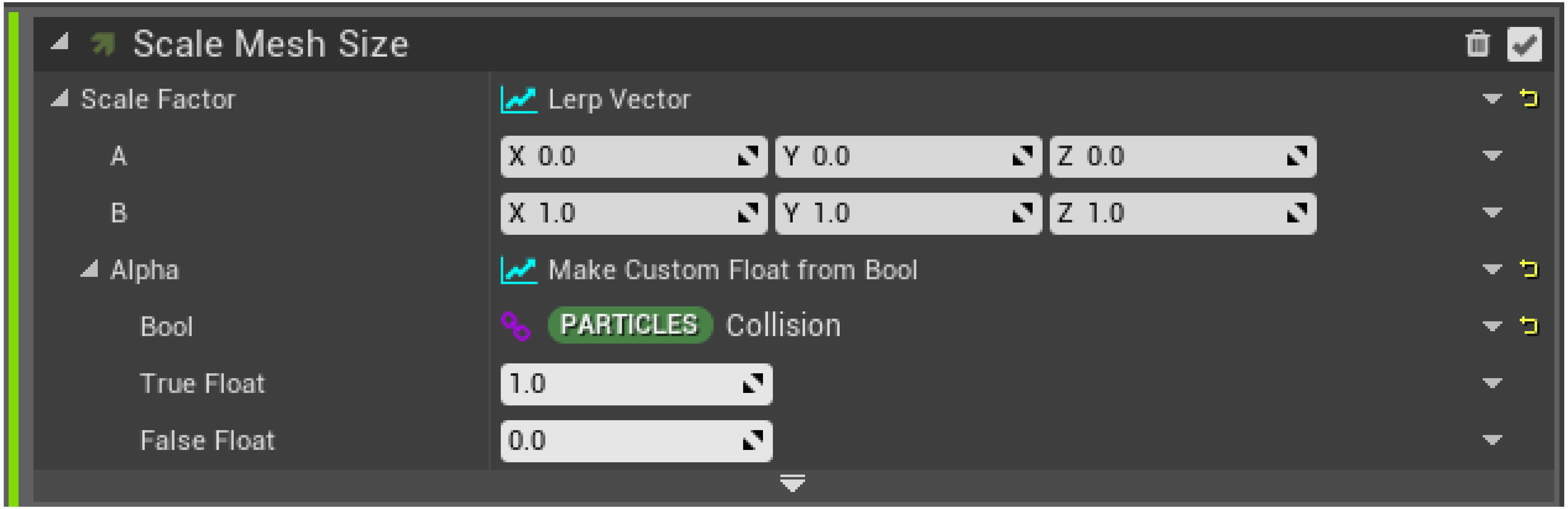
A PARTICLES Position

B ENGINE OWNER Position

B EMITTER EffectRadius

Comparison Type A Greater Than B





Scale Mesh Size

Scale Factor: Lerp Vector

A: X 0.0 Y 0.0 Z 0.0

B: X 1.0 Y 1.0 Z 1.0

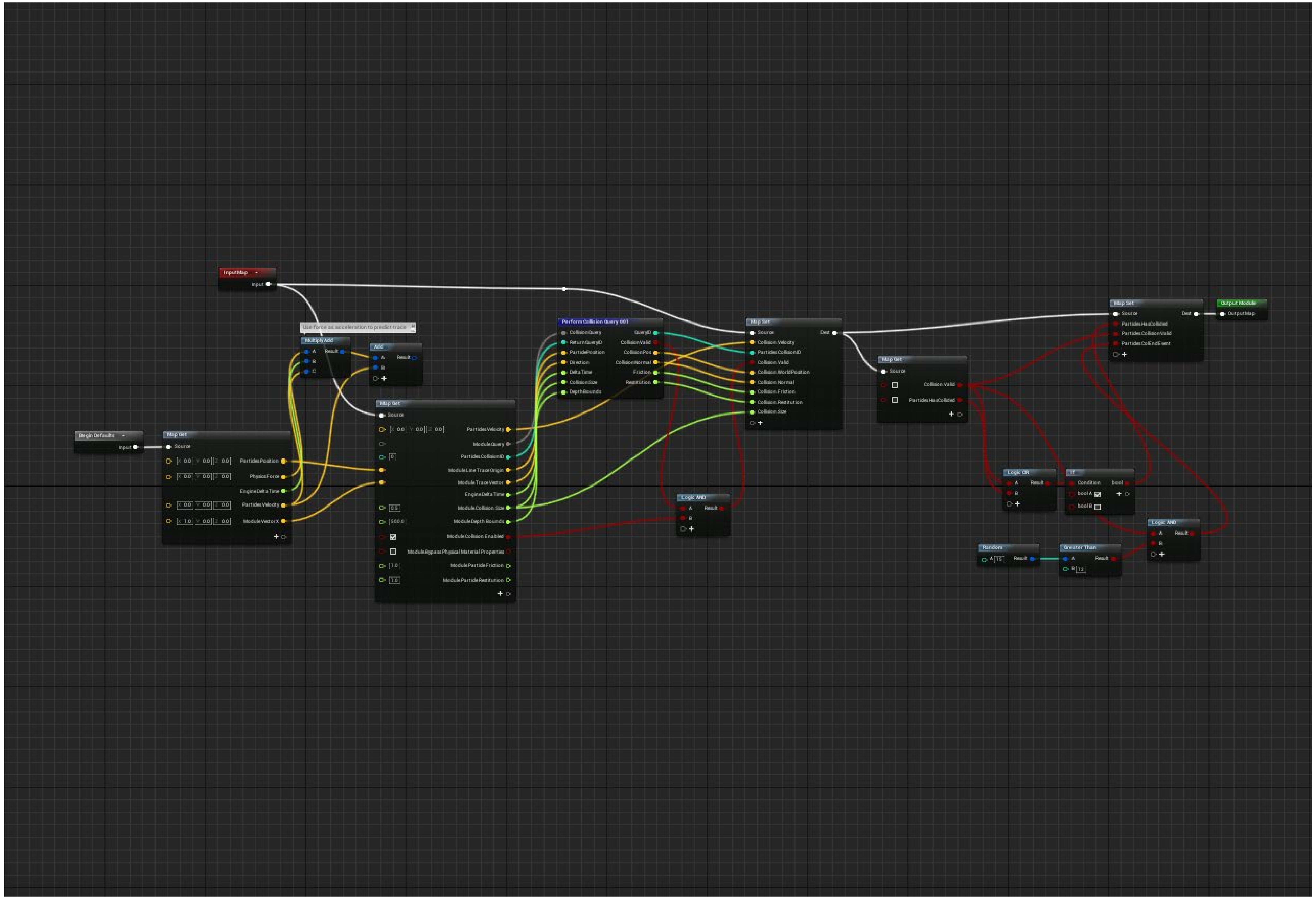
Alpha: Make Custom Float from Bool

Bool: PARTICLES Collision

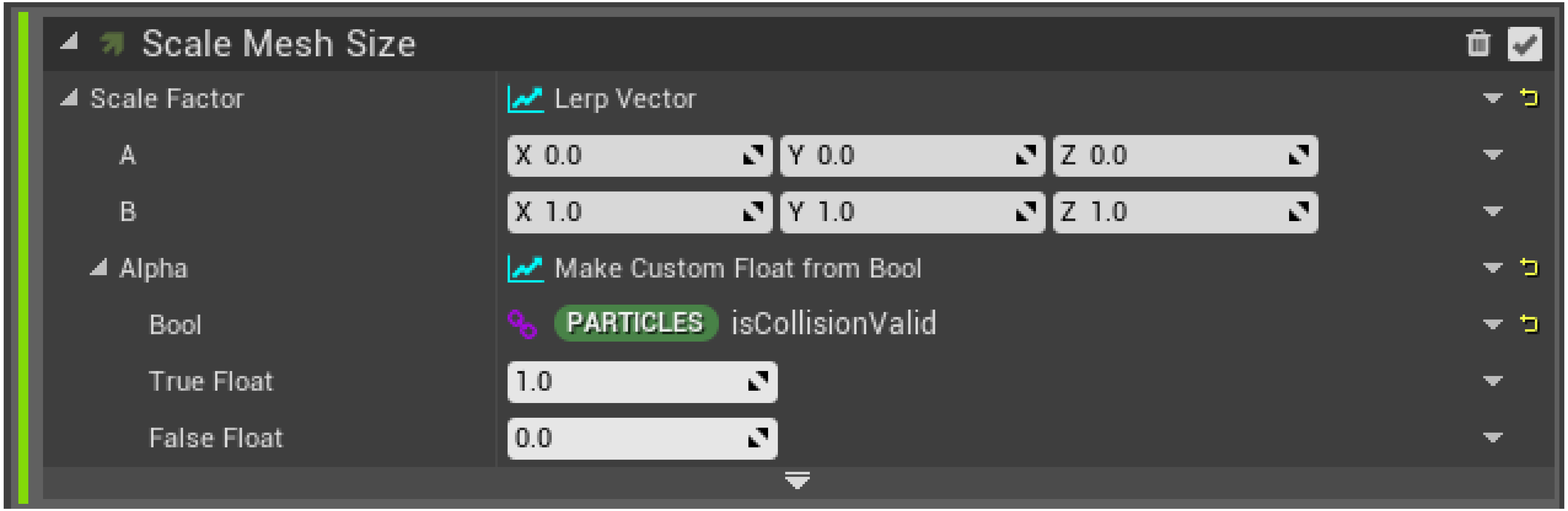
True Float: 1.0

False Float: 0.0

(Particles) Collision


InputMap
Input
Use force as acceleration to predict trace
Multiply Add
A
B
C
Result
Add
A
B
Result
Begin Defaults
Input
Map Get
Source
ParticlesPosition
