

GDC

# The Audio Technology of



Guy Somberg

GAME DEVELOPERS CONFERENCE

MARCH 16-20, 2020 | #GDC20

# The Audio Technology of



Guy Somberg

The GDC logo is positioned at the top center of the slide. It features the letters 'GDC' in a bold, white, sans-serif font. The logo is partially overlaid by a dark blue diamond shape that points downwards. The background of the slide is a solid teal color, with two thin, light blue diagonal lines crossing from the corners towards the center. There are also several small, white geometric shapes (diamonds and squares) scattered around the edges of the slide.

GDC

# Short and Snappy Title

Long and Boring Subtitle

A large, pink triangle with a gradient from light pink at the top to a darker pink at the bottom is positioned at the bottom center of the slide. It is partially overlaid by the text and the GDC logo.

**GAME DEVELOPERS CONFERENCE**  
MARCH 16–20, 2020 | #GDC20

The GDC logo is centered at the top of the slide. It consists of the letters "GDC" in a bold, white, sans-serif font. The letters are set against a dark blue, diamond-shaped background that is part of a larger geometric design. The overall background of the slide is a light teal color, with thin, dark teal lines forming a large 'X' shape that divides the space into four quadrants. In the top-left and bottom-right corners, there are small, dark teal diamond shapes. In the top-right and bottom-left corners, there are small, dark teal squares arranged in a 2x2 grid.

# GDC

# ???

The Audio Technology of Torchlight 3

A large, stylized pink triangle is positioned at the bottom center of the slide, pointing upwards. It has a gradient from a darker pink at the top to a lighter pink at the bottom. The text "GAME DEVELOPERS CONFERENCE" is written in a bold, dark blue, sans-serif font across the middle of the triangle. Below it, the text "MARCH 16-20, 2020 | #GDC20" is written in a smaller, dark blue, sans-serif font.

**GAME DEVELOPERS CONFERENCE**  
MARCH 16-20, 2020 | #GDC20

The GDC logo is positioned at the top center of the slide. It features the letters 'GDC' in a bold, white, sans-serif font. The logo is partially overlaid by a dark blue geometric shape that resembles a stylized 'V' or a mountain peak. The background of the slide is a vibrant teal color, decorated with thin white lines forming a large 'X' and several small white diamond shapes.

# Your Pet Has Returned

The Audio Technology of Torchlight 3

**GAME DEVELOPERS CONFERENCE**

MARCH 16–20, 2020 | #GDC20

The GDC logo is positioned at the top center of the slide. It features the letters 'GDC' in a bold, white, sans-serif font. The logo is partially overlaid by a dark blue diamond shape that points downwards. The background of the slide is a vibrant teal color, decorated with thin, light blue diagonal lines forming a large 'X' shape. Small white geometric icons, including squares and diamonds, are scattered in the corners.

# Taming the Chaos

The Audio Technology of Torchlight 3

**GAME DEVELOPERS CONFERENCE**

MARCH 16–20, 2020 | #GDC20

The GDC logo is centered at the top of the slide. It consists of the letters "GDC" in a bold, white, sans-serif font. The letters are set against a dark blue, diamond-shaped background that is part of a larger geometric design. The overall background of the slide is a light teal color, with thin, dark teal lines forming a large 'X' shape that divides the space into four quadrants. In the top-left and bottom-right corners, there are small, dark teal diamond shapes. In the top-right and bottom-left corners, there are small, dark teal squares arranged in a 2x2 grid.

# GDC

# ???

The Audio Technology of Torchlight 3

The Game Developers Conference logo is located at the bottom center of the slide. It features a large, pink, upward-pointing triangle that is partially obscured by a smaller, dark teal triangle. The text "GAME DEVELOPERS CONFERENCE" is written in a bold, dark teal, sans-serif font across the middle of the pink triangle. Below this, the text "MARCH 16-20, 2020 | #GDC20" is written in a smaller, dark teal, sans-serif font.

**GAME DEVELOPERS CONFERENCE**  
MARCH 16-20, 2020 | #GDC20

# About Guy

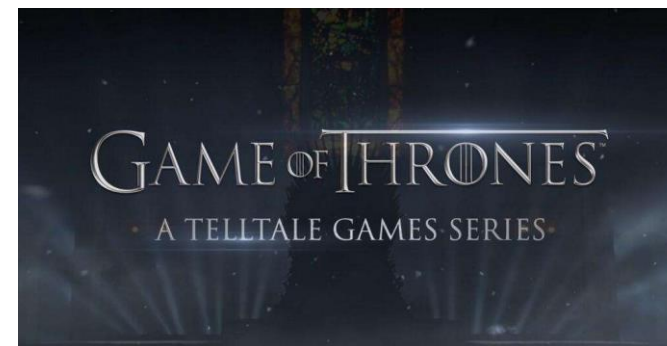
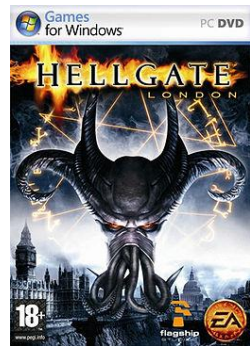
- In games since 2002
- Owned the audio engine at (nearly) every company





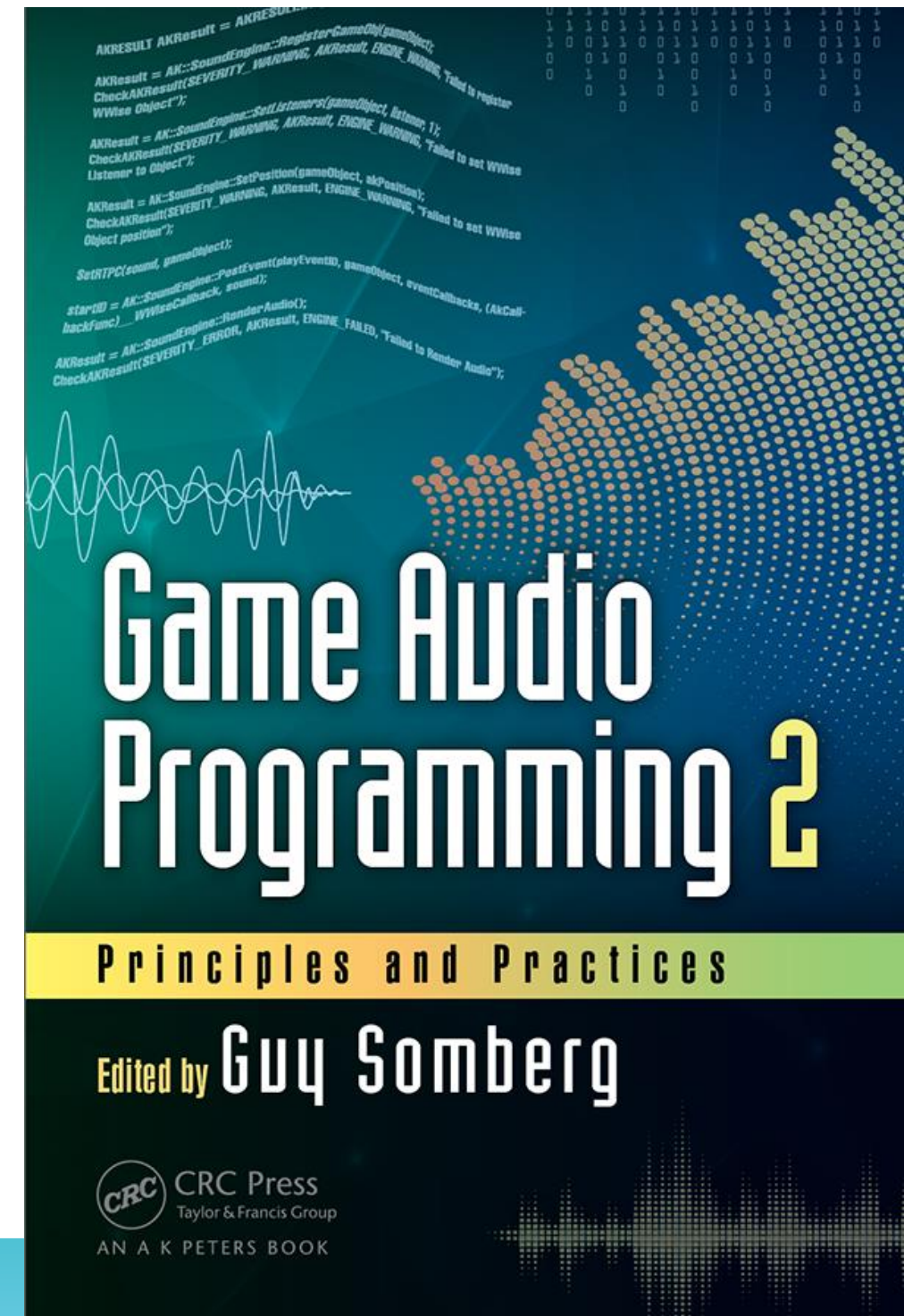
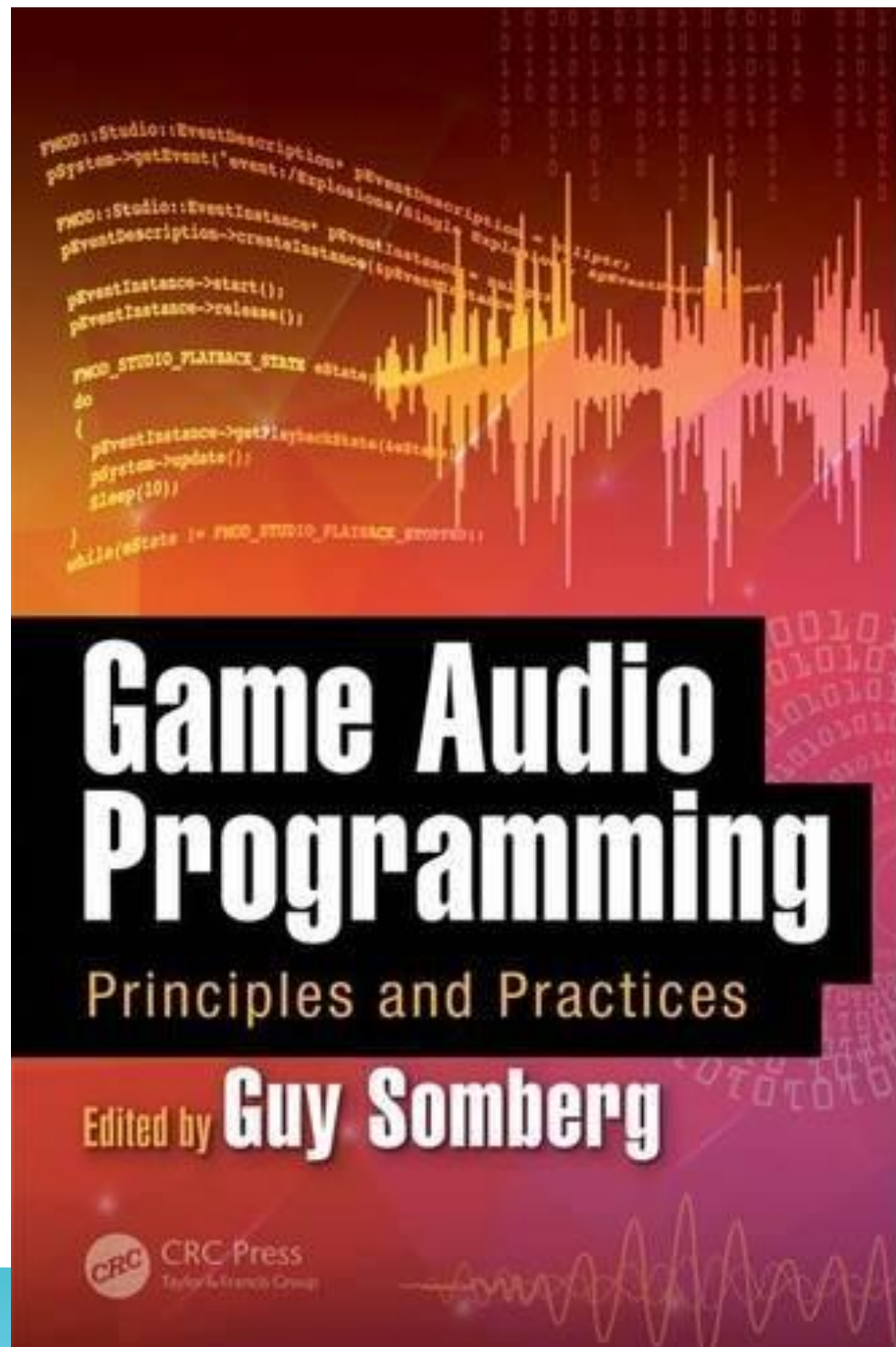
# About Guy

- ...and shipped lots of games

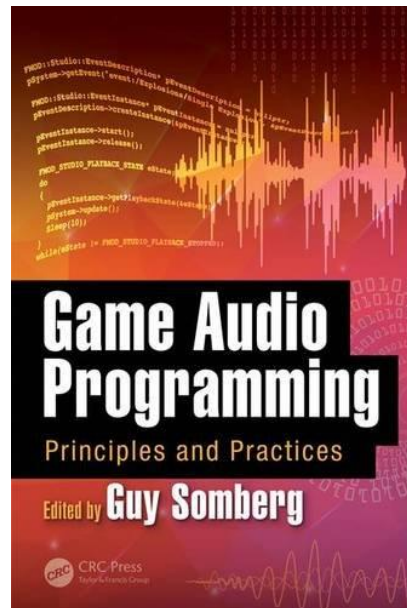




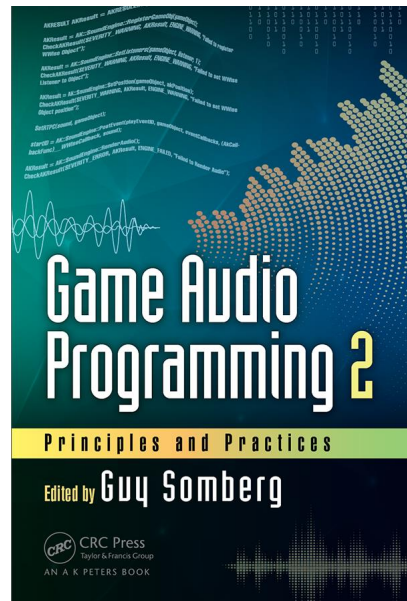
# Shameless Plug



# Shameless Plug



<https://www.crcpress.com/Game-Audio-Programming-Principles-and-Practices/Somberg/p/book/9781498746731>

