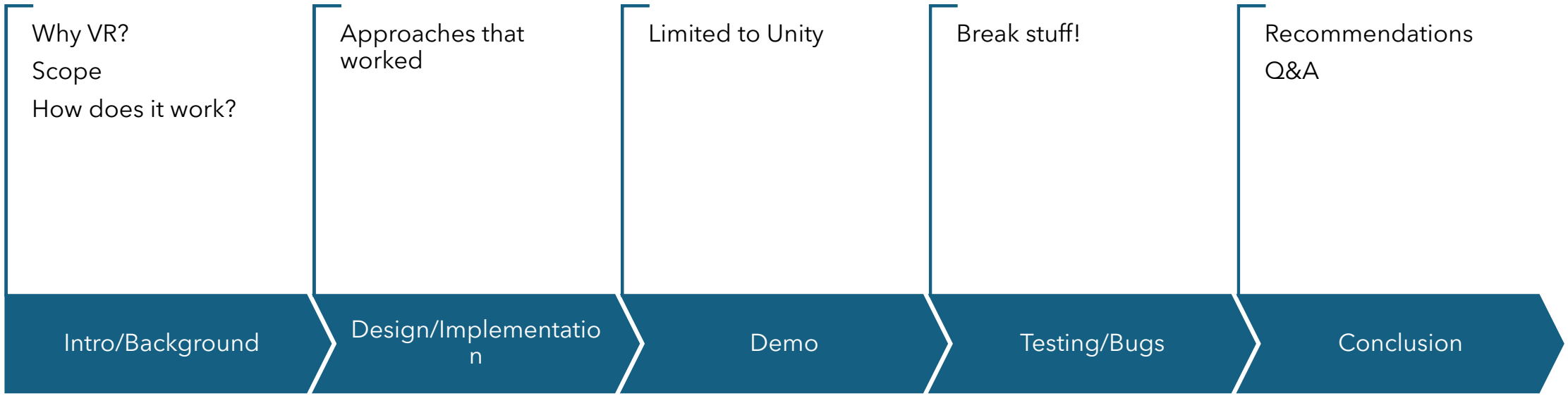
Museum of Network Models

COMP 2100 03B
Anthony Ferrucci
Robert Yildirim
Raghav Vaid

November 17, 2024

MONM: roadmap



Problem

- Networking is "invisible"
- Relies on analogies
- Concepts too abstract
- Difficult for students to understand

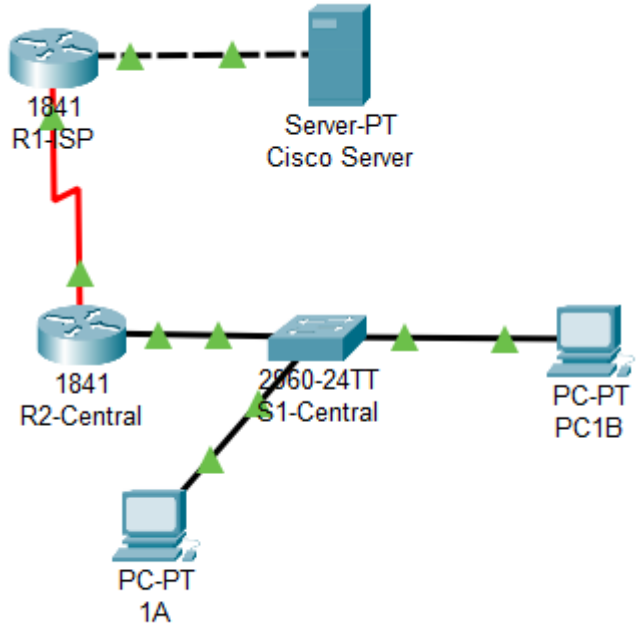
ex. Wireshark capturing packets out of thin air