PhysicsForce
EngineDeltaTime
ParticlesVelocity
ModuleVectorX
Map Get
Source
ParticlesVelocity
ModuleQuery
ParticlesCollisionID
ModuleLineTraceOrigin
ModuleTraceVector
EngineDeltaTime
ModuleCollisionSize
ModuleDepthBounds
ModuleCollisionEnabled
ModuleBypassPhysicalMaterialProperties
ModuleParticleFriction
ModuleParticleRestitution
Perform Collision Query 001
CollisionQuery
ReturnQueryID
ParticlePosition
Direction
DeltaTime
CollisionSize
DepthBounds
QueryID
CollisionValid
CollisionPos
CollisionNormal
Friction
Restitution
Logic AND
A
B
Result
Map Set
Source
Dest
Collision.Velocity
Particles.CollisionID
Collision.Valid
Collision.WorldPosition
Collision.Normal
Collision.Friction
Collision.Restitution
Collision.Size
Map Get
Source
Collision.Valid
ParticlesHasCollided
Logic OR
A
B
Result
If
Condition
bool A
bool B
bool
Random
A
Result
Greater Than
A
B
Result
Logic AND
A
B
Result
Map Set
Source
Dest
ParticlesHasCollided
ParticlesCollisionValid
ParticlesColEndEvent
Output Module
OutputMap

PRO TV
AP



Scale Mesh Size

Scale Factor	Lerp Vector		
A	X 0.0	Y 0.0	Z 0.0
B	X 1.0	Y 1.0	Z 1.0
Alpha	Make Custom Float from Bool		
Bool	PARTICLES isCollisionValid		
True Float	1.0		
False Float	0.0		