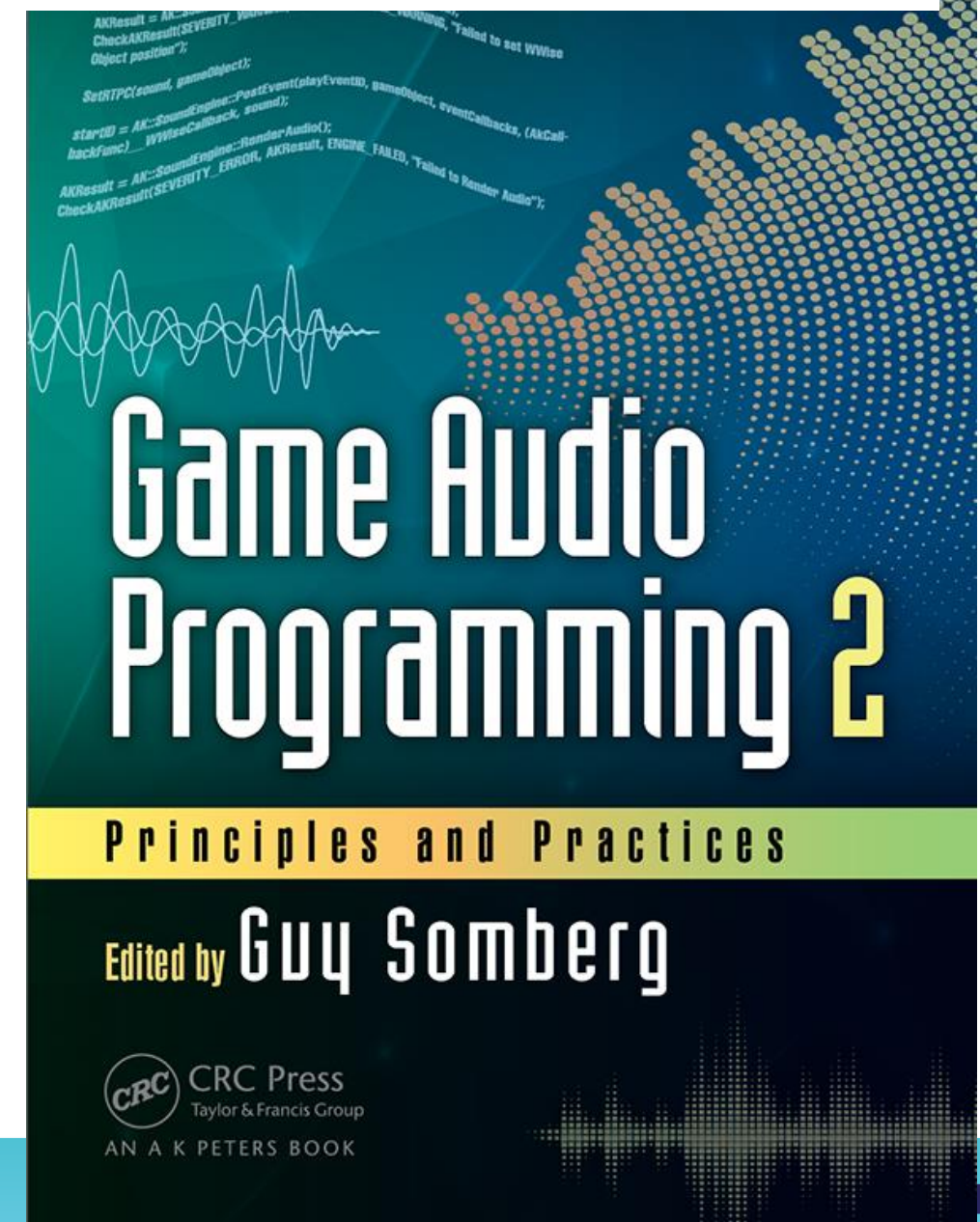
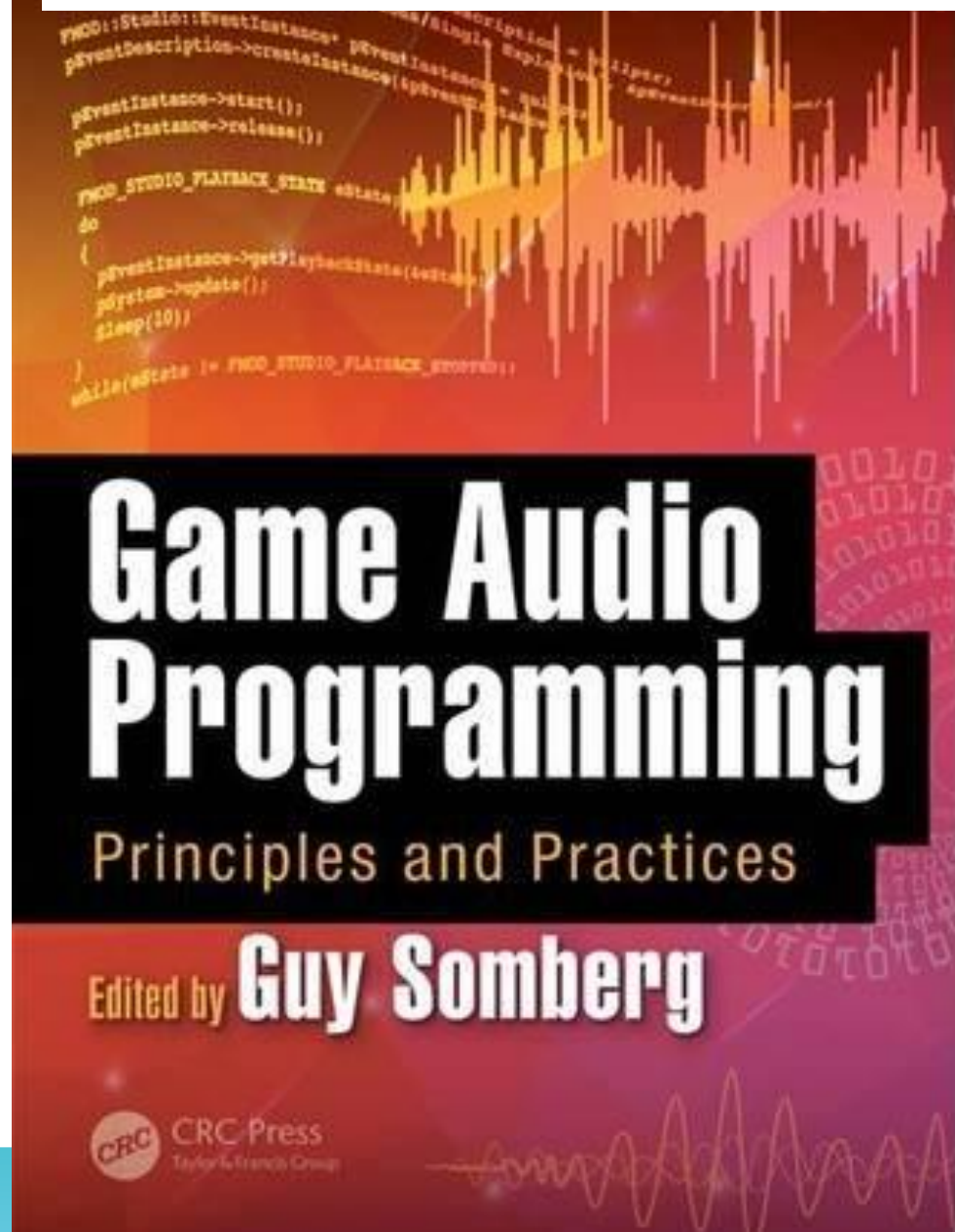


<https://www.crcpress.com/Game-Audio-Programming-2-Principles-and-Practices/Somberg/p/book/9781138068919>



# Shameless Plug

## Coming Soon: Volume 3!







# TORCHLIGHT FRONTIERS



# ARPGs are Hard

- Everything is happening on the screen in front of you
  - See also: RTS, Adventure Games, etc.
- Chaotic action
  - See also: FPS, MOBA, etc.
- Randomized level layout
  - See also: Roguelikes, Strategy, etc.

# Today's Topics

- Importance-Based Mixing
- Volumetric Sounds
- Screen-Space Distance Attenuation

Not talking about:

- Narrator
- Music
- Modified FMOD Studio Unreal Plugin
- Tool-time Bank Building
  - Other cool stuff that we've done

# Today's Topics

- Importance-Based Mixing
- Volumetric Sounds
- Screen-Space Distance Attenuation



# Mixing Woes

- Fundamental Problem:
  - Chaotic mix: dozens of events playing all at once
  - Traditional mixing techniques: Snapshots, Prioritization, Culling, HDR
    - All useful, and we use most of them
  - All insufficient when faced with our game

# Early Attempts

- Special case: “Nearby monster count”
- Offline: Categorization
- Didn’t really solve the problem



The  
Chasm  
Level: 5

RedOctober\_NOLOGIN0







The  
Chasm  
Level: 5

RedOctober NOLOGIN0



# Solution: Importance

- In Torchlight 3, **Importance** is the most important mixing technique that we use
  - Thanks to **Tomas Neumann** and **Paul Lackey** from the Overwatch team!

# Importance

- Assign each object an importance score
- Sort all objects by score
- Place sorted objects into buckets
- Apply effect to each sound in the bucket

# Importance Scores

- Extremely game-specific
- In Torchlight 3:
  - Identity
  - Distance to player
  - Relative “drama” score
  - Skill target
- Total score = weighted sum of individual scores





The  
Chasm  
Level: 5

RedOcto OGINO

0.88

0.84

0.5

0.65

2.13

2.27

0.64

1.99

1.5

1.7

3.0

1.53

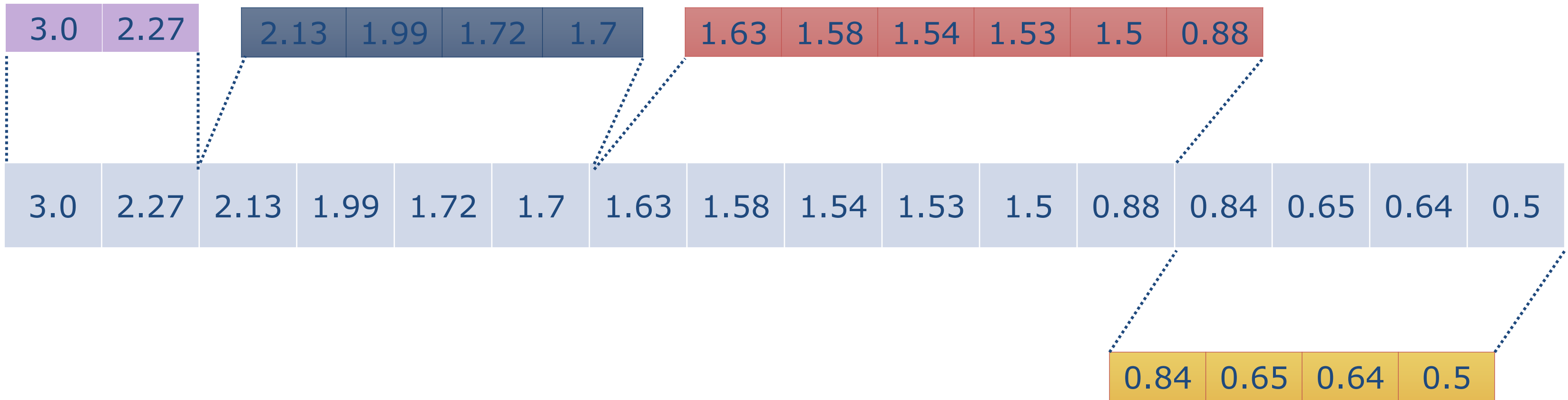
1.72

1.58

1.63

1.54







The Chasm  
Level: 5

RedOcto OGINO

0.88

0.84

0.5

0.65

2.13

0.64

2.27

1.99

1.5

1.7

3.0

1.53

1.72

1.58

1.63

1.54



# Importance Bucket Effects

- Volume, Peaking Filter, High Shelf Filter
  - Implementation detail: Peaking and High Shelf Filters implemented using Multiband EQ

Priority	Effect
1	Peaking Filter
2	No change
3	Volume Reduction
4	Volume Reduction, High Shelf
5	Volume Reduction, High Shelf

# Importance in Action

- Live Demo

# Debug Display

[CATEGORY: AudioImportance]

PRIORITY 1

rm\_f\_player\_C\_0 - 3.00  
gobchanter\_b\_C\_2 - 1.99  
goblin\_brute\_b\_C\_2 - 2.13  
goblin\_brute\_b\_C\_3 - 2.27  
gobgeneric\_stabby\_b\_C\_4 - 1.72

PRIORITY 2

gobgeneric\_stabby\_b\_C\_7 - 1.55  
Railman\_Turret\_Actor1a\_C\_0 - 1.53  
gobgeneric\_stabby\_b\_C\_5 - 1.54  
Railman\_FreightCar\_Actor\_C\_0 - 1.58  
Railman\_CabooseCar\_Actor\_C\_0 - 1.63  
peteagle\_body\_03\_harpy\_bp\_C\_0 - 1.70

PRIORITY 3

dm\_m\_player\_C\_0 - 0.88  
petcat\_tuxedo\_bp\_C\_0 - 0.64  
gob\_prop\_torch\_02\_bp4 - 0.84  
gcave\_break\_minecart\_03\_bp\_C\_0 - 0.65  
gob\_prop\_tool\_hammer\_01\_bp\_C\_0 - 0.71

PRIORITY 4

gob\_prop\_torch\_02\_bp\_2 - 0.00  
gob\_prop\_torch\_02\_bp2 - 0.00  
gob\_prop\_torch\_02\_bp3 - 0.00  
gob\_prop\_torch\_02\_bp4 - 0.00  
gob\_prop\_torch\_02\_bp\_2 - 0.00  
gob\_prop\_torch\_02\_bp2\_5 - 0.00  
gob\_prop\_torch\_02\_bp3\_8 - 0.00  
gob\_lootable\_cauldron\_01\_bp\_2 - 0.00  
gob\_prop\_tool\_axe\_01\_bp\_C\_0 - 0.46  
gob\_prop\_tool\_pickaxe\_01\_bp\_C\_0 - 0.50



# Debug Display





# Debug Display



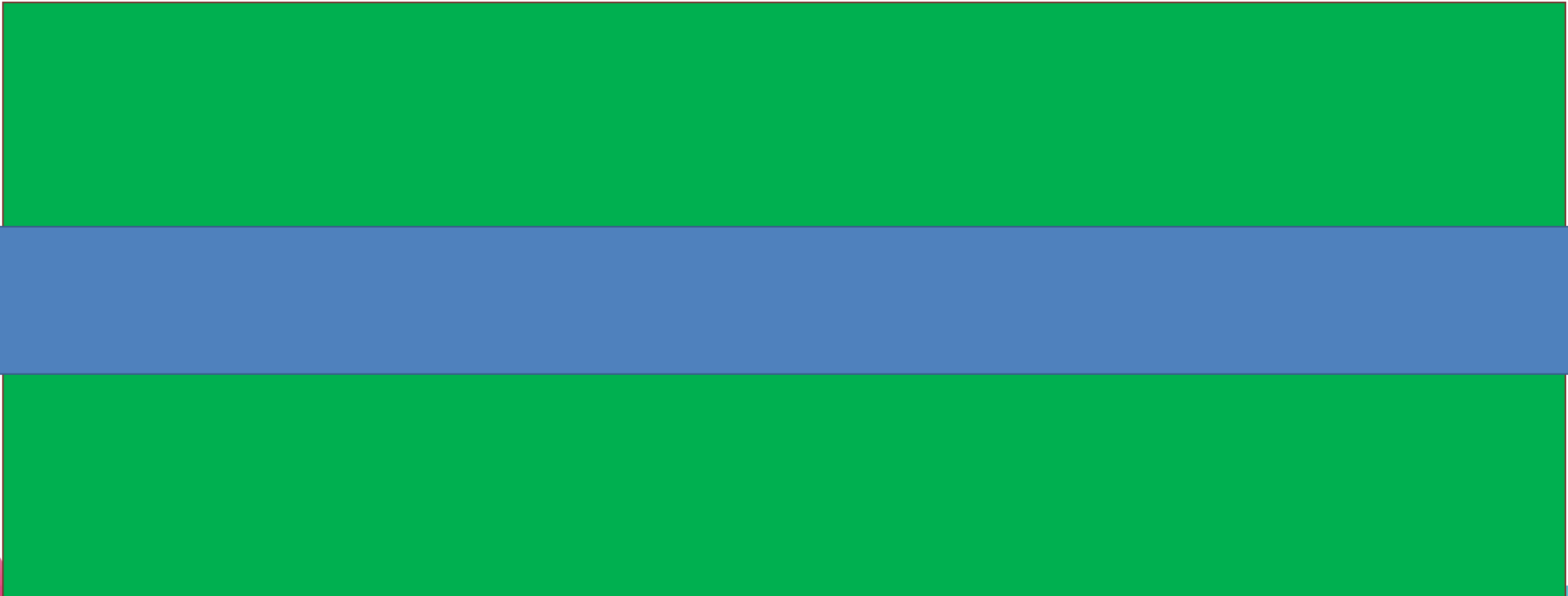
# Today's Topics

- Importance-Based Mixing
- **Volumetric Sounds**
- Screen-Space Distance Attenuation



# The Problem

- In one word: rivers





# Failed Idea #1

- Single sound source at the river center

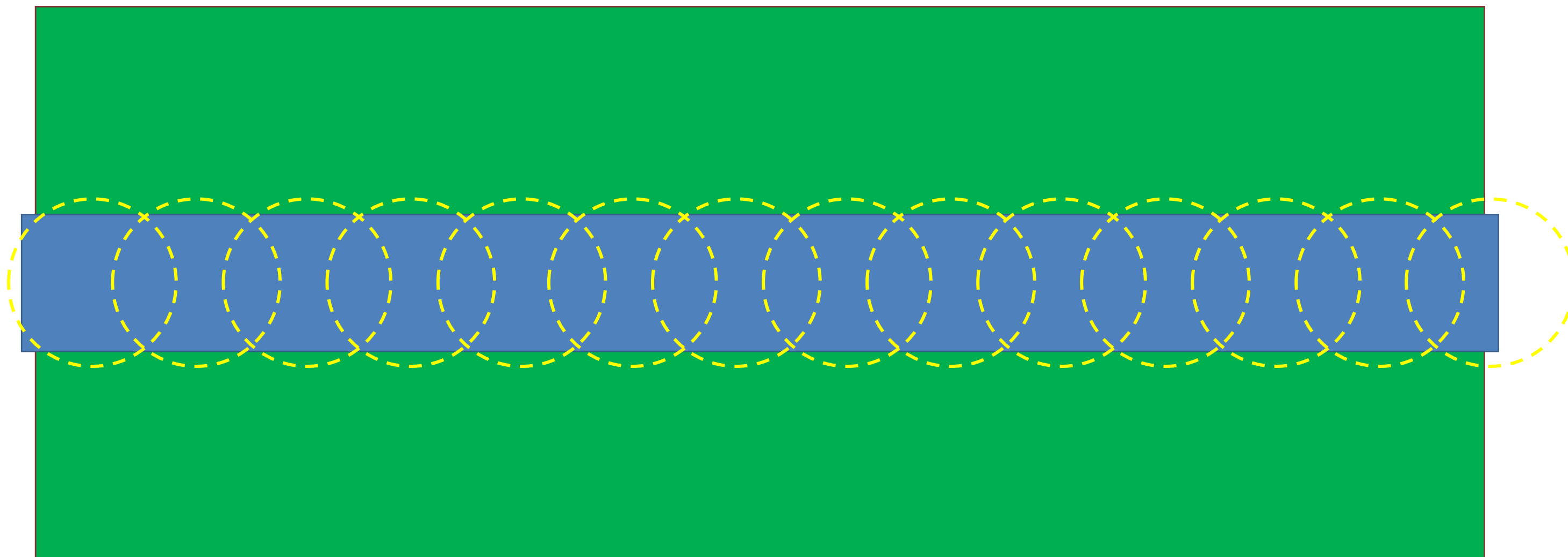


# Nope!

- No control over attenuation
- No control over shape
- Doesn't account for randomized levels

# Failed Idea #2

- Many smaller sound sources



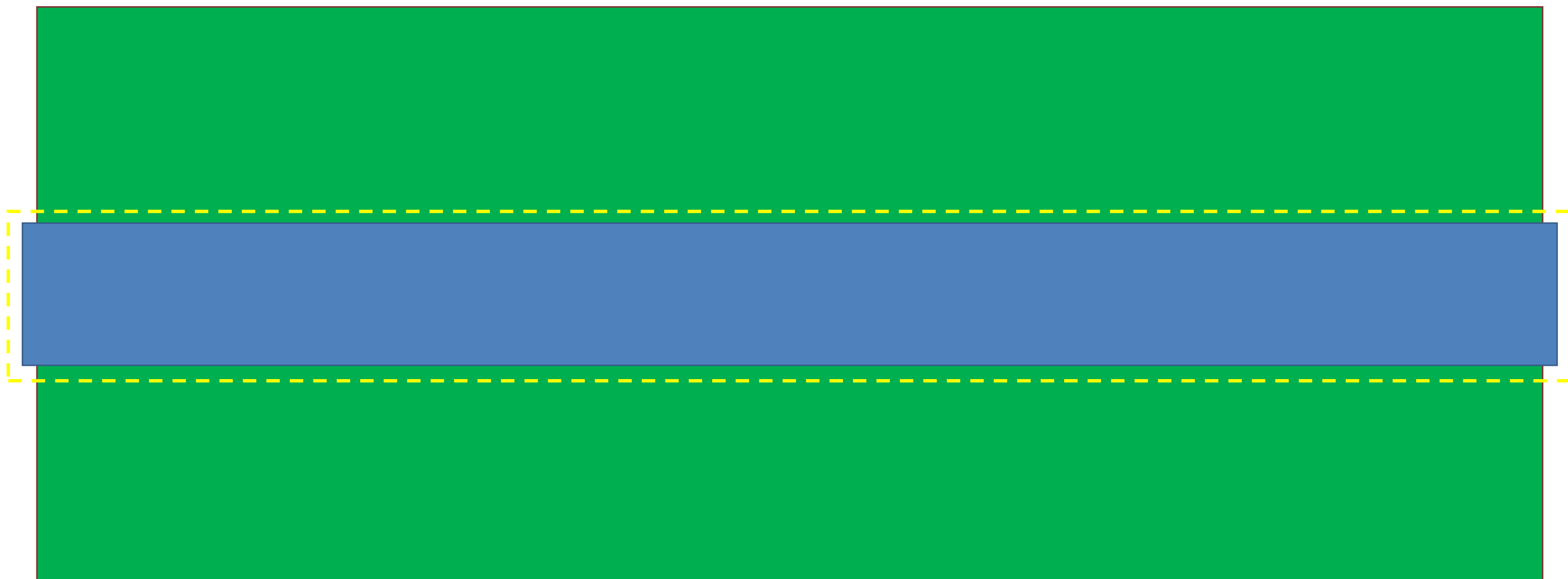
# Nope!