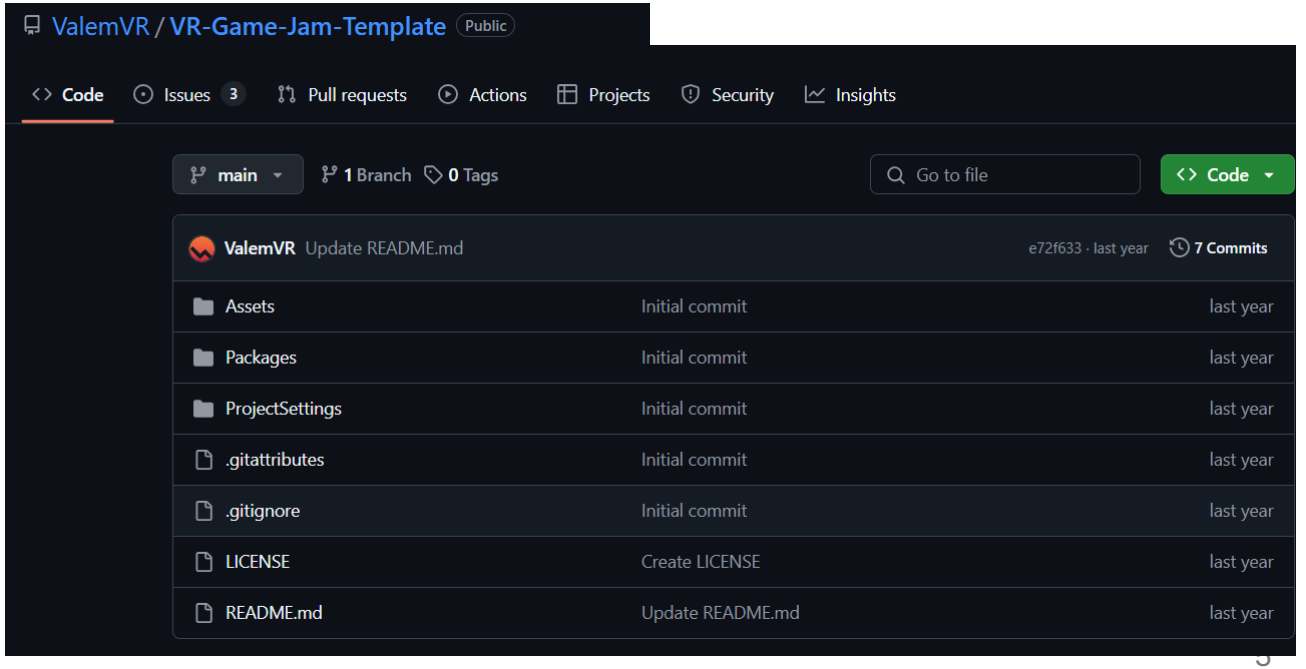
Objectives

- Visualize networking
- Bring analogies to life
- Learn about VR
  - Used in more industries
- Scope changed with problems we encountered