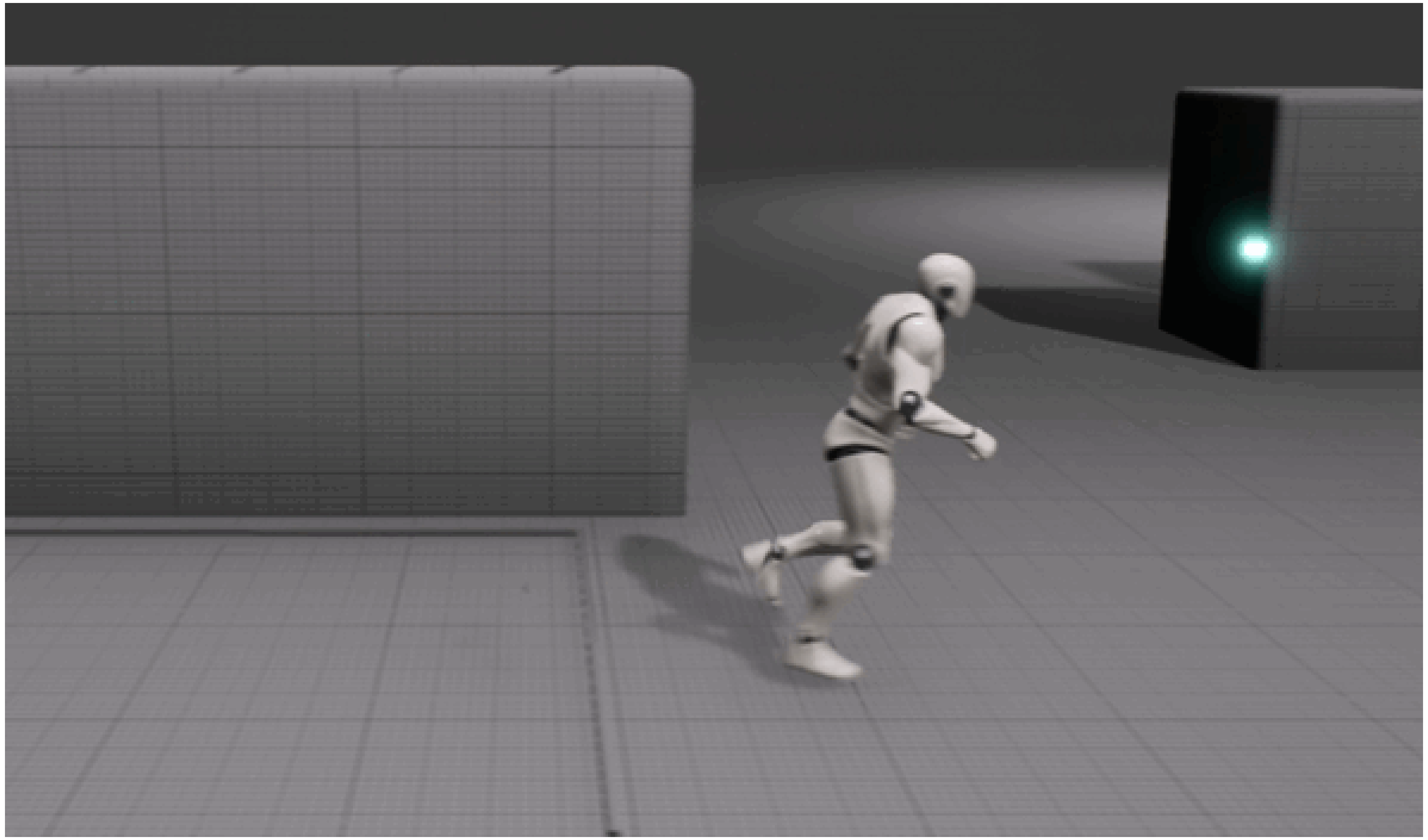
(Particles) isCollisionValid

SELF DOCUMENTING VFX

- You should be able to understand functionality just by looking at it

Collision -> isCollisionValid

Radius -> getEffectRadius



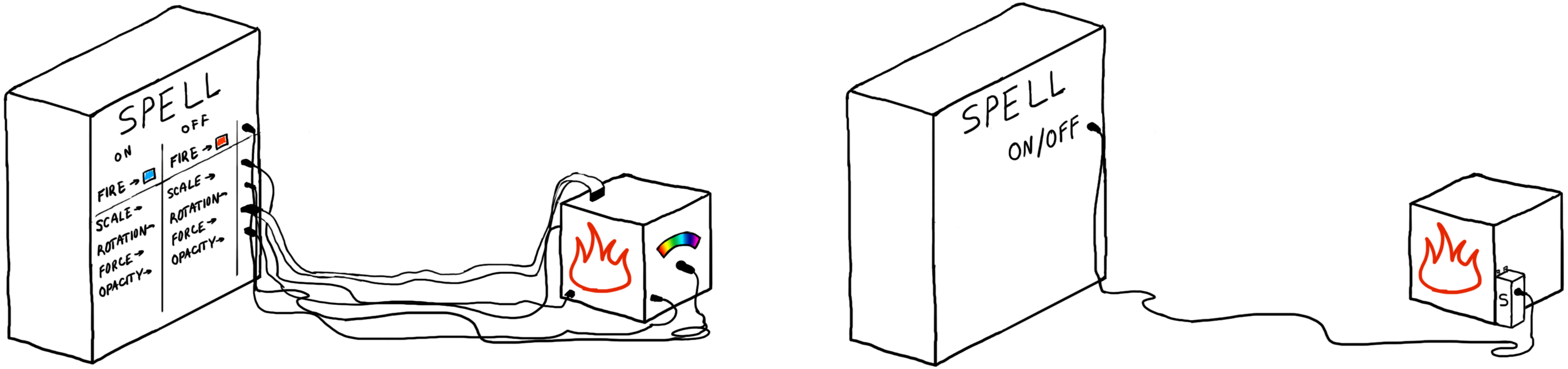
EXTENSION AND SUPPORT

- Avoid magic numbers where possible
- Use variable names that are self-explanatory
- Write documentation