- Control over attenuation
- Control over shape
- Deals with random levels
- But:
  - Too many sound sources
  - Phasing
  - Mind-numbing setup



# Failed Idea #3

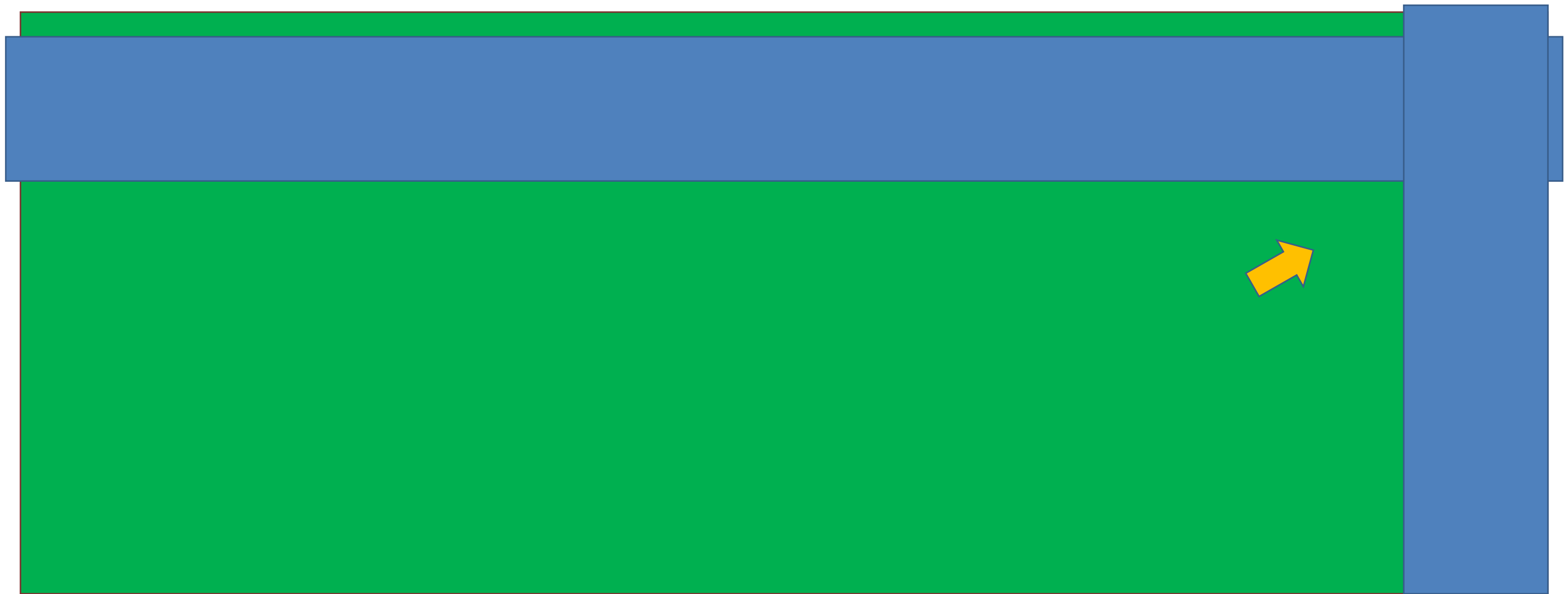
- Nearest point on box

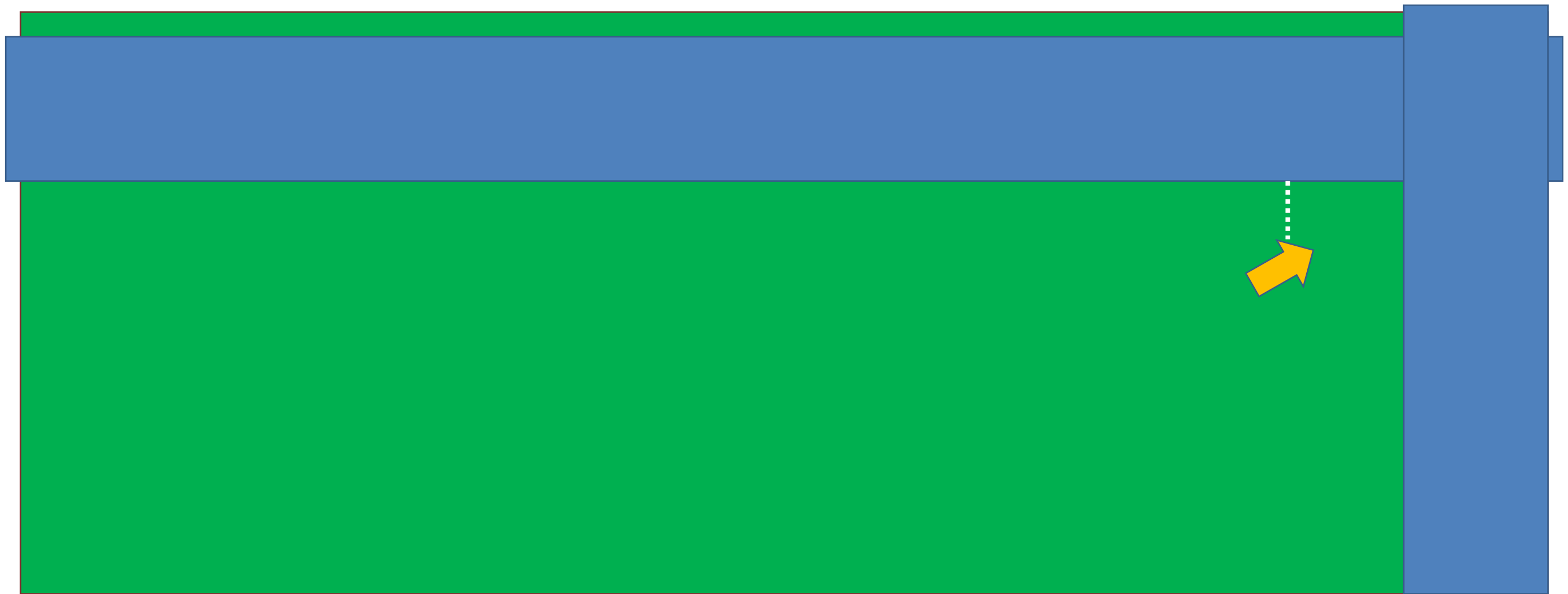


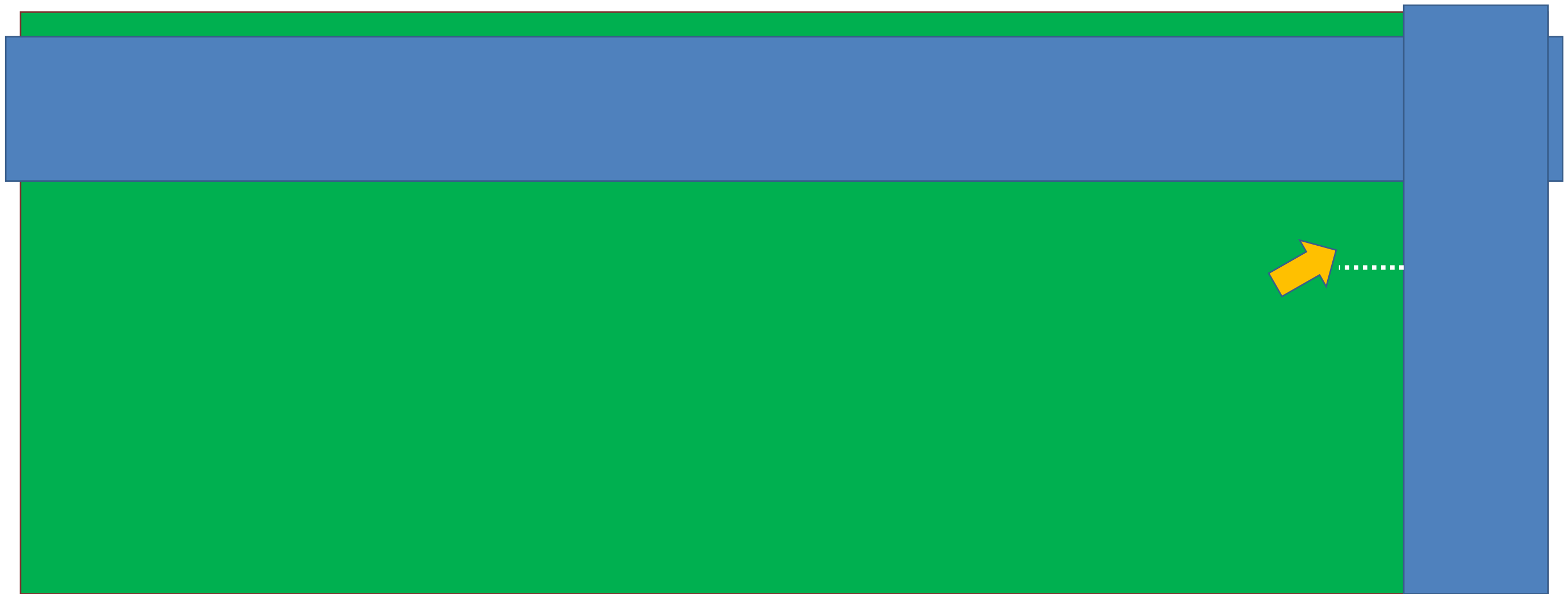
# Promising!

- Control over attenuation
- Control over shape
- Deals with random levels
- Only one sound source
- Easy setup

• But...









# Darn :(

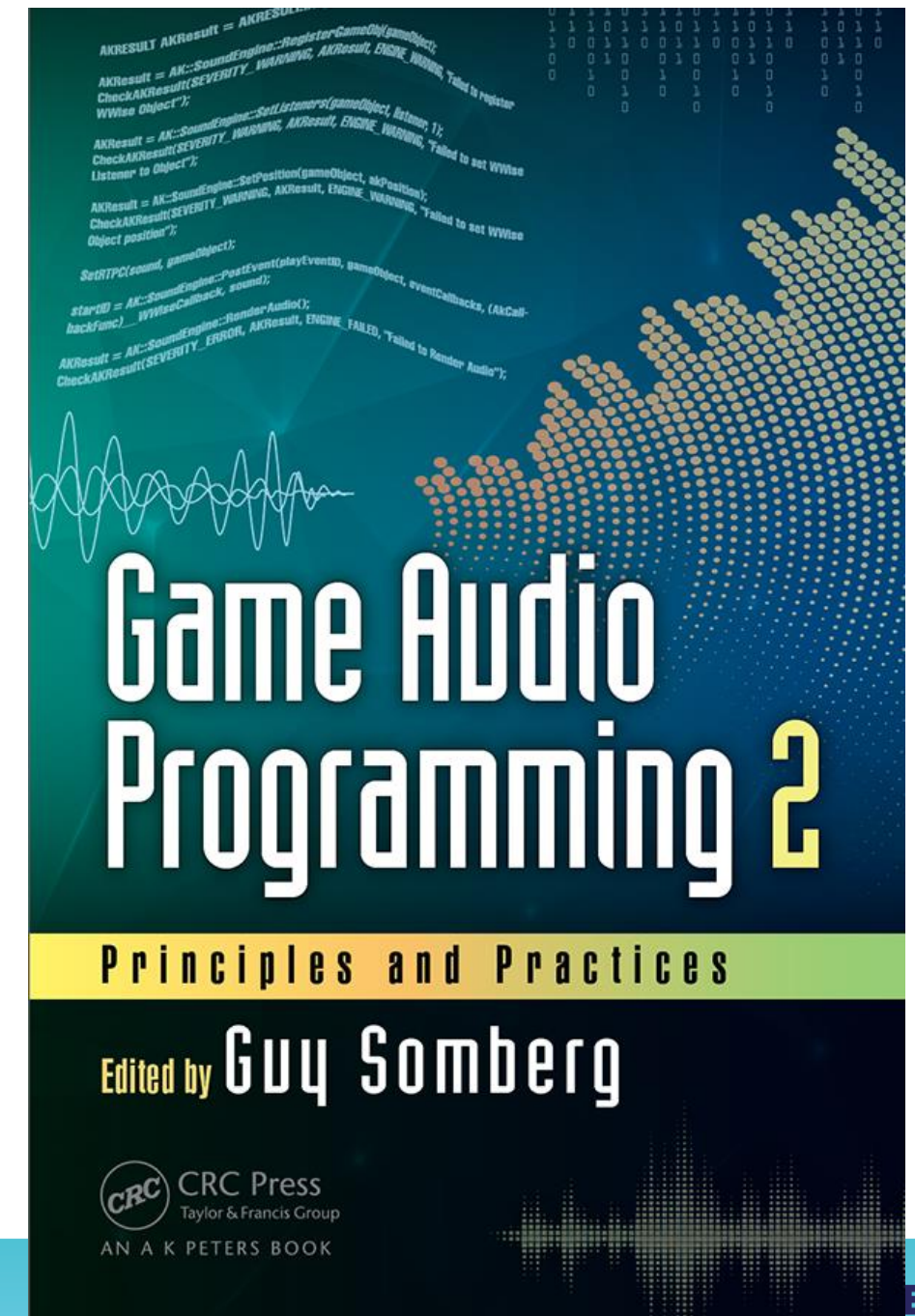
- So close!

# Take a Step Back

- We want one single sound source
- Position + Direction is insufficient to describe our sound source
- So what do we actually want to describe?

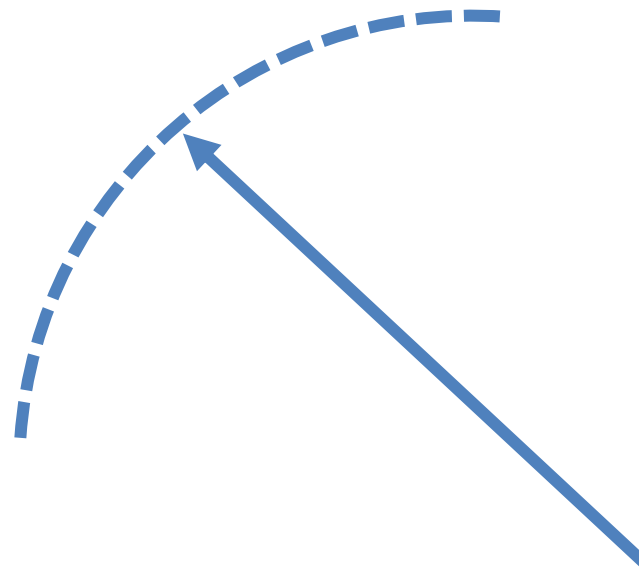
# Working Solution

- Game Audio Programming Principles and Practices Volume 2
- Chapter 12: "Approximate Position of Ambient Sounds of Multiple Sources" by Nic Taylor
- Thanks, Nic!