Background and Related Work

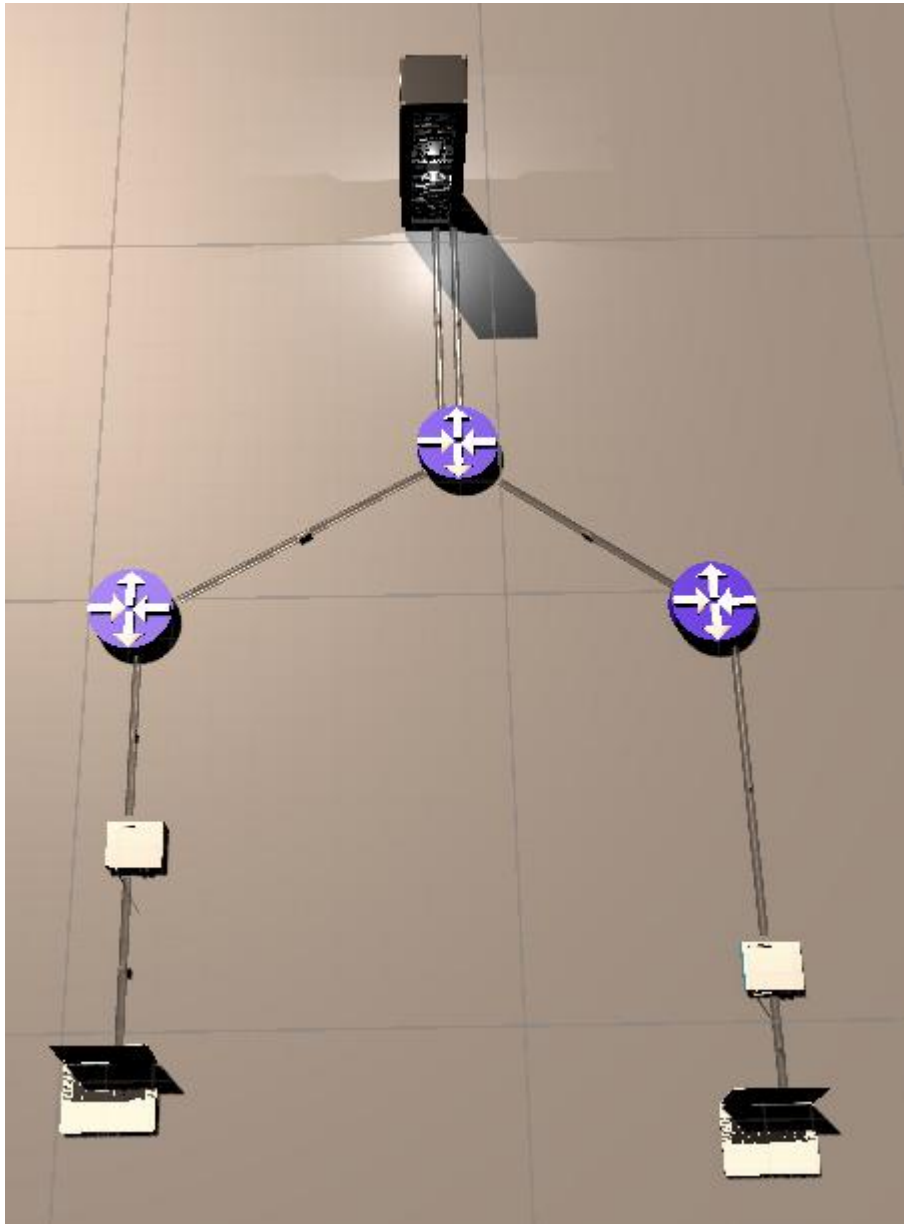
- VR projects unrelated to networking
- CISCO is 2D