TAKEAWAYS

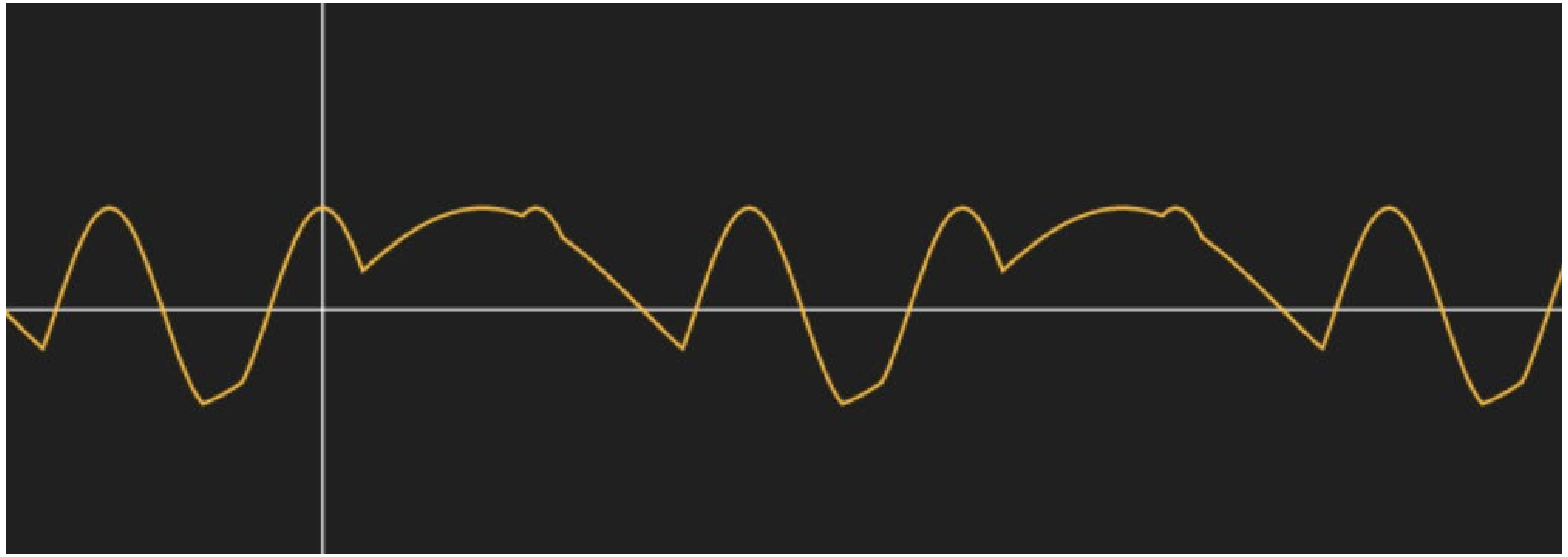
TAKEAWAYS

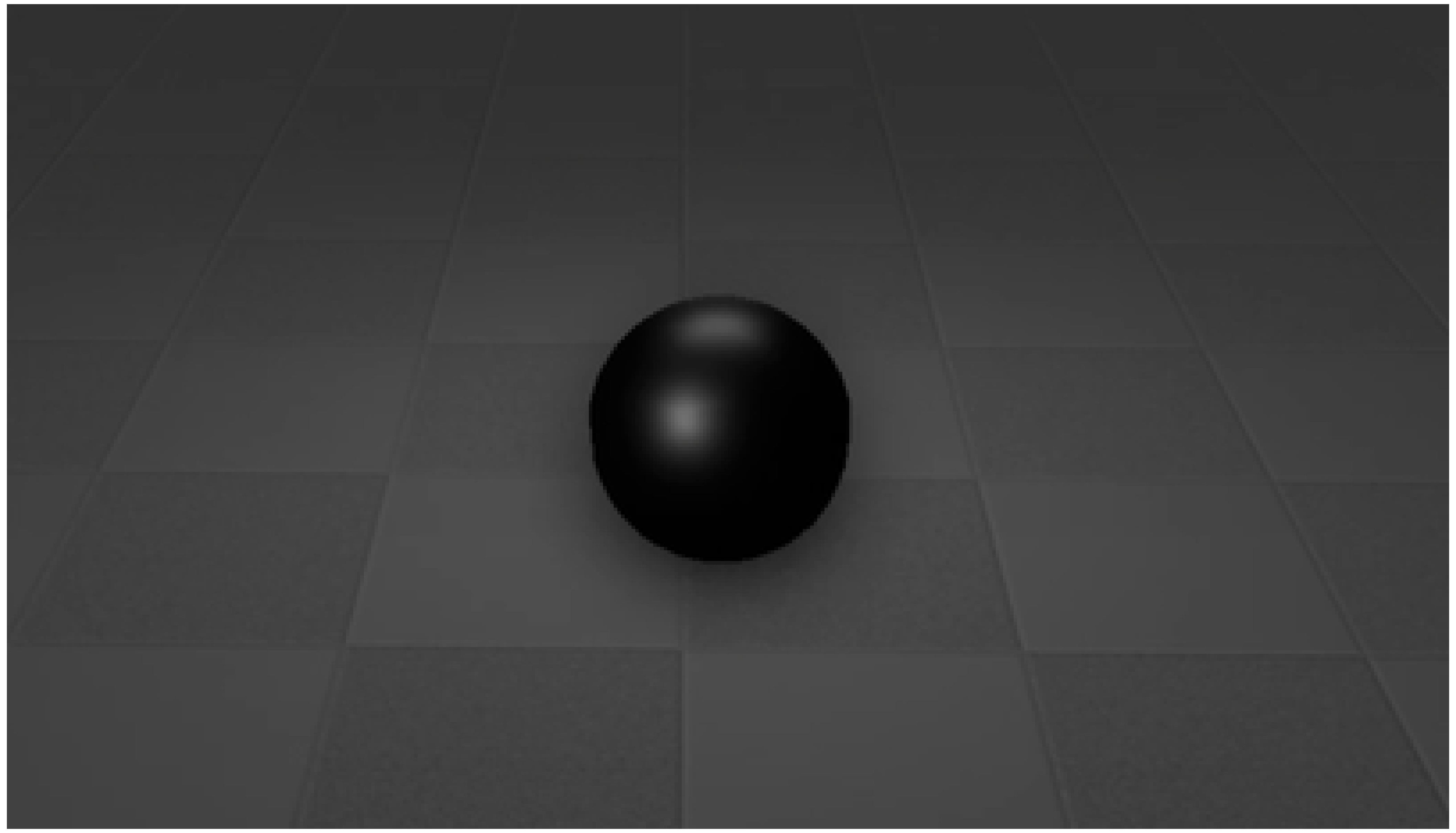
Abstraction



TAKEAWAYS

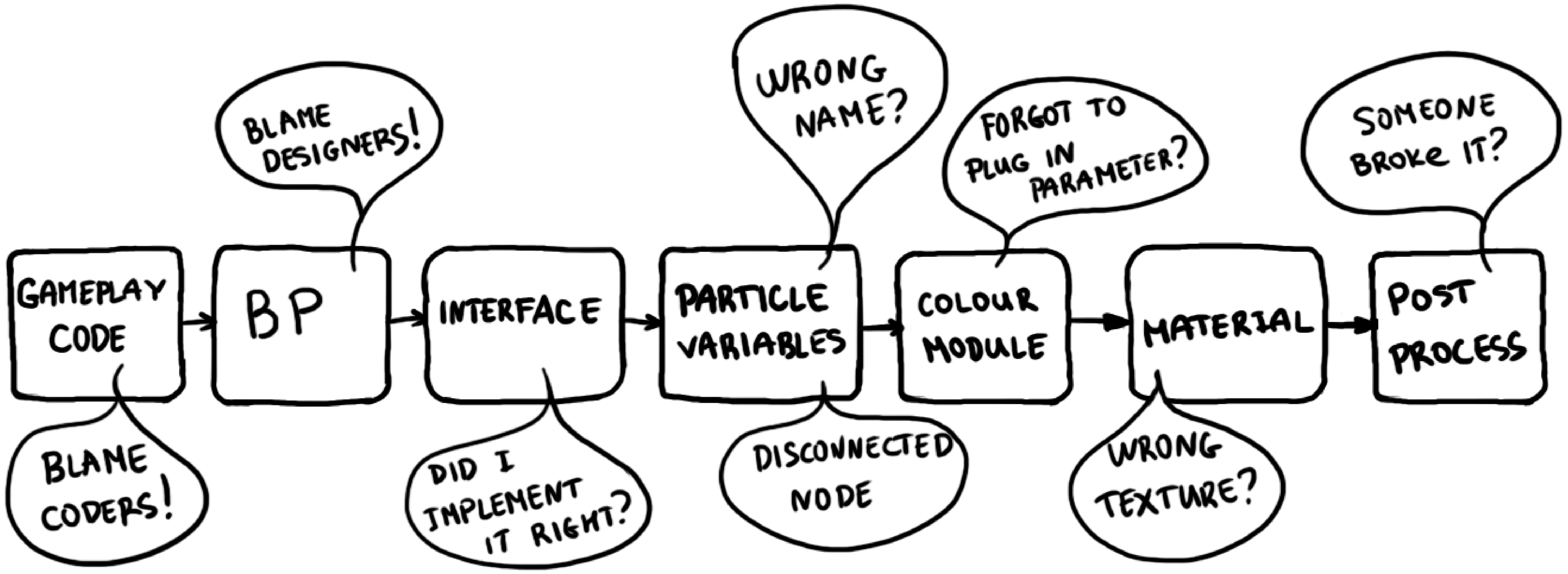
Understanding the system