# Working Solution

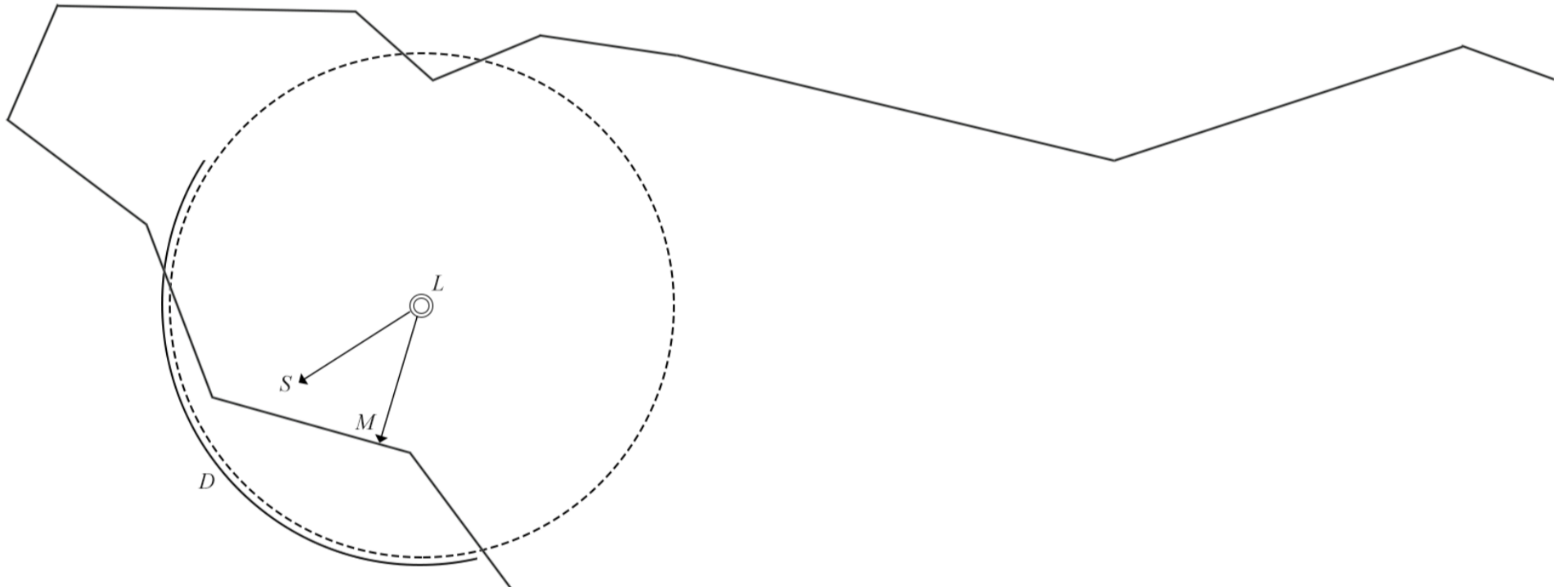
- Direction
- Magnitude
- Spread





# Analytical Solution

- Direction and Spread should be impacted less the farther a point is from the listener
- Spread of two parallel lines with listener equidistant should be 1 (full spread)
- Spread of a line segment that passes through the listener should be 0
- Small changes in listener position should result in small changes to Direction, Magnitude, and Spread
  - Subdividing a line segment should not alter Magnitude, Direction, or Spread



# Math!

$$\hat{\sigma} = \lim_{\Delta s_i \rightarrow 0} \sum_{i=1}^n \frac{\hat{v}}{\|\hat{v}\|} W(\hat{v}) \Delta s_i = \int_C \frac{\hat{v}}{\|\hat{v}\|} W(\hat{v}) \Delta s$$



# But...

- All of that is just for line segments
- Upgrading to two dimensions gets us both more features and more flexibility

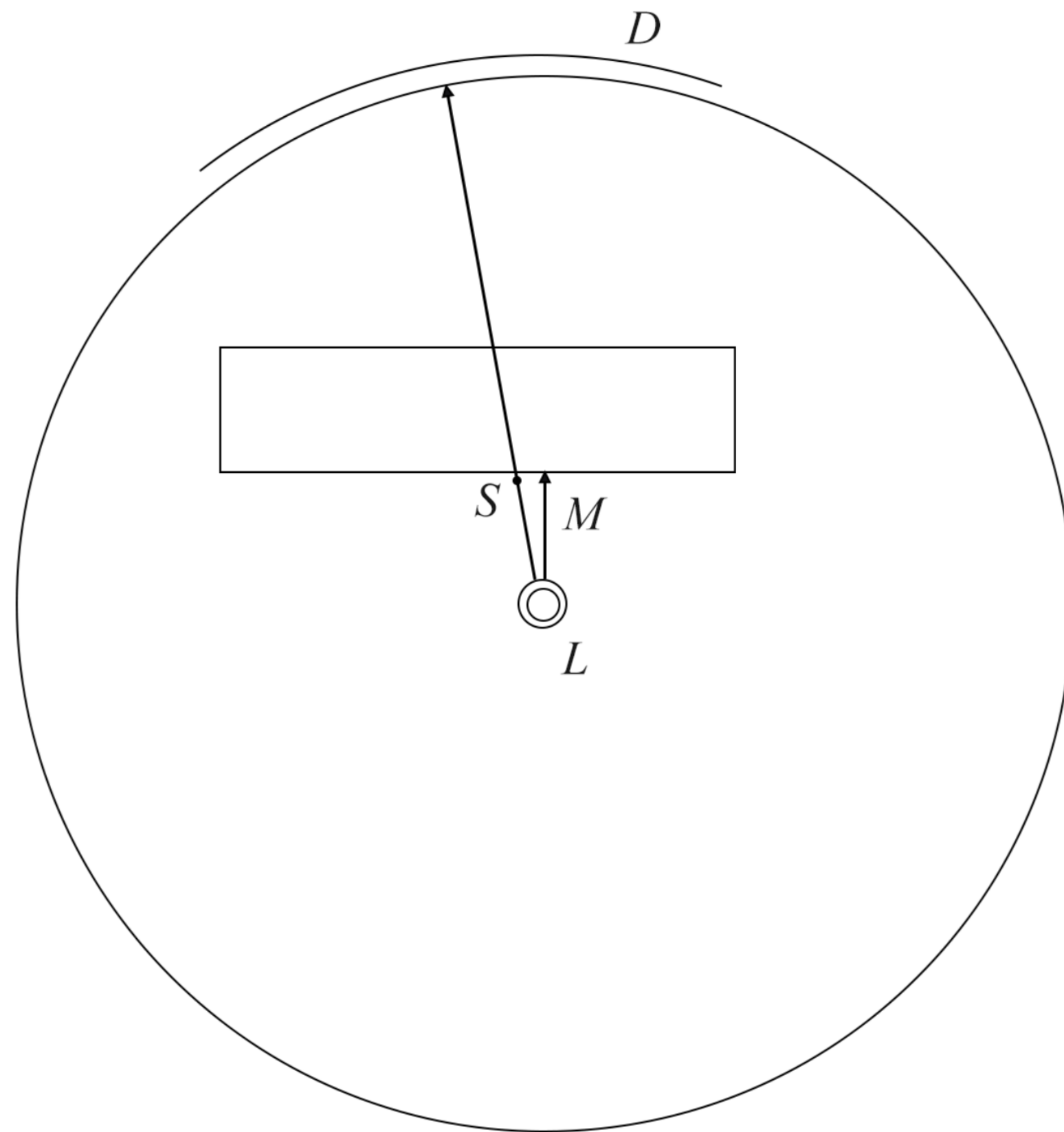
# More Math!

$$\hat{\sigma} = \lim_{\Delta s_i \rightarrow 0} \sum_{i=1}^n \frac{\hat{v}}{\|\hat{v}\|} W(\hat{v}) \Delta s_i = \int_C \frac{\hat{v}}{\|\hat{v}\|} W(\hat{v}) \Delta s$$

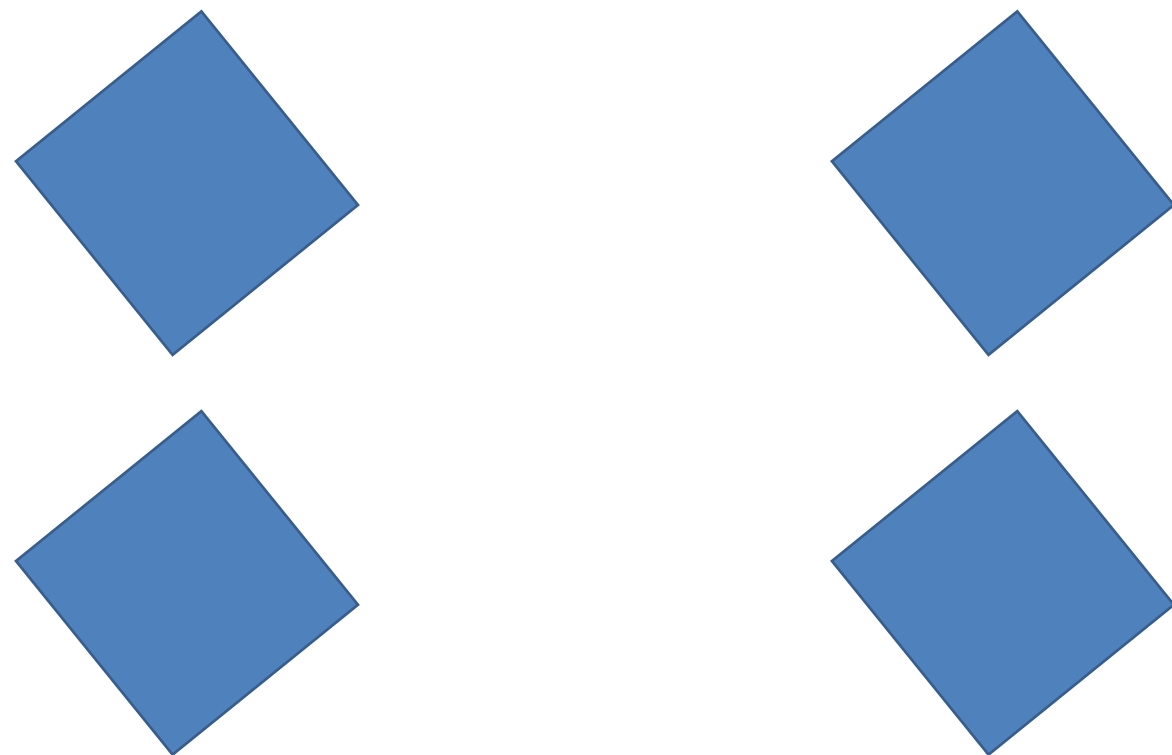
Becomes:

$$\hat{\sigma} = \iint_R \frac{\langle x, y \rangle}{\sqrt{x^2 + y^2}} \left( 1 - \sqrt{x^2 + y^2} \right) \Delta x \Delta y$$

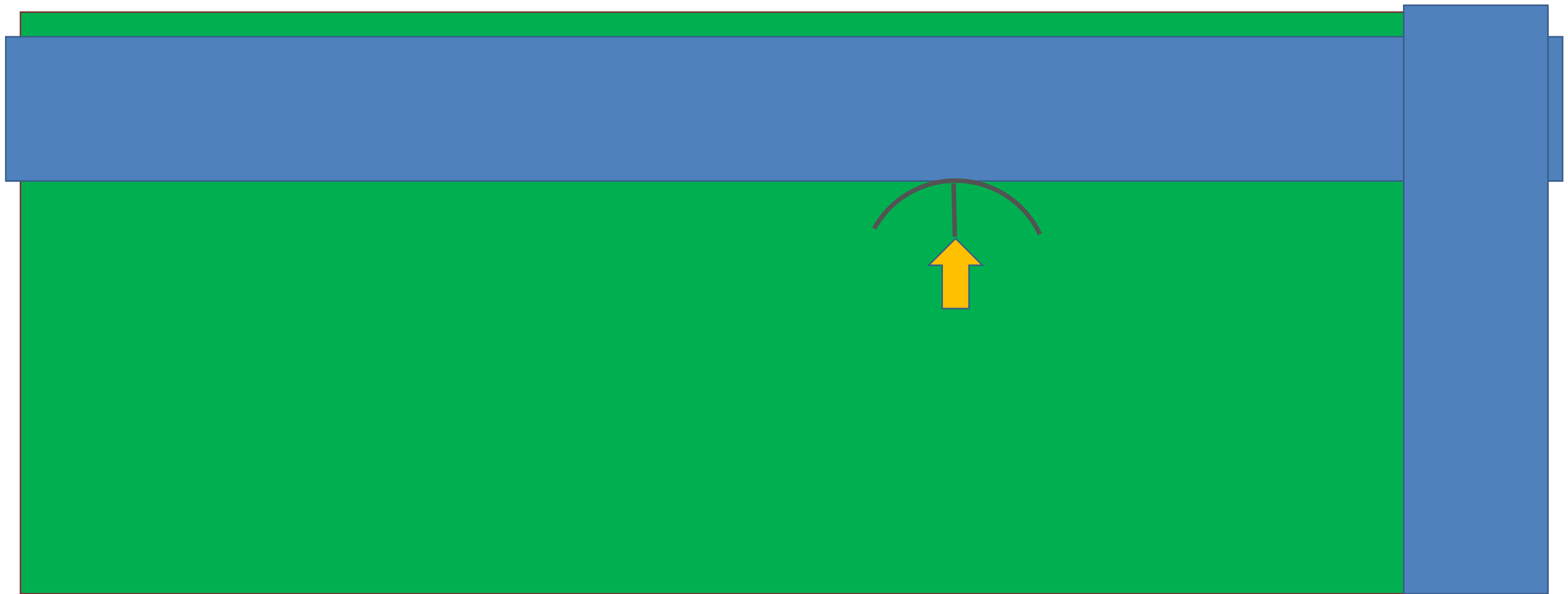
Image Credit: Nic Taylor









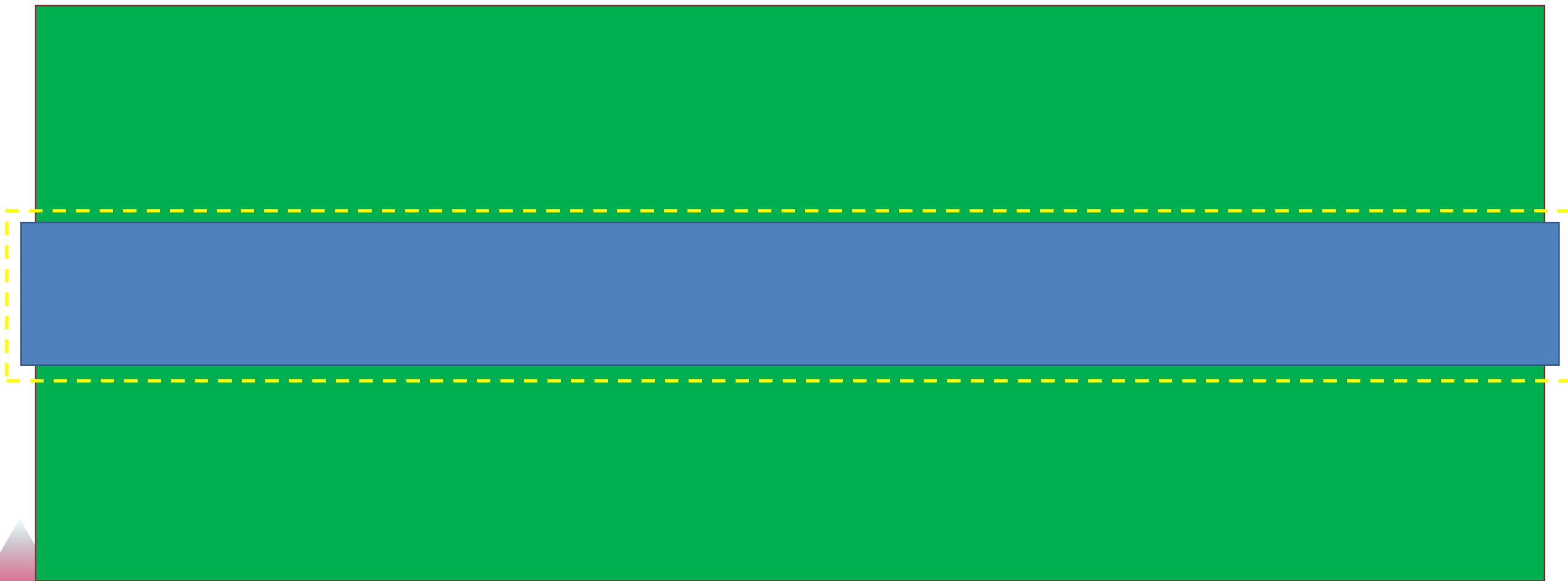






# Details: Setup

- Setup is easy:



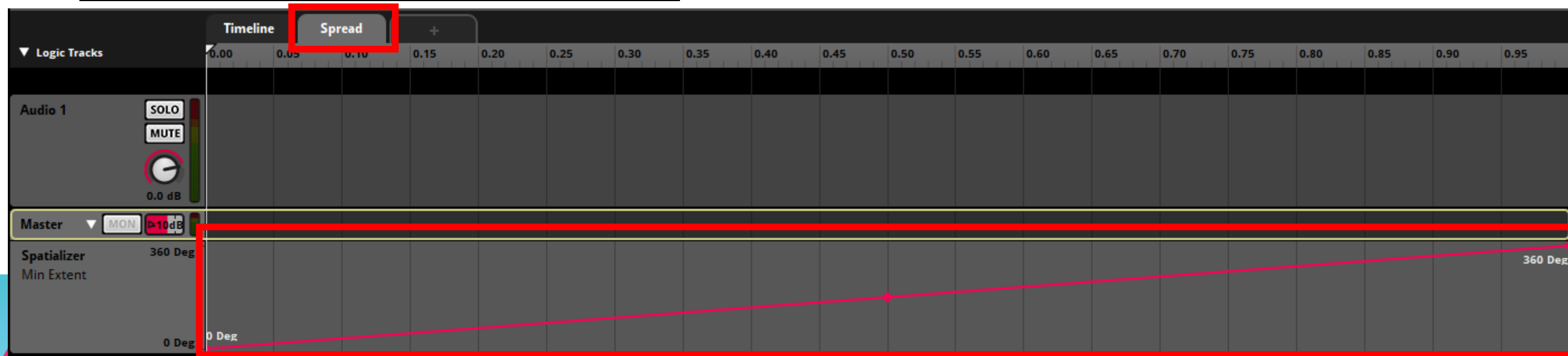
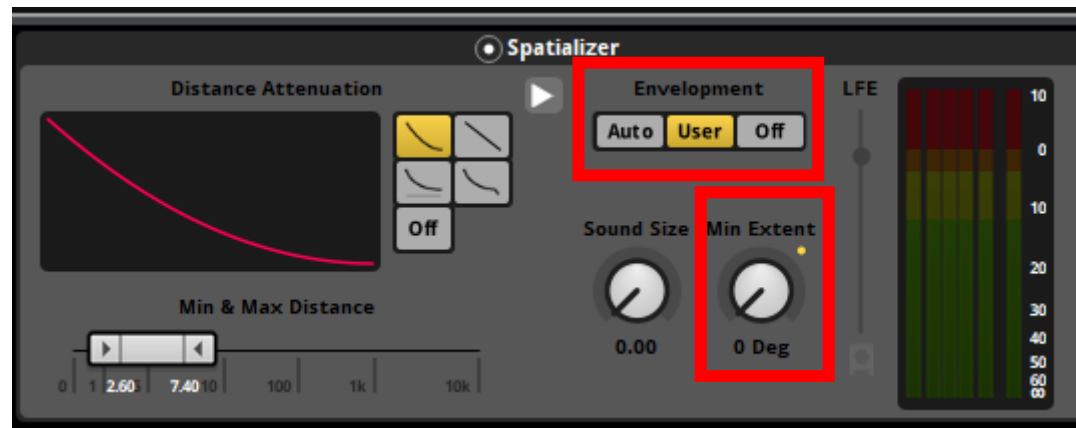
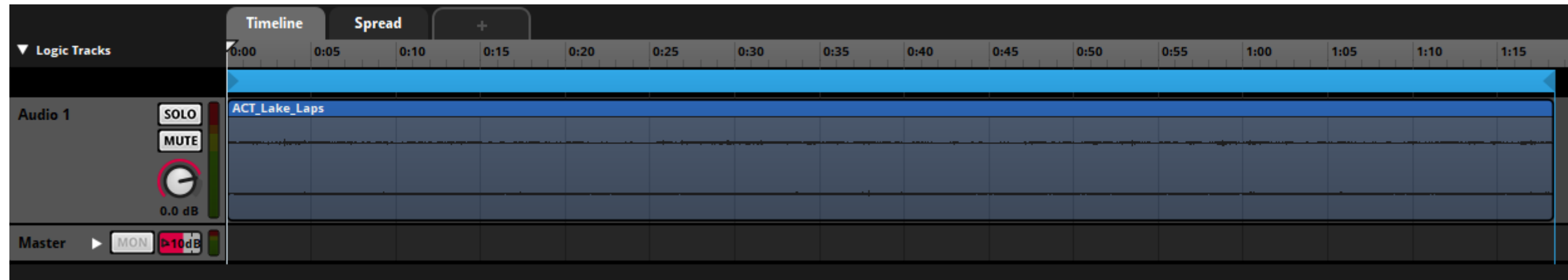
# Details: Box Subdivision

- Two solutions:
  - Subdivide boxes down
  - Build up the shape using voxels
- Both are covered in Game Audio Programming 2
- TL3 subdivides
  - (Minimum axis length – 0.5 meters)

# Details: Edge Cases

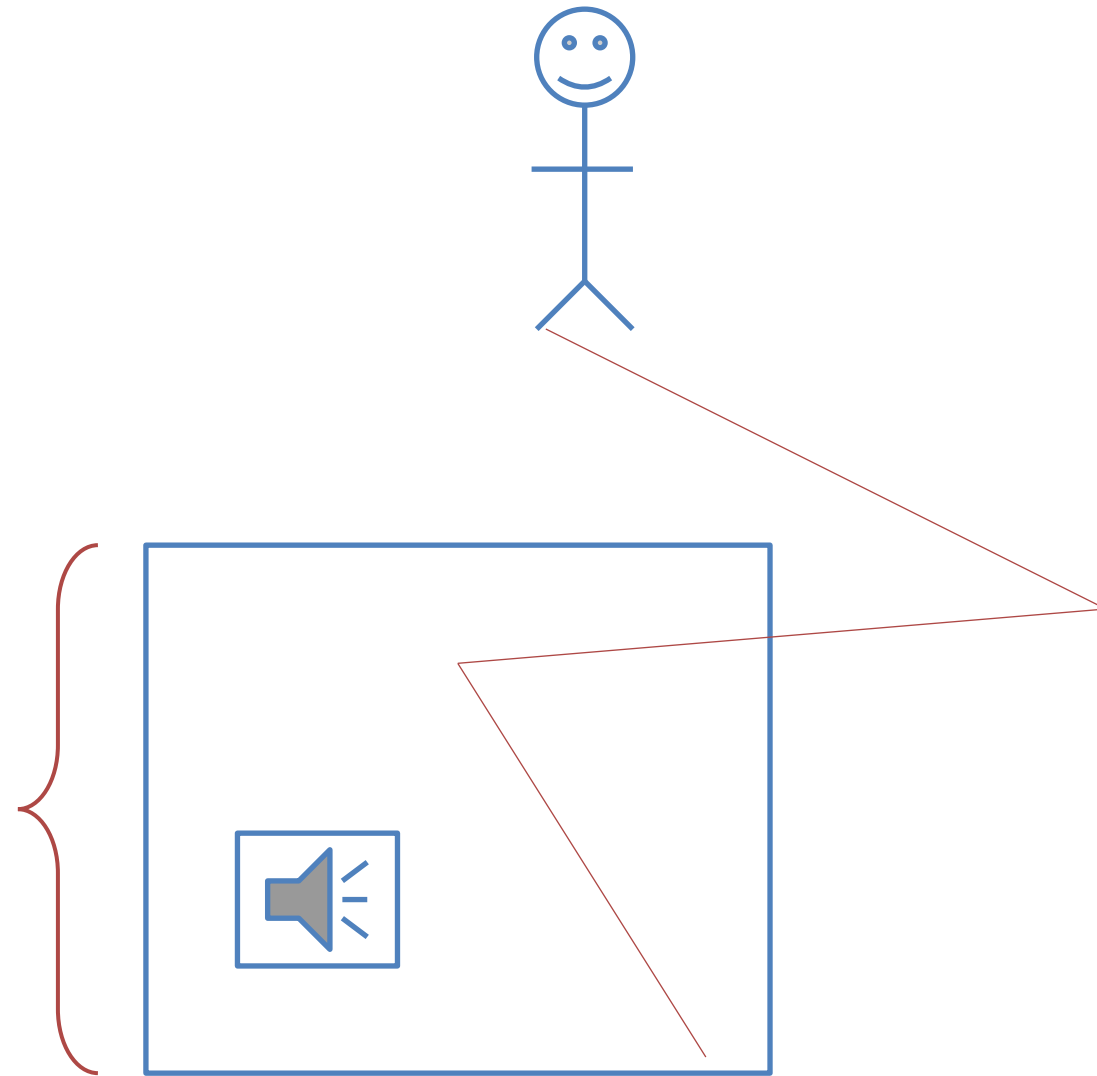
- Near field
  - As you get near to the volume, the sound needs to transition to full spread
- Popping
  - Add a seek speed to the direction and spread so that sudden changes don't pop
- Close to zero
  - Certain edge cases are close to zero, and can cause strange values to pop up.
  - TL3 still has a few of these unsolved

# Details: FMOD Studio Setup





# Details: Elevation



# Details: Debug Display

- Show as much detail as possible:
  - Subdivided Boxes
  - Direction
  - Spread arc

# Volumetric Sounds in Action

- Live Demo

# Today's Topics

- Importance-Based Mixing
- Volumetric Sounds
- Screen-Space Distance Attenuation



# A True Story



Me



Sound Designer

# A True Story



Me

I want an 'is on screen' parameter, please.



Sound Designer

# A True Story



Me

No, I won't do that for you.



Sound Designer

# A True Story



Me

What problem are you trying to solve?



Sound Designer

# A True Story



Me

I only want this sound to be audible if it's on the screen.



Sound Designer



# A True Story



Me

Isn't that what distance  
attenuation is for?



Sound Designer

# A True Story

?



Me

There is no meaningful value for max distance that will express the mix that I am trying to create.



Sound Designer

# A True Story



Me

What? Why not?

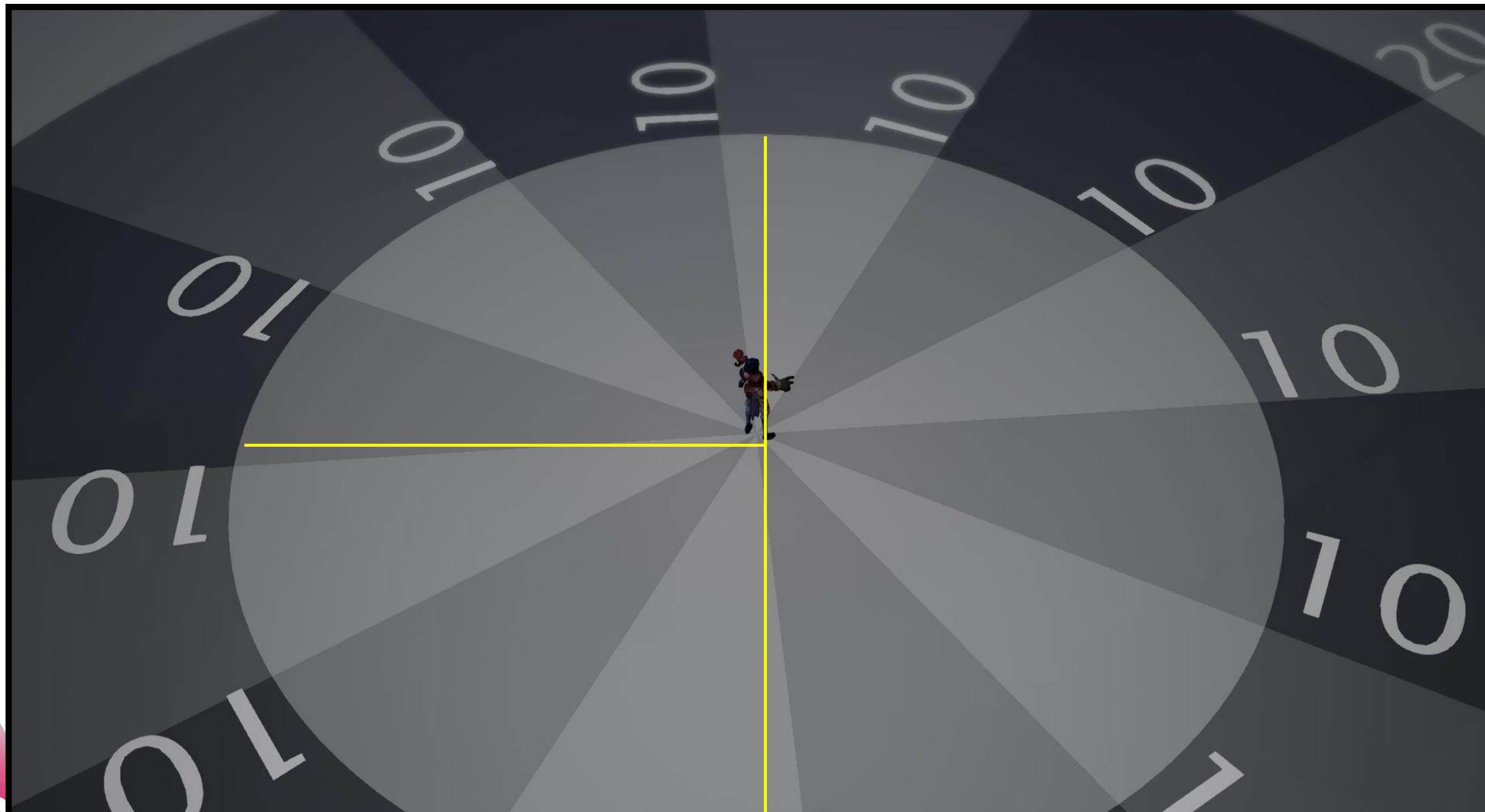


Sound Designer

# The Problem



# The Problem

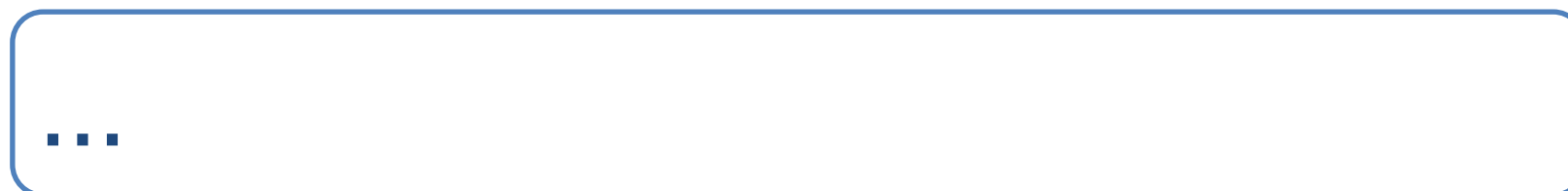




# A True Story



Me



Sound Designer

# A True Story



Me

Oh.



Sound Designer

# A True Story



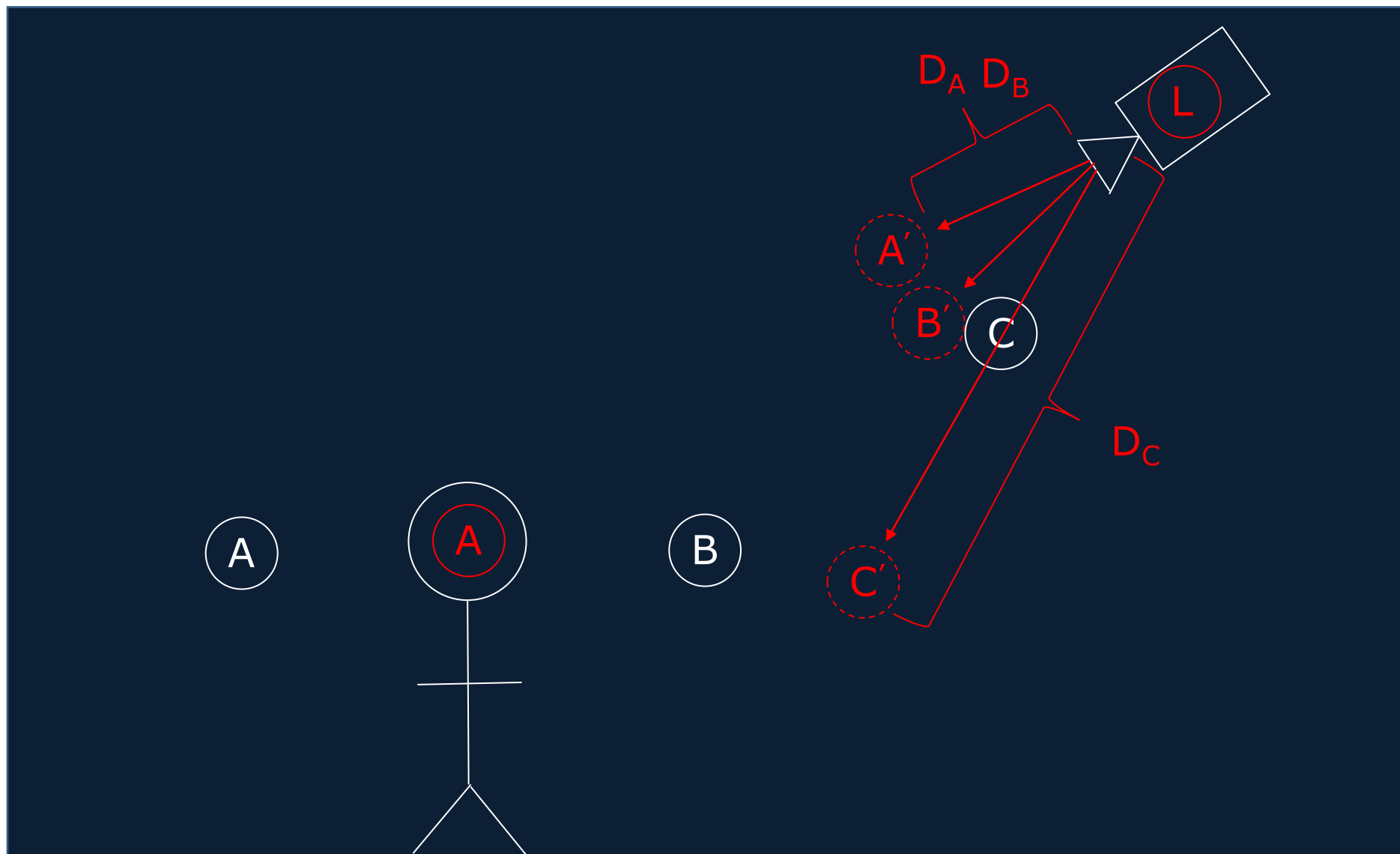
Me

Well, @#\$!@%.