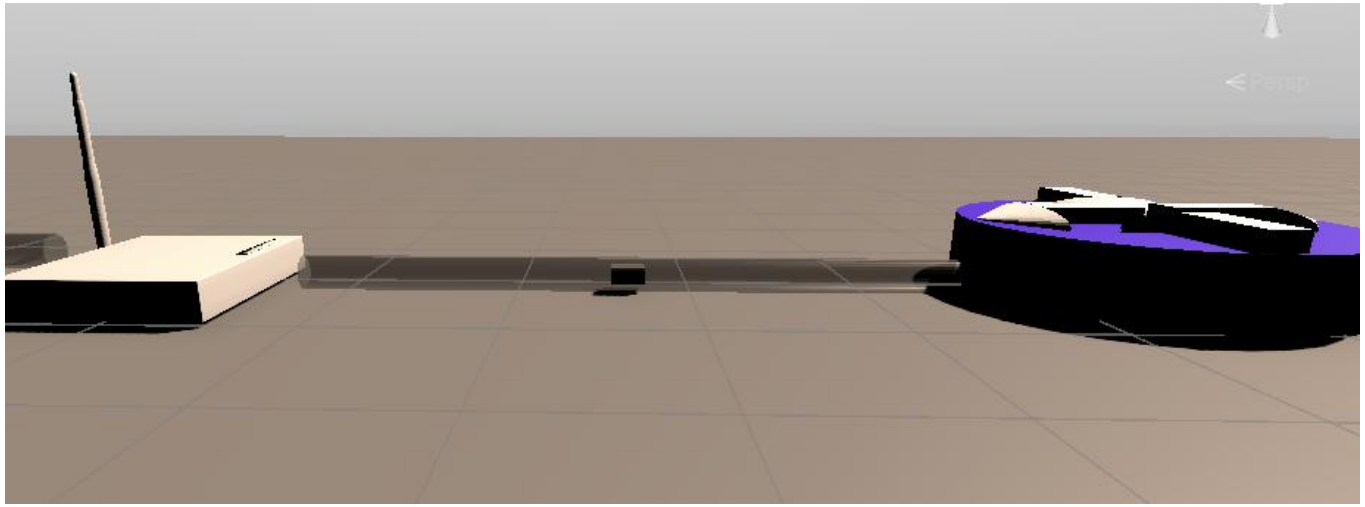
Architecture Used

- Client-Server architecture
- Low-throughput
  - Lag
  - Concise movement



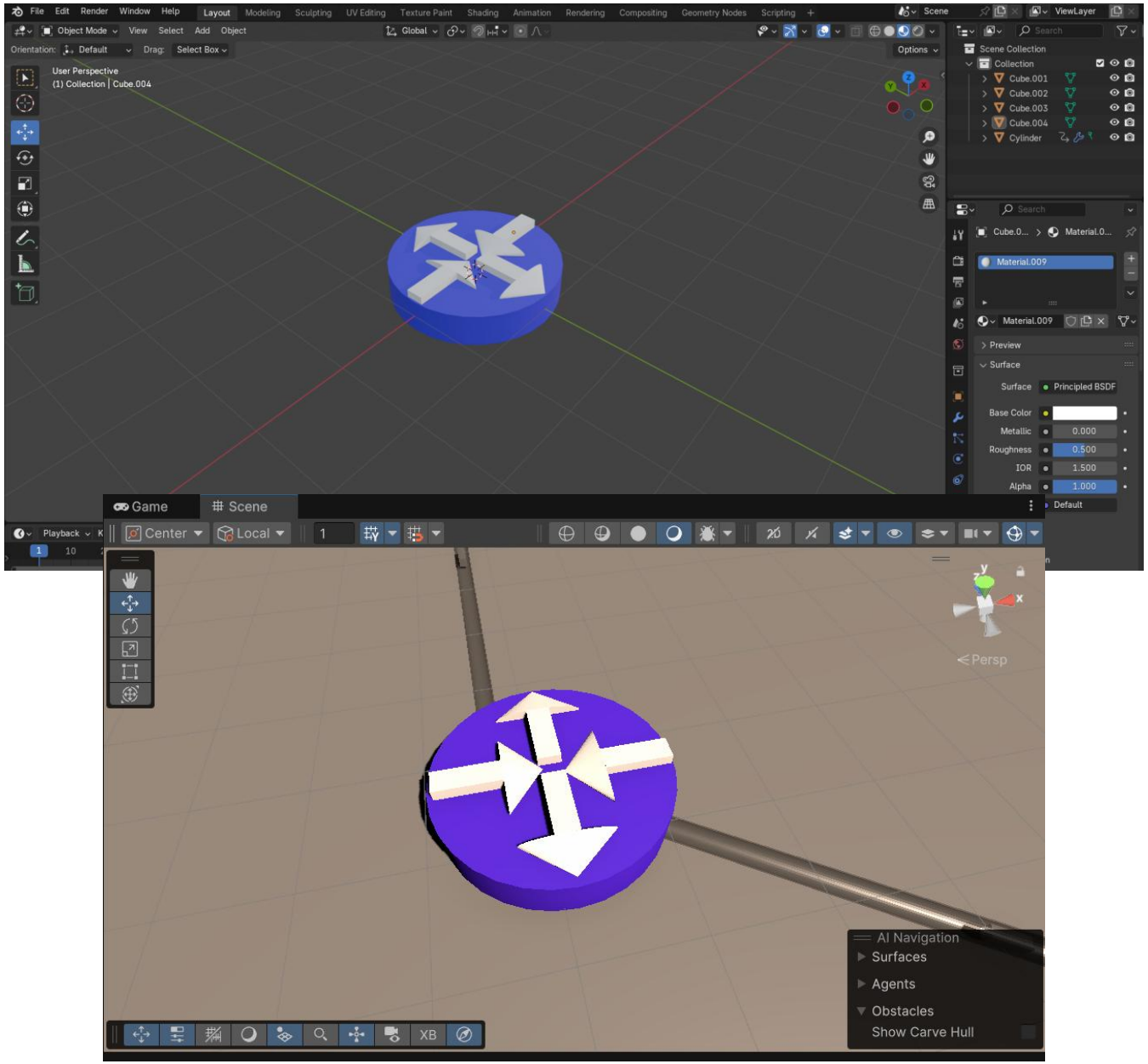