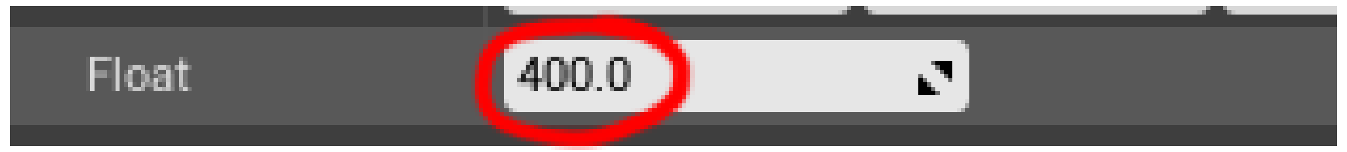
TAKEAWAYS

Debug principles

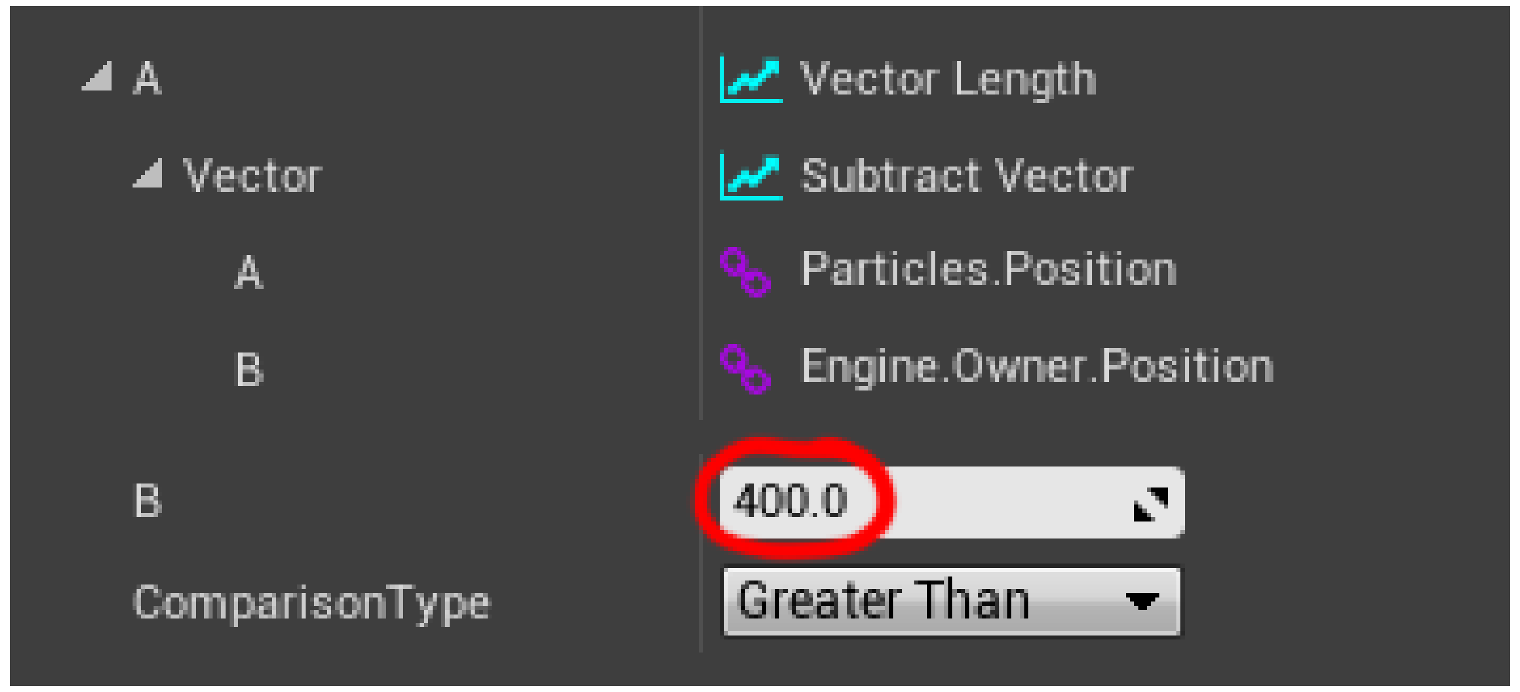


TAKEAWAYS

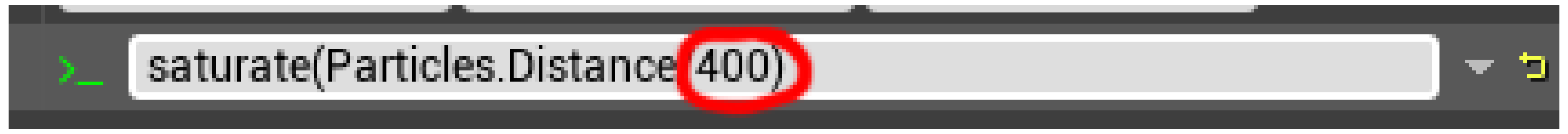
Extension and support



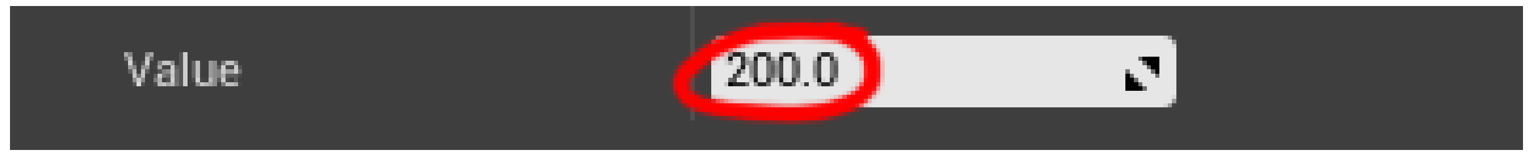
Float	400.0



◢ A	Vector Length
◢ Vector	Subtract Vector
A	Particles.Position
B	Engine.Owner.Position
B	400.0
ComparisonType	Greater Than



```
saturate(Particles.Distance 400)
```



Value	200.0

TAKEAWAYS

- Abstraction
- Understanding the system
- Debug principles
- Extension and support
- Self-discipline

THANK YOU!

- Creative Assembly
- Sarah Grissom, Christina Wun and Mike Lyndon
- All of you!

Questions? @vfxana