Sound Designer

# This is a given:



# Original Listener Algorithm

- For each Channel
  - Calculate distance ( $d$ ) from Channel Position to Attenuation Position
  - Find normalized vector ( $\vec{N}$ ) from Listener Position ( $L$ ) to Channel position
  - Place actual playing Channel at  $L + d\vec{N}$

# Screen-Space Distance Attenuation

- Project points onto unclipped screen space.
- Make distance equal to the screen-space distance to attenuation position



# Adjusted Listener Algorithm

- For each Channel
  - Project Channel Position and Attenuation Position to screen space, and calculate 2D distance ( $d$ )
  - Find normalized vector ( $\vec{N}$ ) from Listener Position ( $L$ ) to Channel position
  - Place actual playing Channel at  $L + d\vec{N}$

# Implementation Details

- Scale is challenging
- Zooming in shouldn't affect attenuation
- Debug Display
- Every sound's distance attenuation must be re-authored

# Scale Issues

- By default, screen-space projections give you either 0..1 or -1..+1
- Sound designer tools don't like to work in such small scales
  - Also: Sound designers don't like to work in such small scales
- Multiply all positions/distances by an agreed-upon factor
  - Torchlight 3 uses a scale of 20

# Camera Zoom

- We can't just call `ProjectWorldToScreen()`



# Camera Zoom

- We can't just call `ProjectWorldToScreen()`



# Camera Zoom

- Project to where the camera would be if it were fully zoomed-out
- Need a function that will do the projection at a different location than the camera
  - (This is particularly useful for debug info)

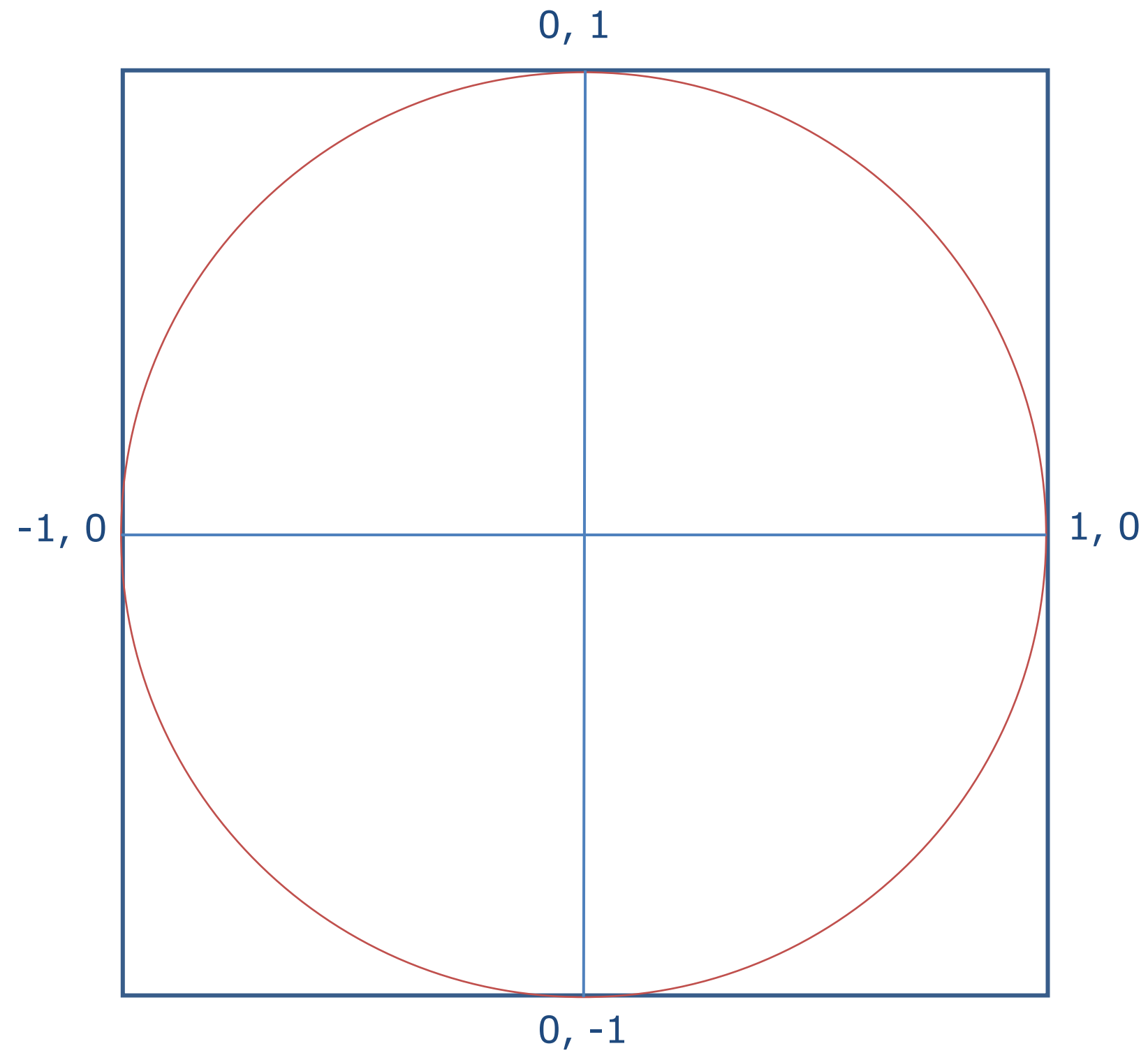


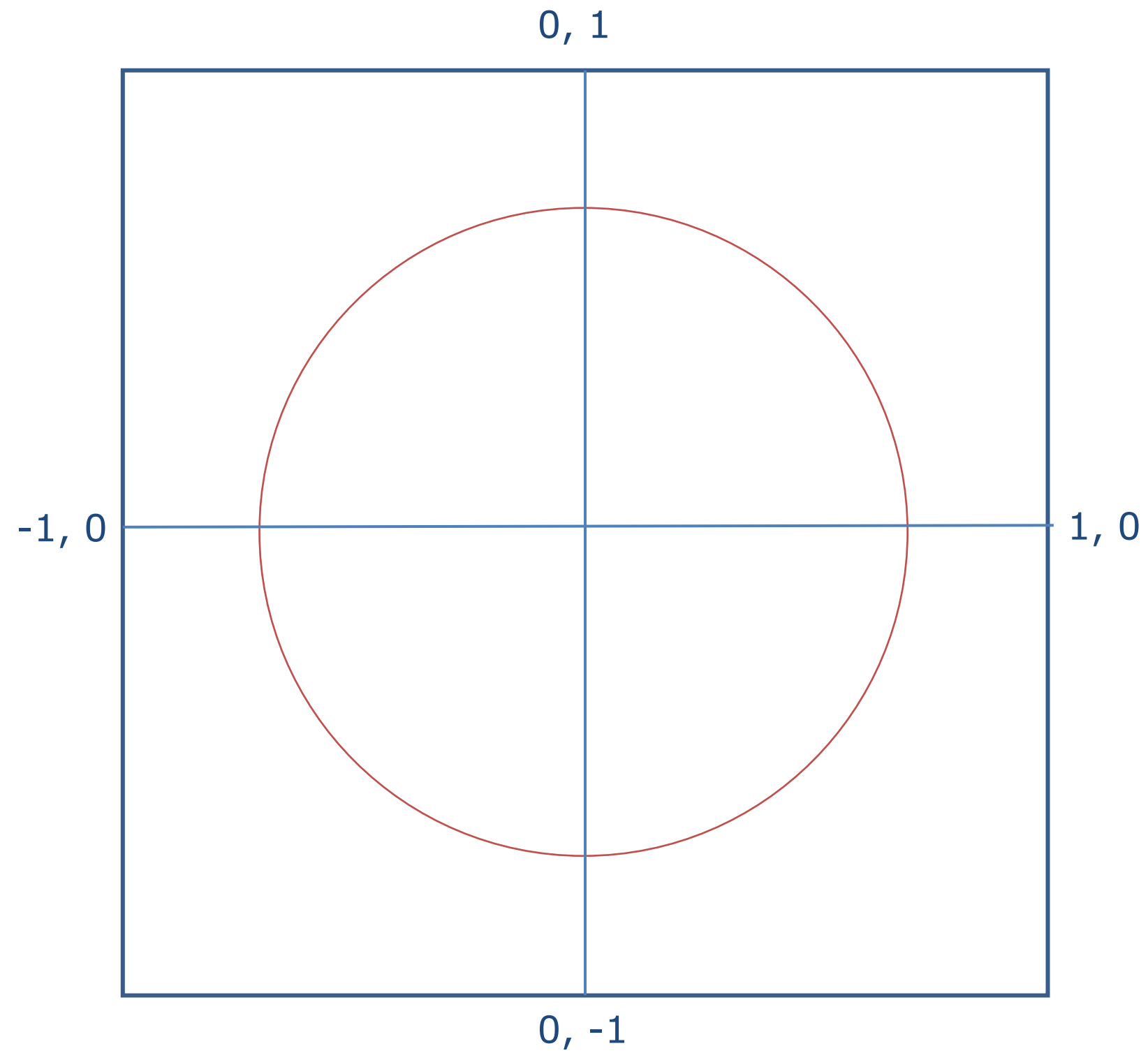
# Debug Display

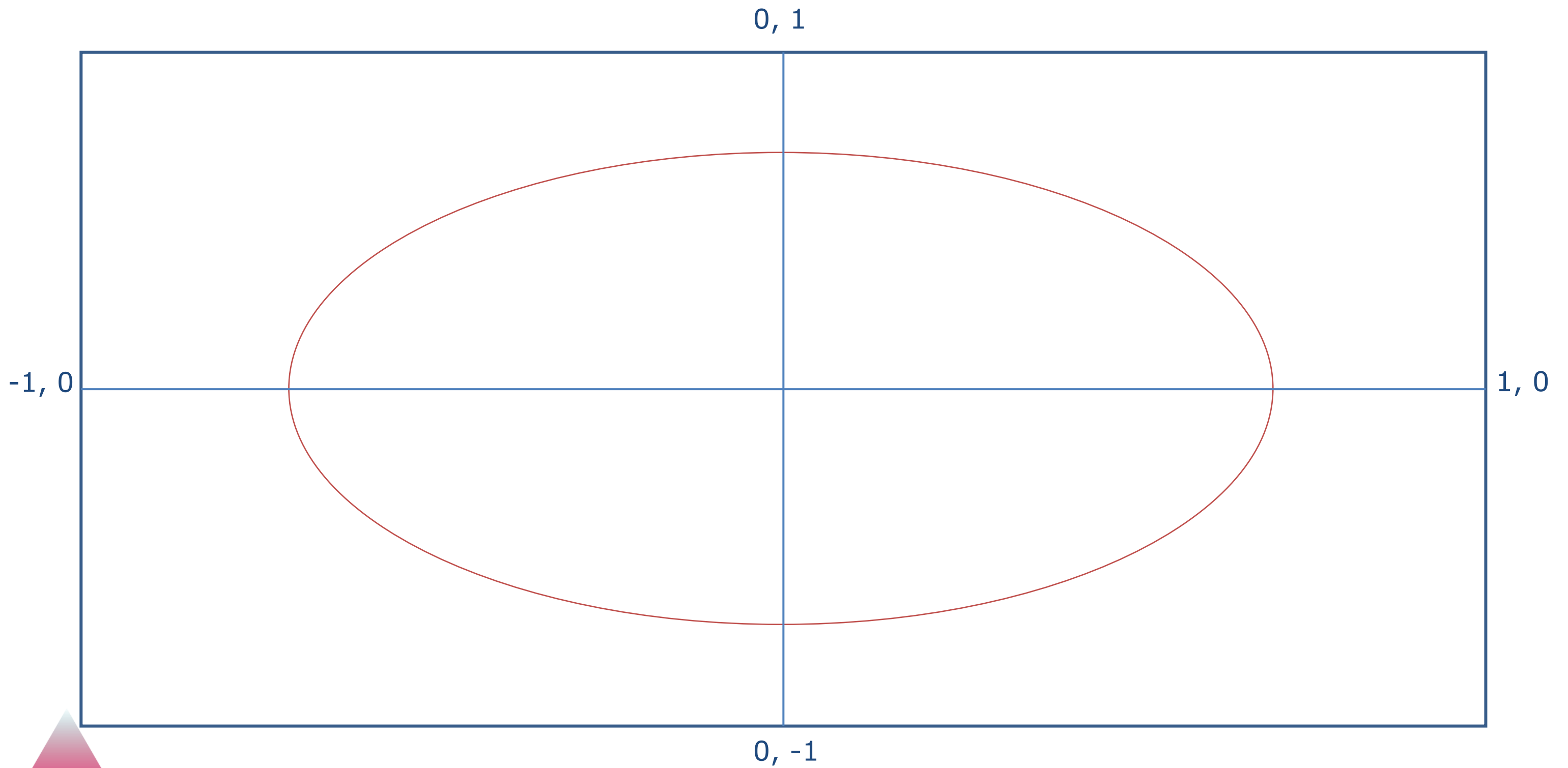
- Traditionally, we display a sphere of the appropriate radius.
- Not anymore: distance attenuation isn't shaped like a sphere (or a circle) anymore
- How do we help the sound designers understand where in world space the sound will be audible?
- We need to invert the process of projection

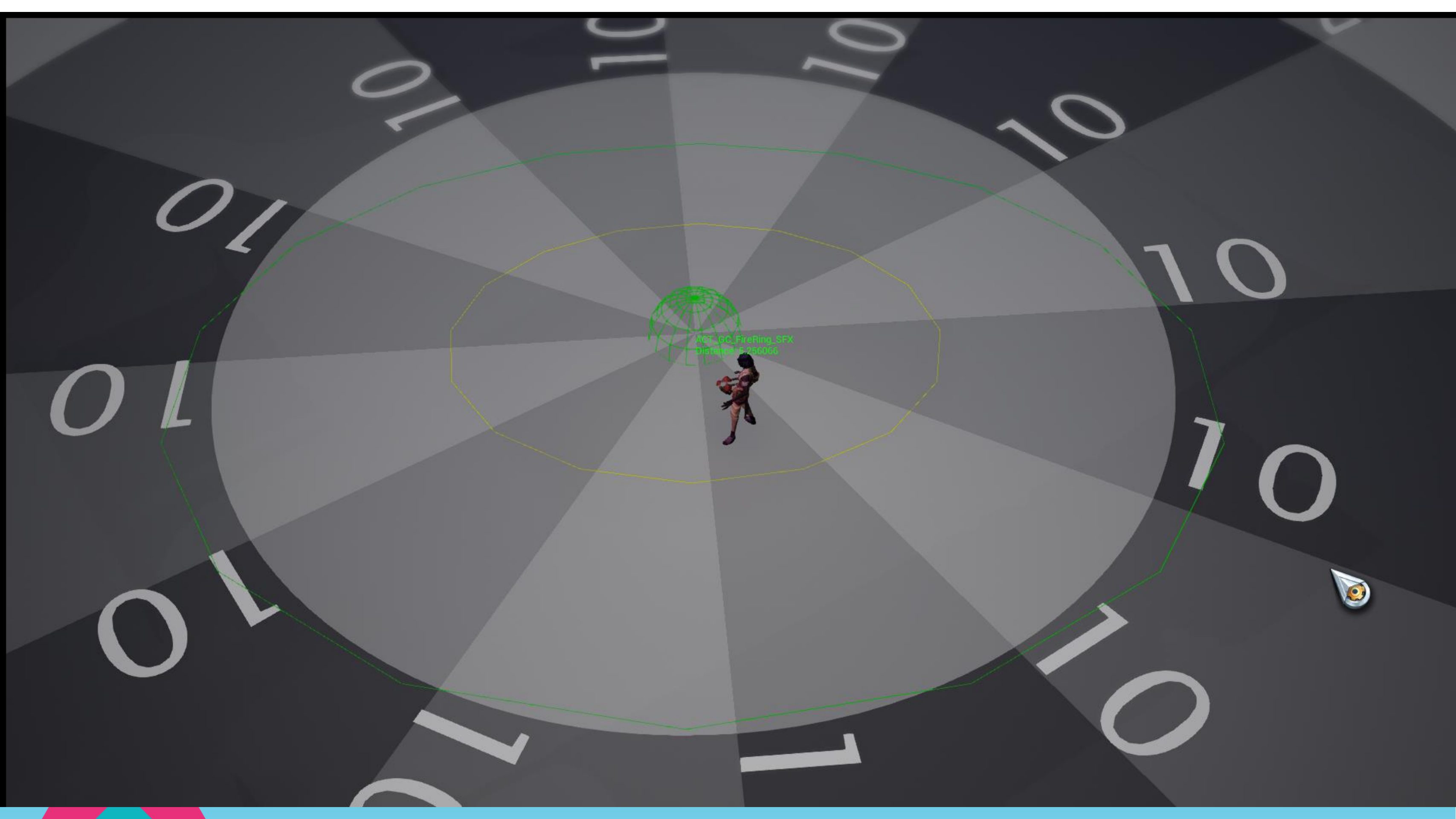
# Algorithm:

- Take the vector ( $S_x$ ,  $S_y$ ,  $A_z$ ) and project onto screen space
- Make a circle in screen-space at a radius of Distance/20 (or whatever your screen-space scale is), centered on the projected point
  - Convert to pixel coordinates if necessary
    - In Torchlight 3, we must first convert from -1..+1 to 0..1, then convert to pixel coordinates
- Deproject screen to world position + ray
- Intersect ray with the attenuation position's plane
  - Flip intersected X and Y around original sound location