Key Features

- Moving cubes = packets
- Objects use prefabs
  - Build just like in CISCO

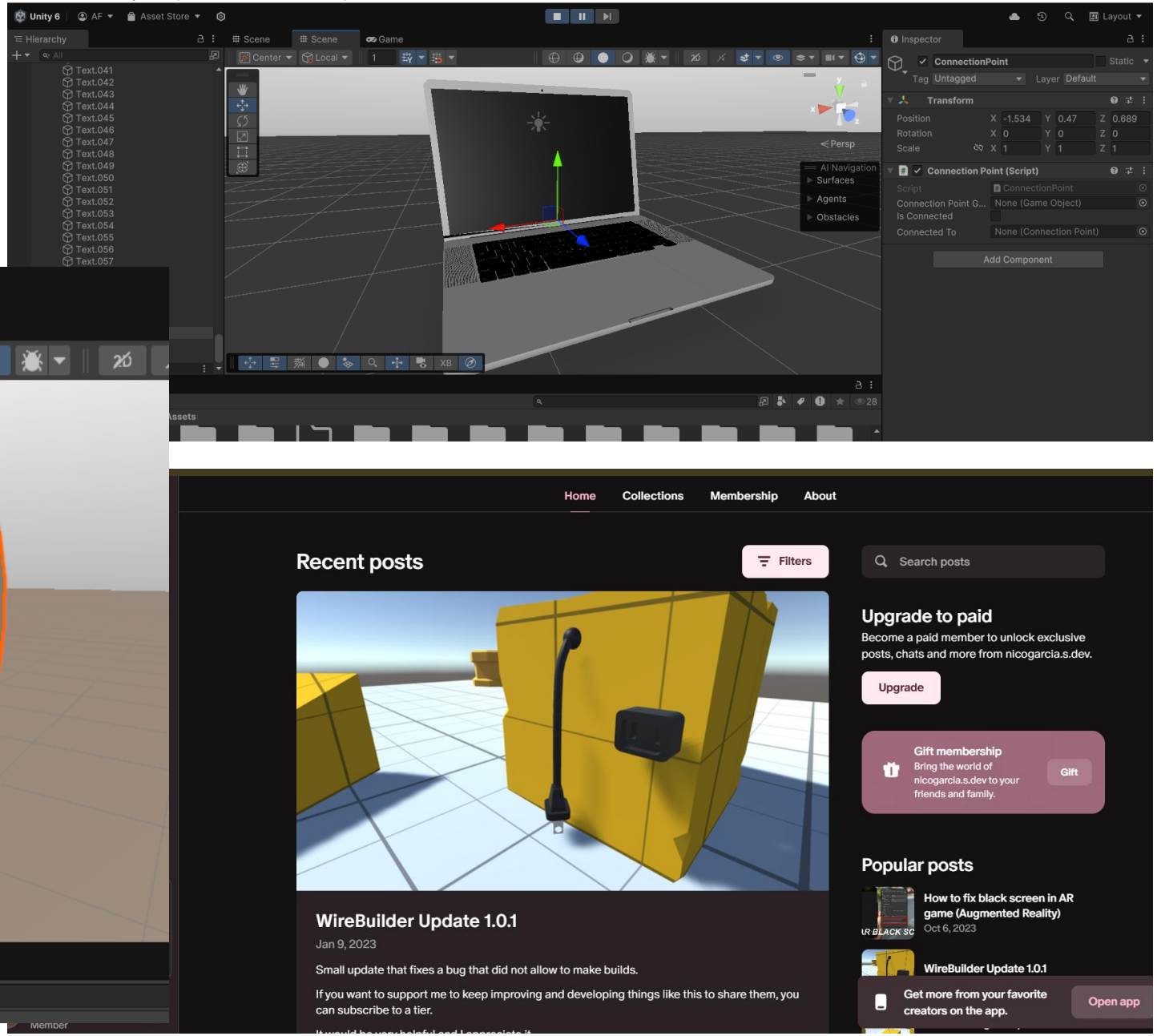


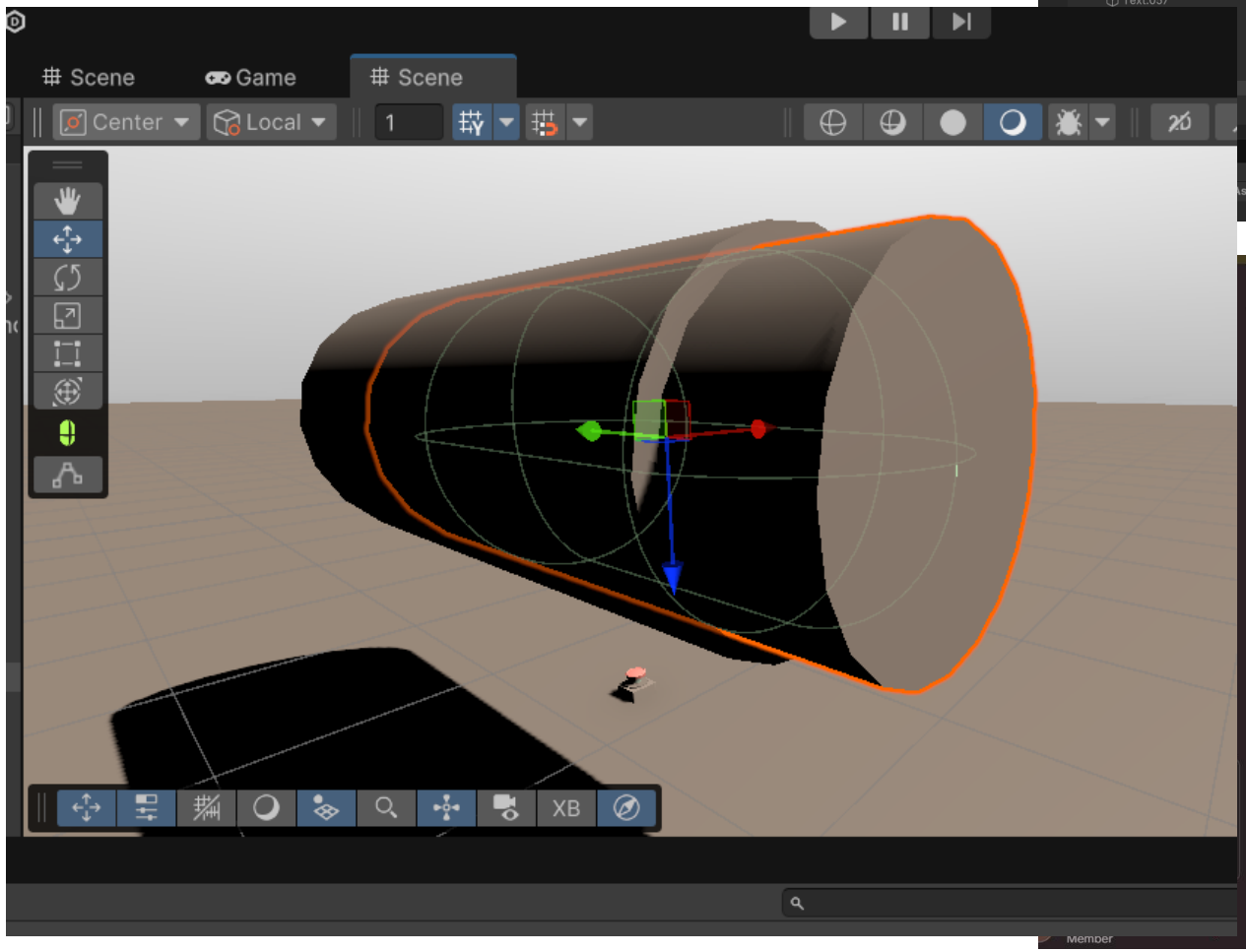
Technologies Used

- Independent platforms were built with **interoperability** in mind
- Blender, Unity, Meta Quest, Android Opensource SDK all support the same file types
- Large design community



Modeling


Unity 6
Hierarchy
Scene
Game
Inspector
ConnectionPoint
Transform
Position
Rotation
Scale
Connection Point (Script)
Add Component
AI Navigation
Surfaces
Agents
Obstacles
Persp


Scene
Game
Scene
Center
Local
XB

Home
Collections
Membership
About
Recent posts
Filters
Search posts
Upgrade to paid
Become a paid member to unlock exclusive posts, chats and more from nicogarcia.s.dev.
Upgrade
Gift membership
Bring the world of nicogarcia.s.dev to your friends and family.
Gift
Popular posts
How to fix black screen in AR game (Augmented Reality)
Oct 6, 2023
WireBuilder Update 1.0.1
Jan 9, 2023
Small update that fixes a bug that did not allow to make builds.
If you want to support me to keep improving and developing things like this to share them, you can subscribe to a tier.
Get more from your favorite creators on the app.
Open app

Scripts (ChatGPT!)

- C#
- Allows for animations
- Manual debugging
- OOP
- Robust .NET/XR libraries

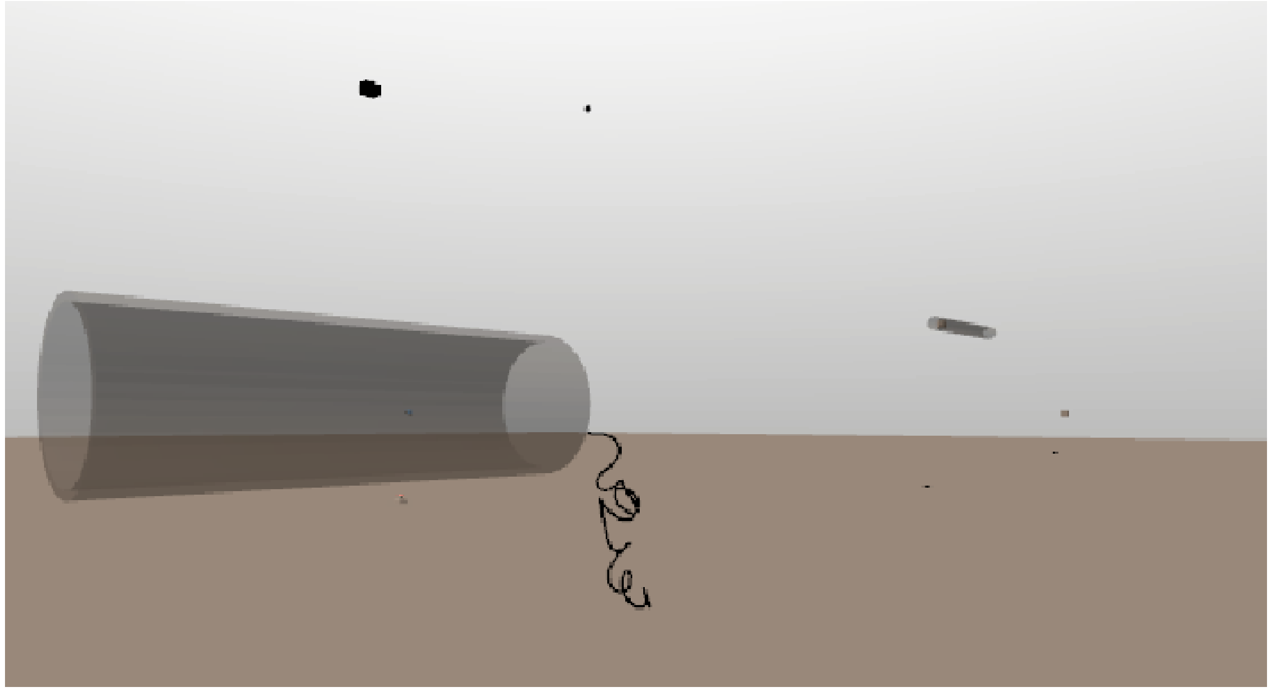
CubeMover.cs U

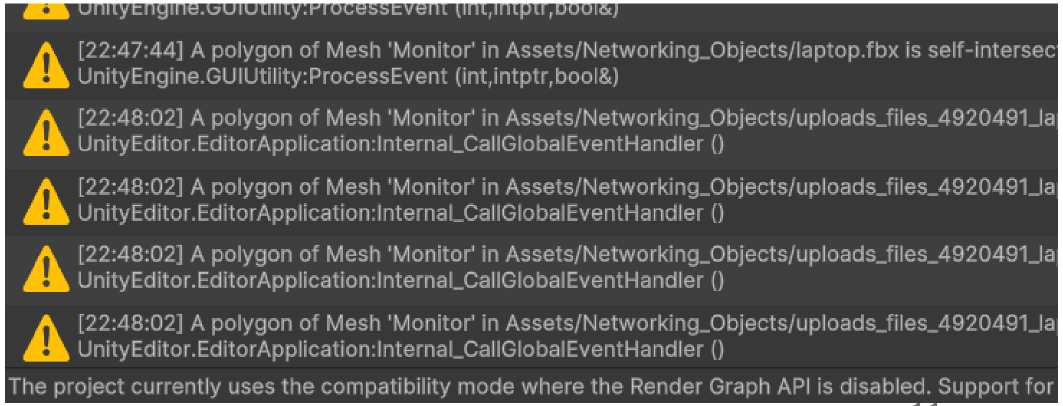
Assets > Networking_Objects > CubeMover.cs > ...

```
3   public class CubeMover : MonoBehaviour
23
        2 references
24      void StartMoving()
25      {
26          // Start the movement and move to the starting position
27          transform.position = startPoint.position;
28          isMoving = true;
29      }
30
        1 reference