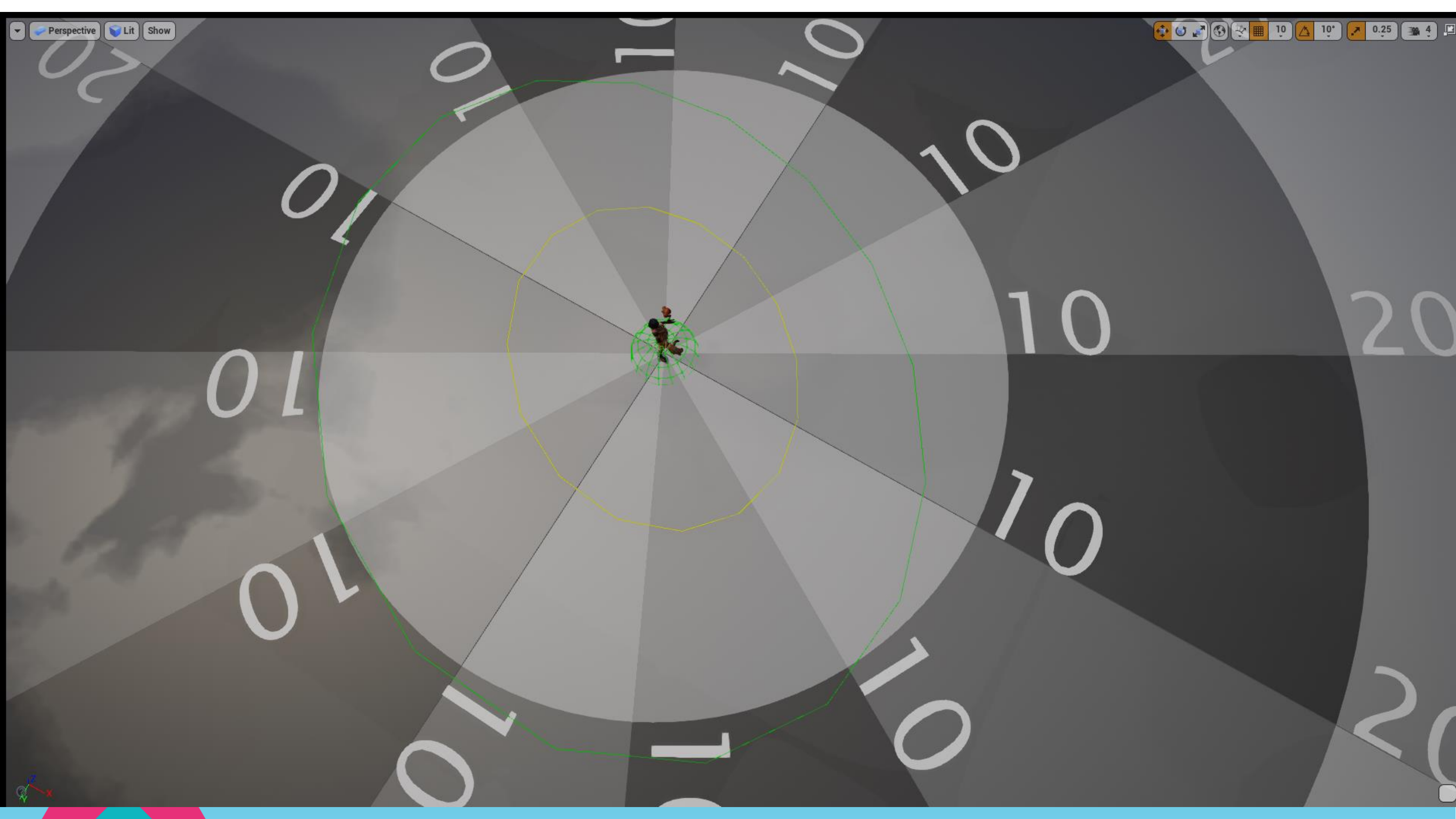


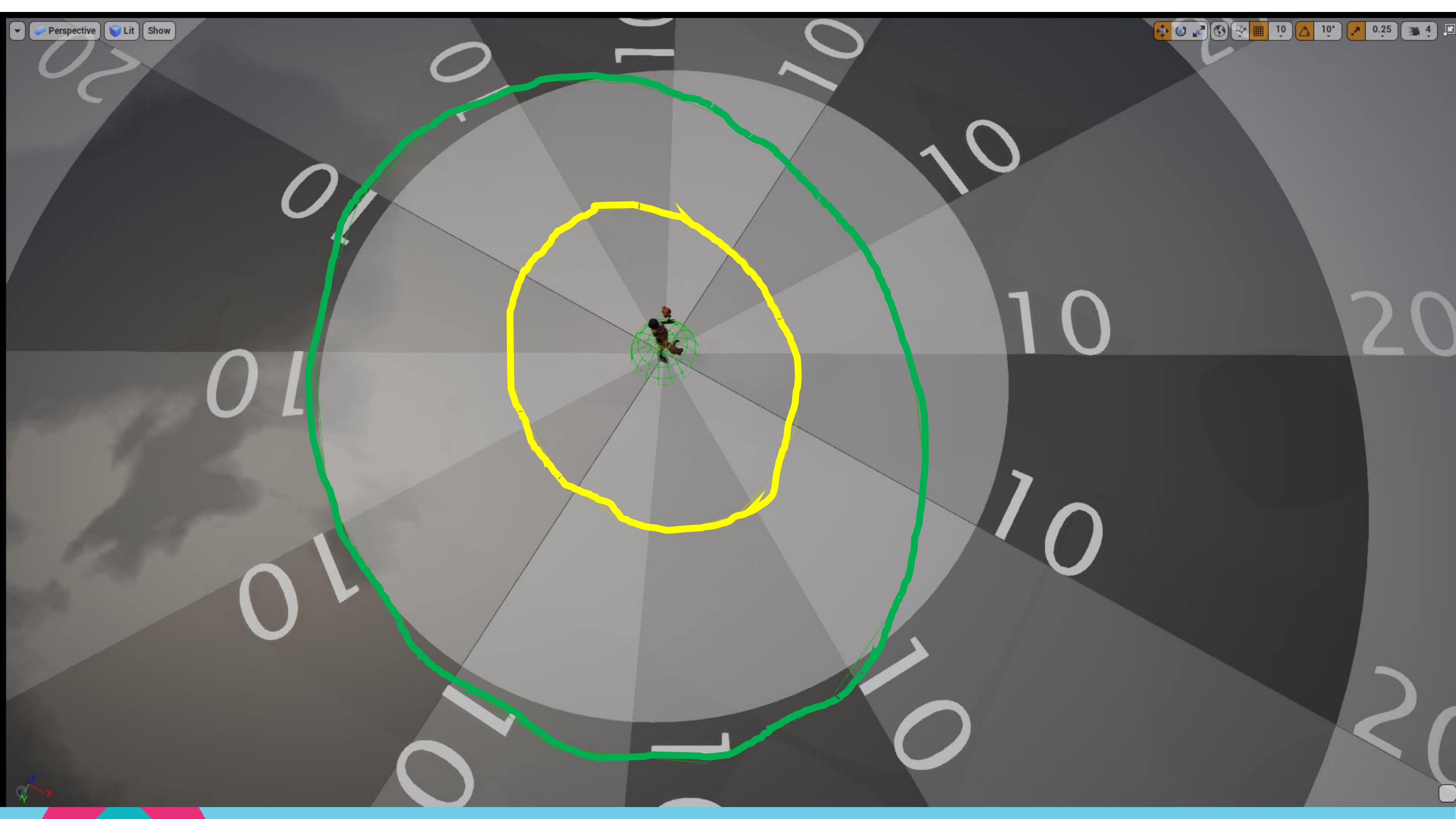






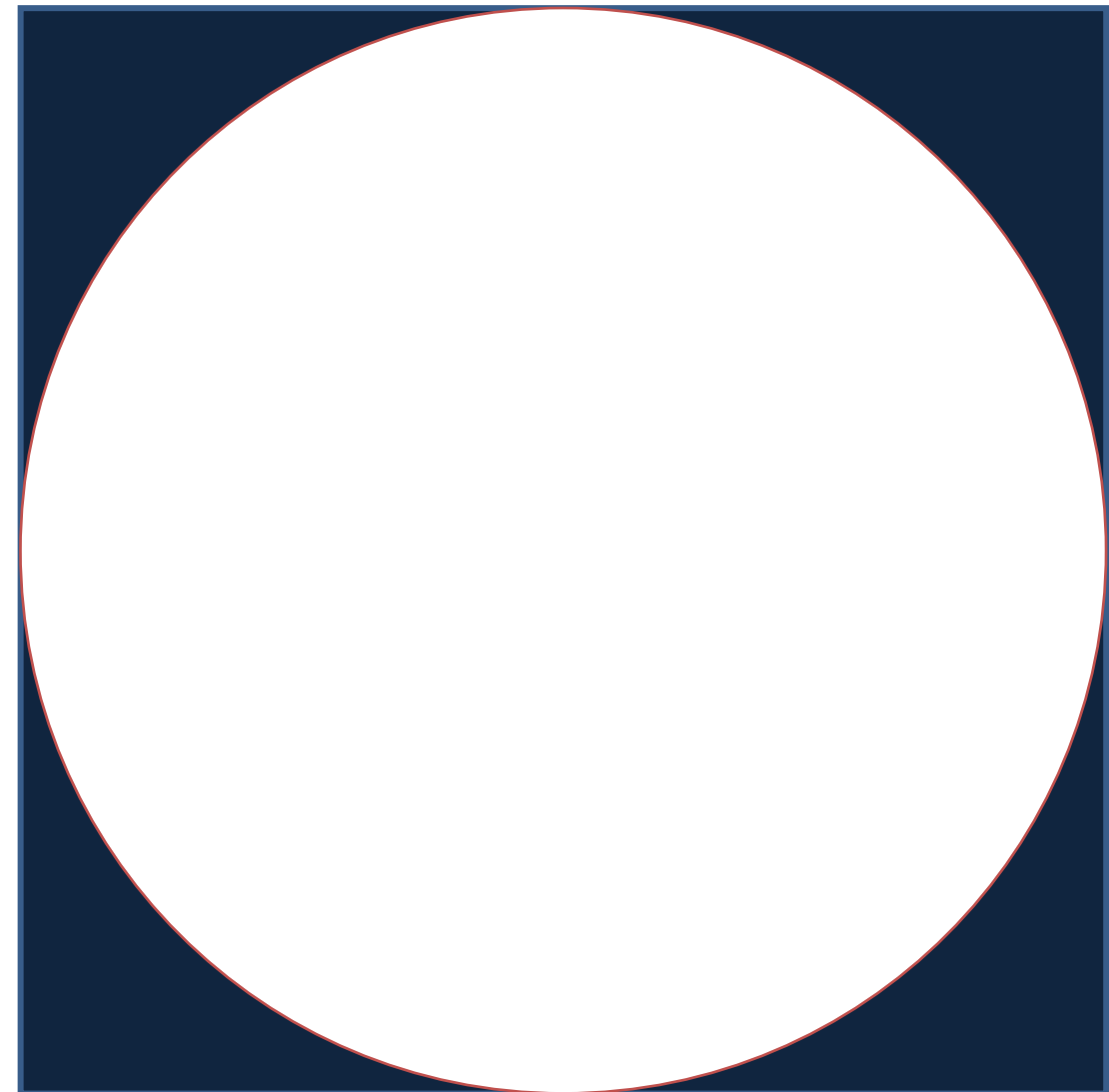


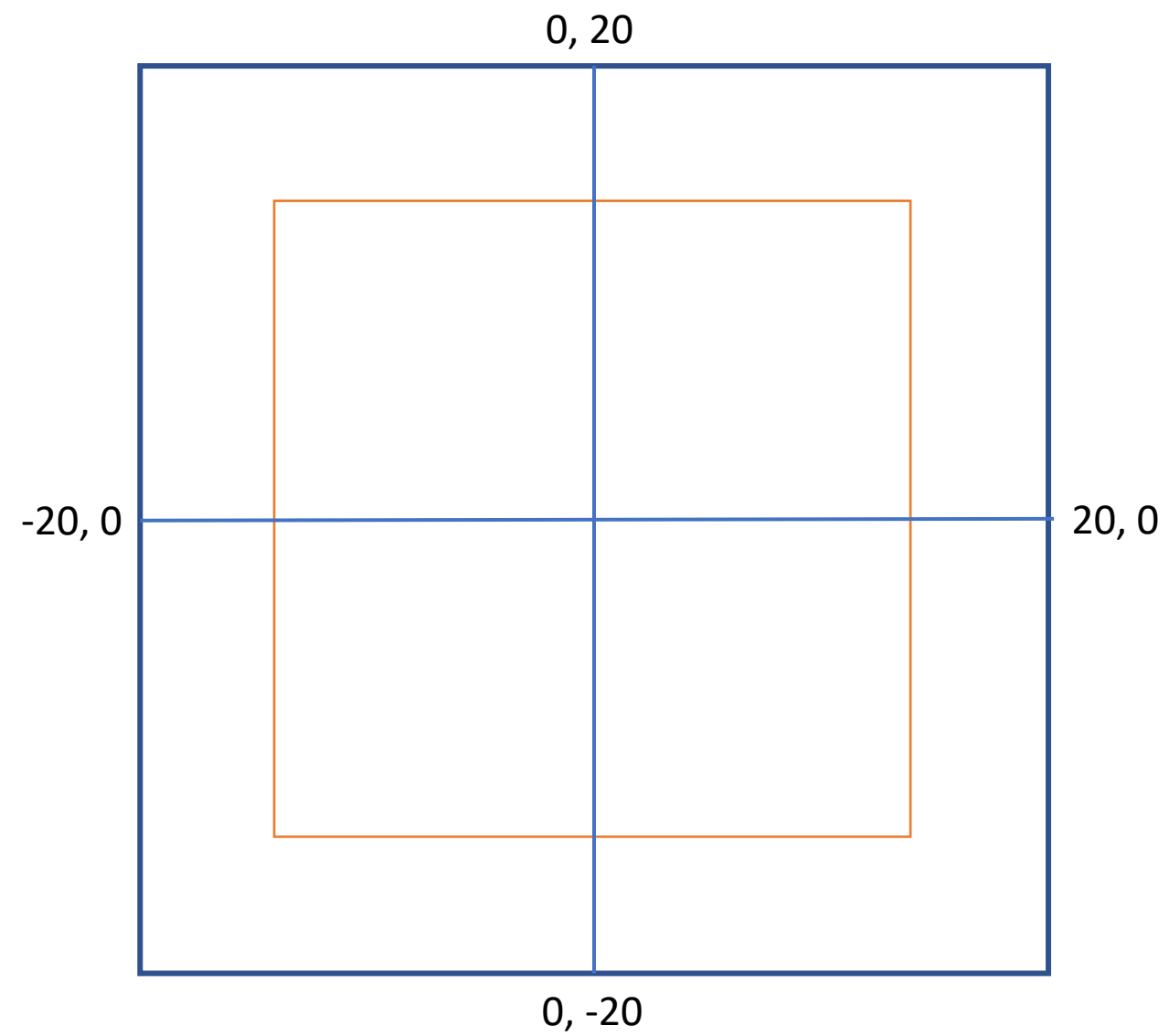


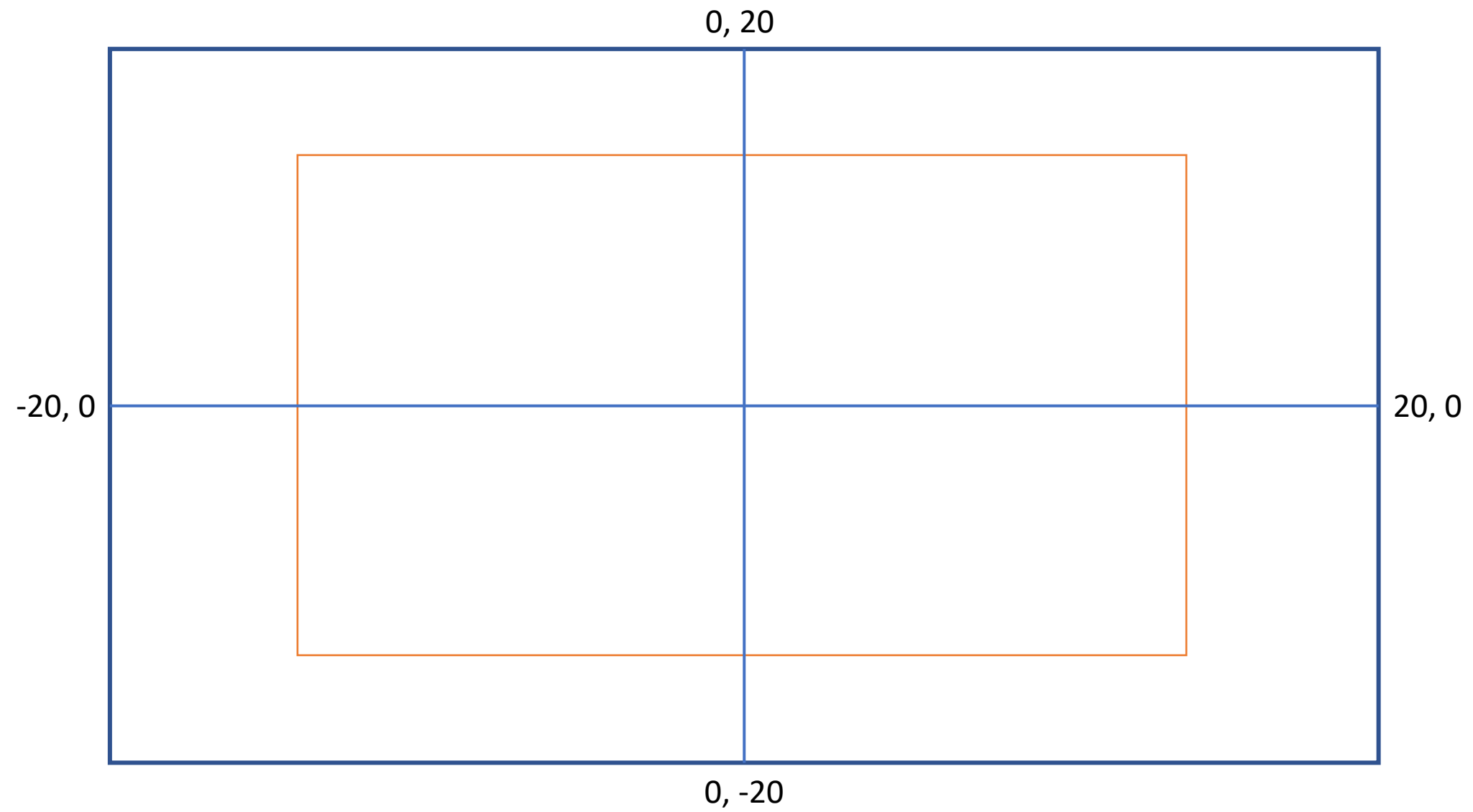


# But wait...

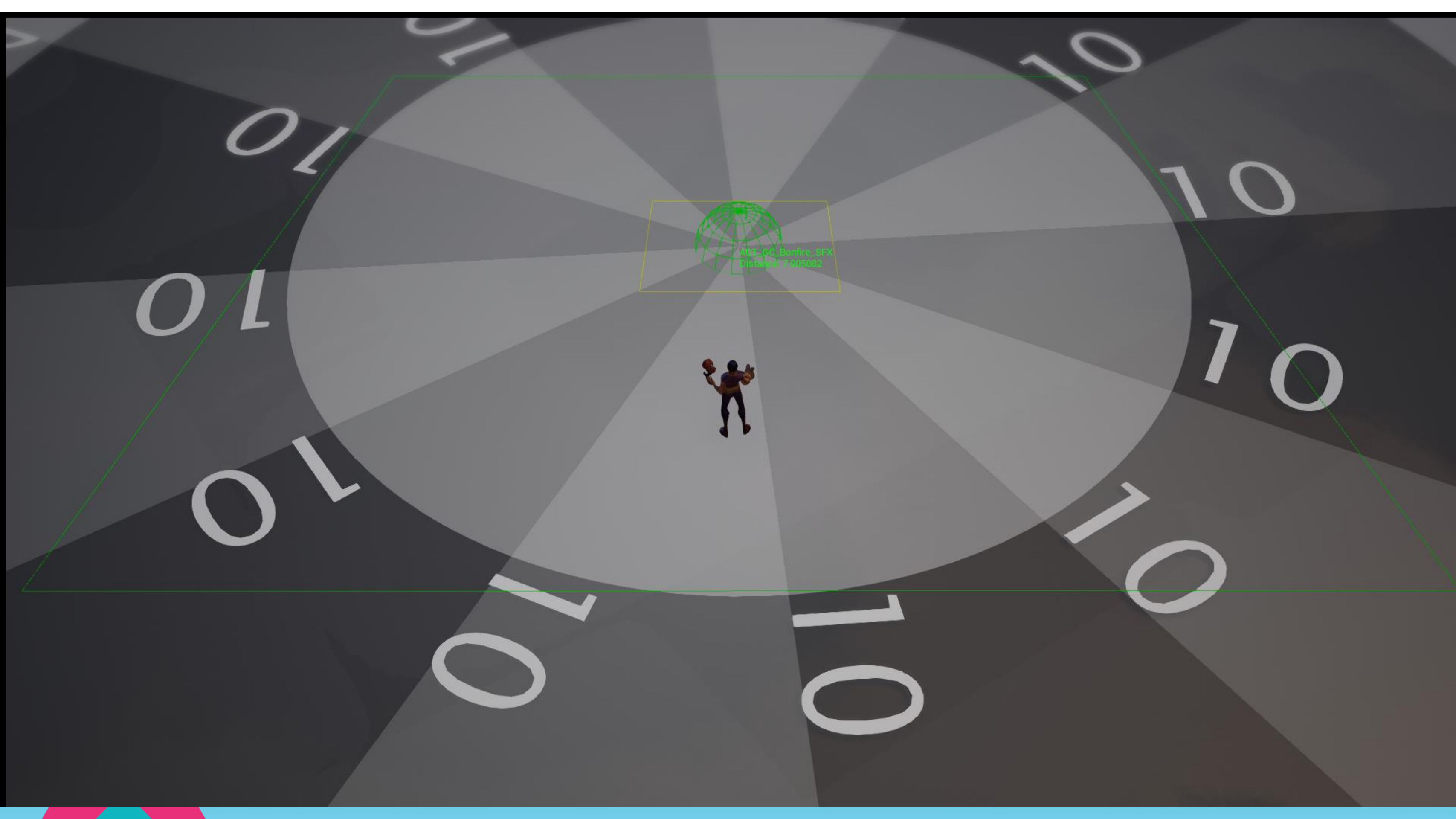
- We've been doing all of this work in screen-space
- What about the corners?

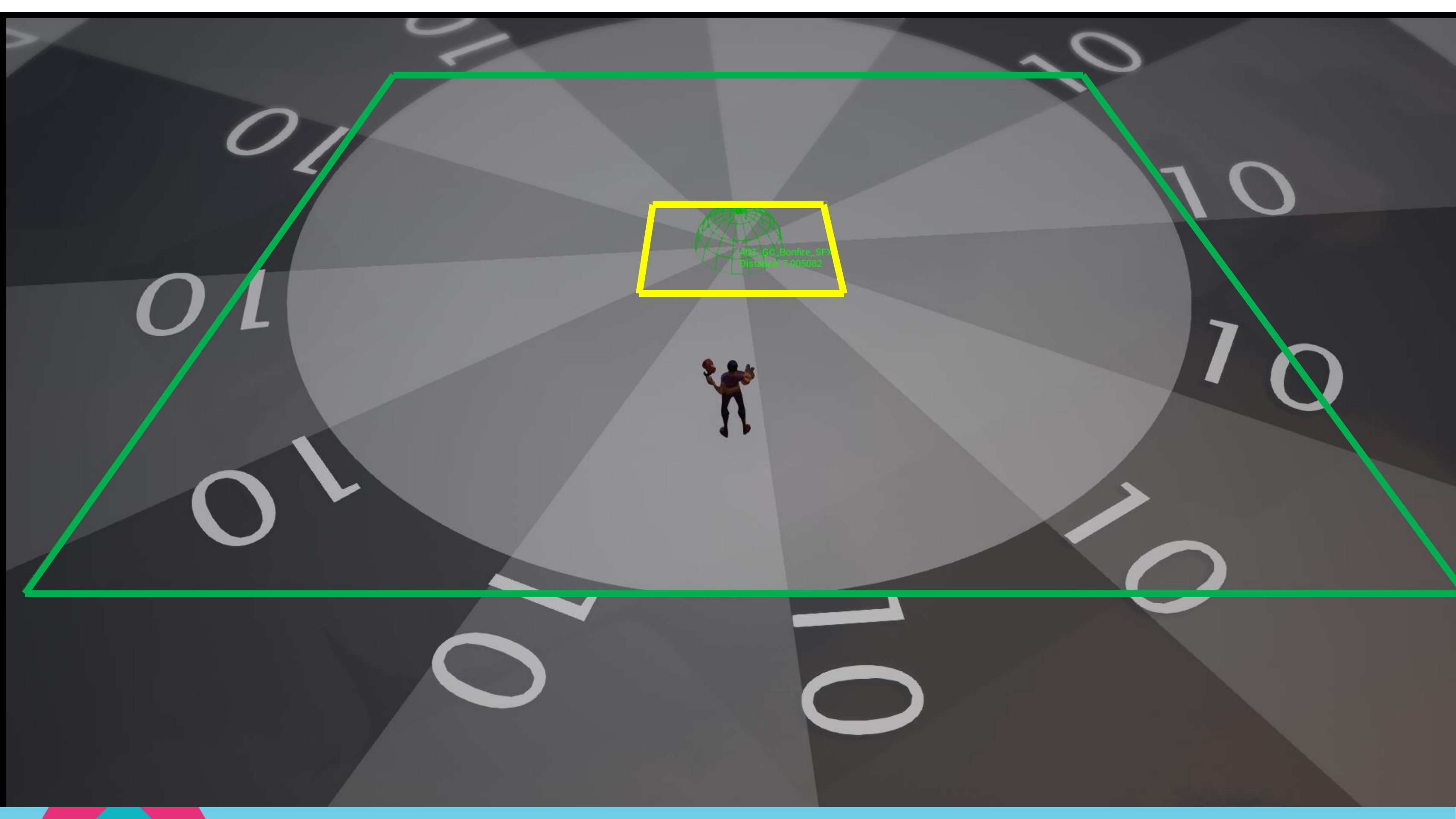




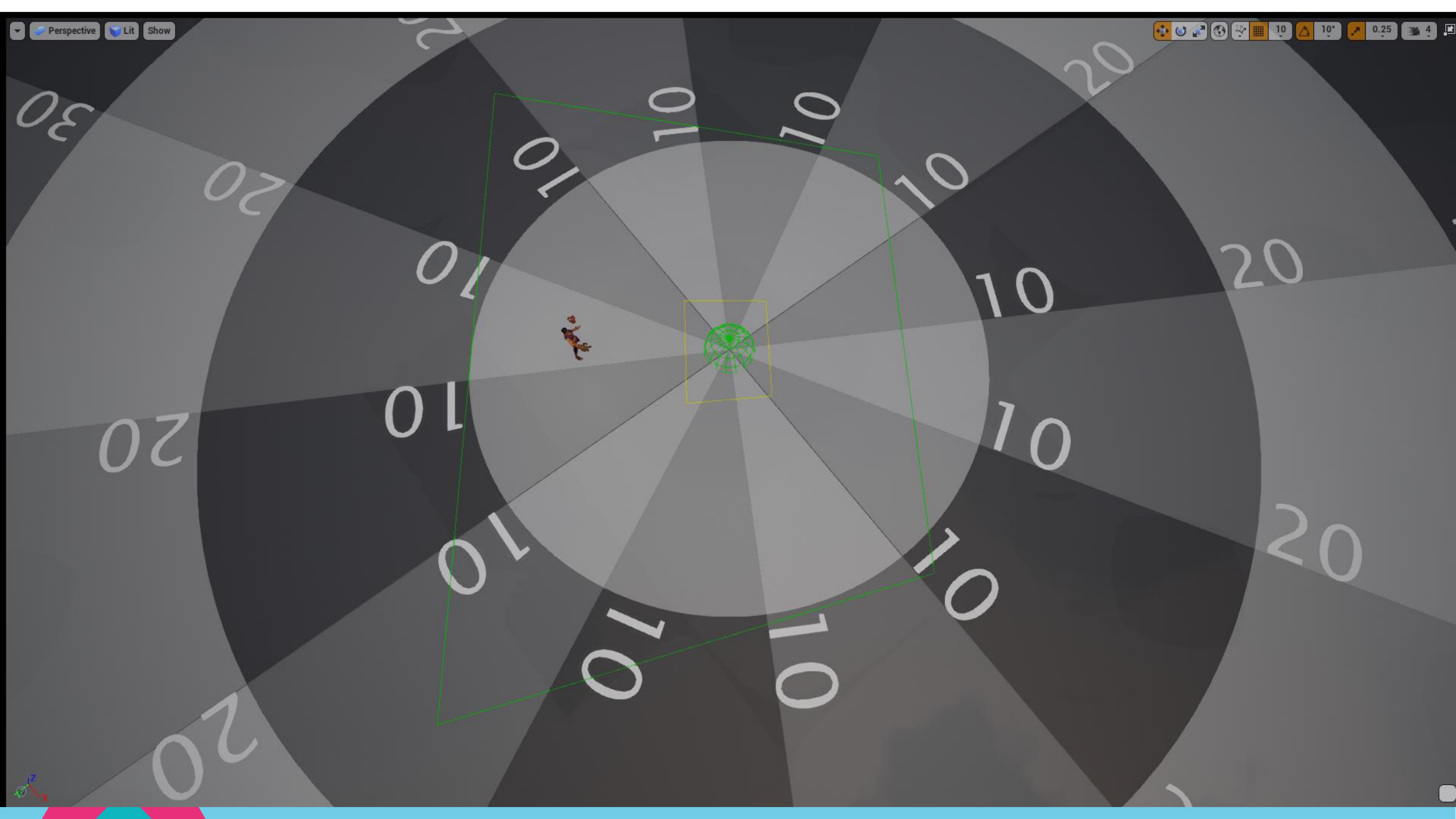


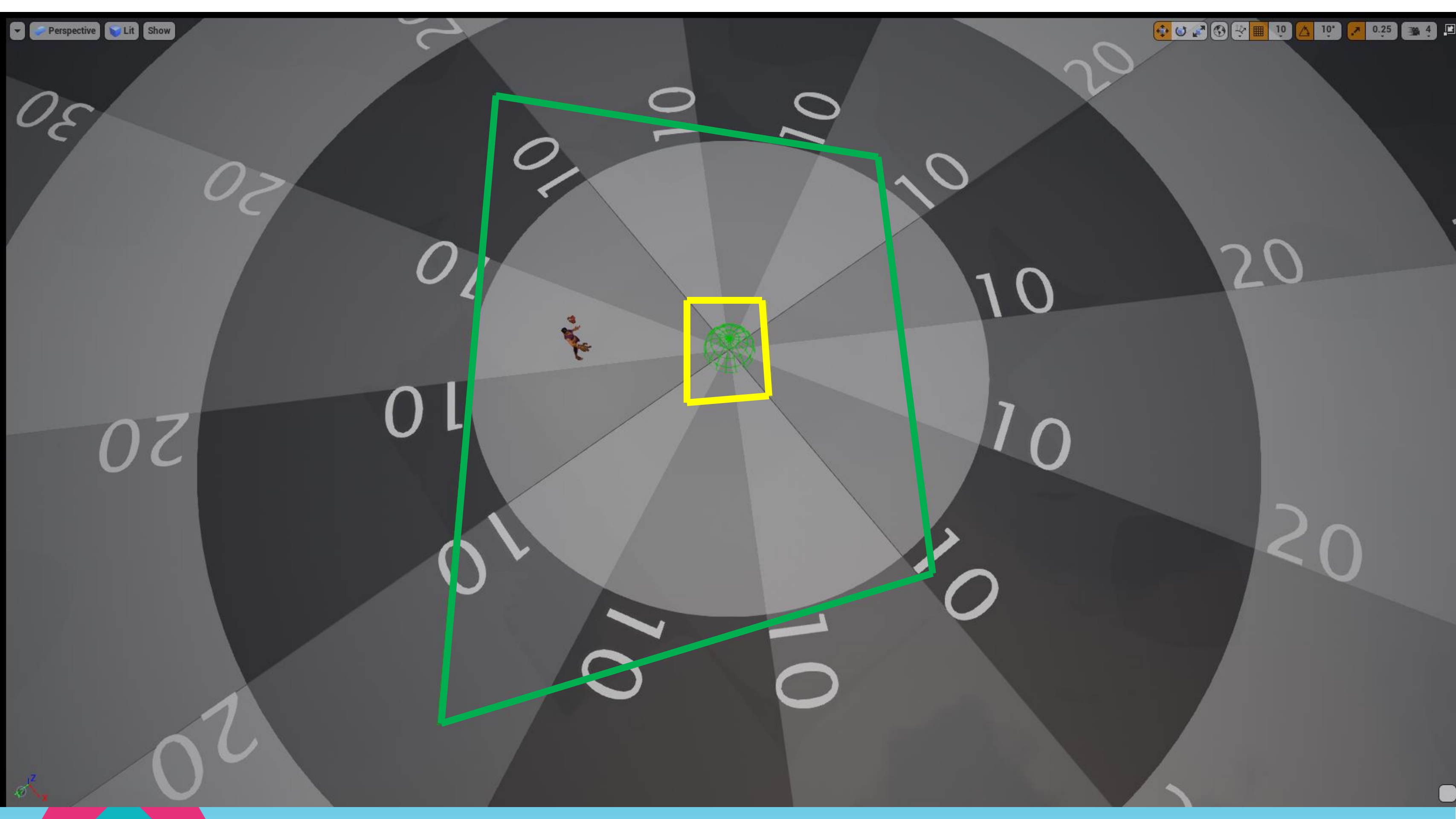






Box: SSC\_Bonfire\_SFV  
Distance: 7.406182





# Content Tendrils

- We have replaced a 3D world-space value with a 2D screen-space value
- Distance no longer means the same thing that it used to
- Every 3D sound event must be reauthored to account for this new idea of distance
  - Every distance calculation must make a decision whether to use 3D or 2D distance



# Screen-Space Distance Attenuation in Action

- Live Demo



# Conclusion

- ARPGs have distinctive (but not unique) challenges
- Importance-Based Mixing is a fundamental feature
  - Part of every audio engine from day 1
- Volumetric Sounds solve the river problem elegantly
  - You don't necessarily have to understand the math
- Screen-space distance attenuation is also a fundamental feature
  - (If you have a fixed-camera game like an ARPG or RTS)

# Conclusion

- Complex features are complex
- Debugging and visualization routines are *critical*
- Good visualization can be hard, but it's always worthwhile

# Questions

- Comments
- Compliments
- Complaints
- Queries
- Inquiries
- Inquests
- Observations
- Opinions
- Remarks
- Commendations
- Objections
- Impressions
- Thoughts
- Commentary
- Assertions
  - (Just kidding! Everybody knows that game devs ignore asserts.)