31      void MoveCube()
32      {
33          // Move the cube smoothly from startPoint to endPoint using Lerp
34          float step = speed * Time.deltaTime;
35          transform.position = Vector3.Lerp(transform.position, endPoint.position, step);
36
37          // Check if the cube has reached the end point
38          if (Vector3.Distance(transform.position, endPoint.position) < 0.1f)
39          {
40              // Stop the movement and reset position after a short delay
41              isMoving = false;
42              Invoke("ResetPosition", 0.2f);  // Reset the position after 1 second delay
43          }
```

Testing Methodology

- Development area
- Unity Game player
- Lots of breaking stuff
- Failed? Try new approach



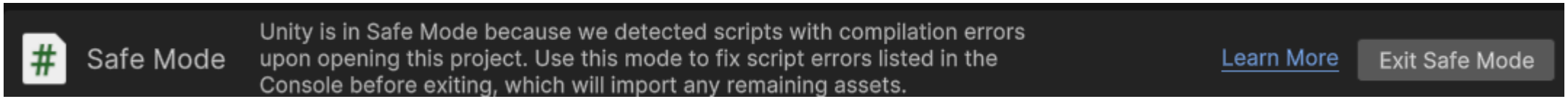


Issue –VR/Unity Not Cooperating

- Unity only worked on one computer for a long time
  - Slowed development
- Project wouldn't open in VR
  - Quest 2 minimum GPU requirement
    - **Intel** processors largely unsupported
    - **NVDIA RTX 2000** ADA Gen laptop GPU

Issue – GitHub Repositories

- Original minigolf template used deprecated code
- Too many references to update
- Had to choose new theme



Conclusion

- Built intuitive way to understand networking
- Modular enough to build own networks
- Recommend using Blender from the start
- Could implement interactive VR elements

Q&A

- Questions?