The webinar will begin shortly
Augmenting reality: Bring digital objects into the real world
The webinar will begin at 8 am PT | 11 am ET | 4 pm BST
Unity

ROAD TO THE METAVERSE

CREATOR SERIES

Unity®

2022

Augmenting reality: Bring digital objects into the real world

Meet your **hosts**

Drew Domokos
Creator & Developer Advocate

Vanesa Mendez
Creator Advocate

Jerome Maurey-Delaunay
Senior Technical Specialist

@jeromemaurey

Ben Radcliffe
Senior Technical Specialist

@lightandalchemy

Agenda

- → **Introduction** (5 minutes)
- → **Augmenting Reality: Bring digital objects into the real world** (45 minutes)
  - What is AR?
  - AR platforms overview
  - Getting started with AR Foundations in Unity!
- → **Q&A** + **Wrap-up** (10 minutes)
- → **Overtime!** (30 minutes)
  - MRTK
  - The Future of AR

Survey available here!

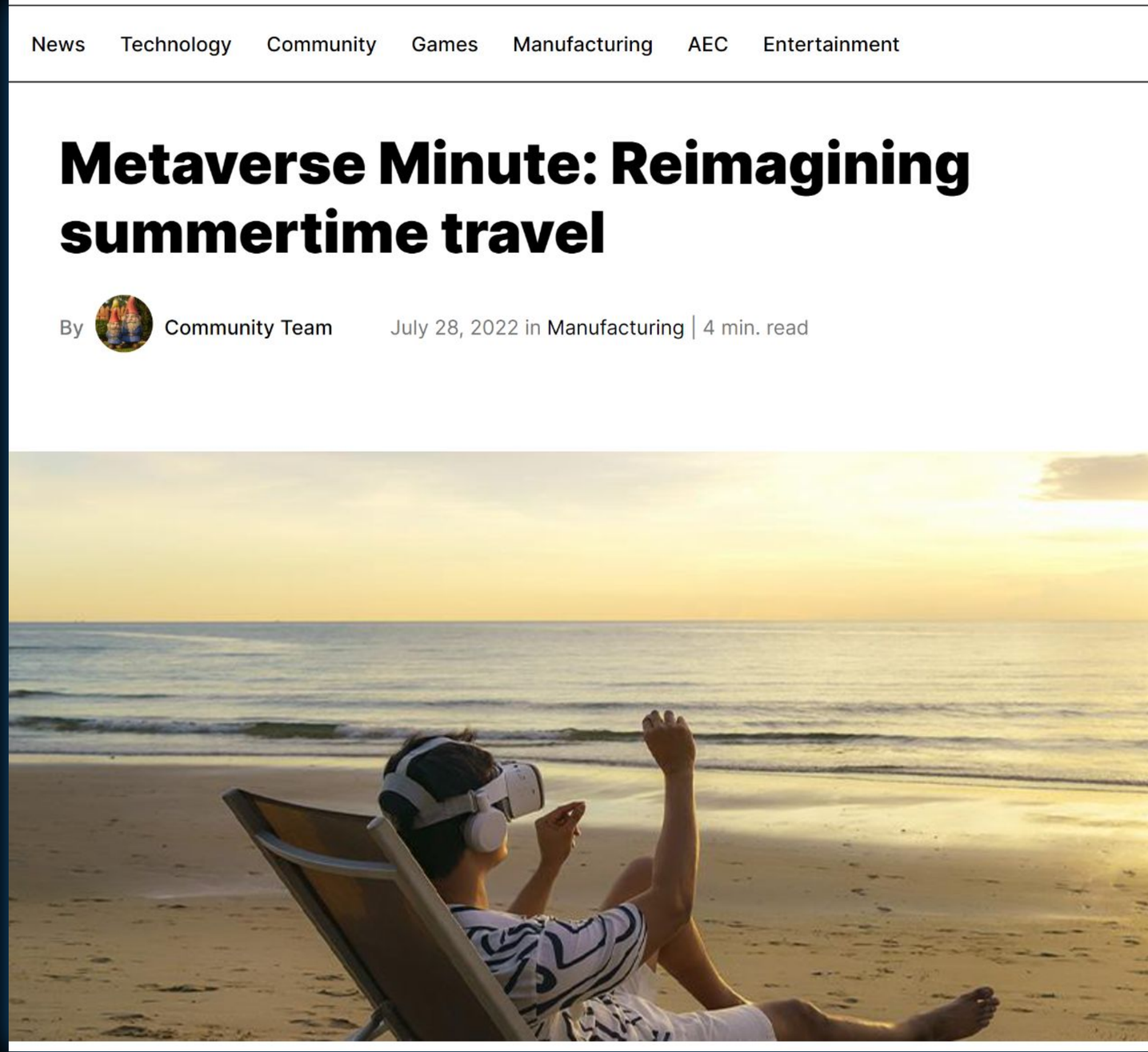
Introduction

Tales from the Metaverse!

Good Times Outside

The Metaverse Minute

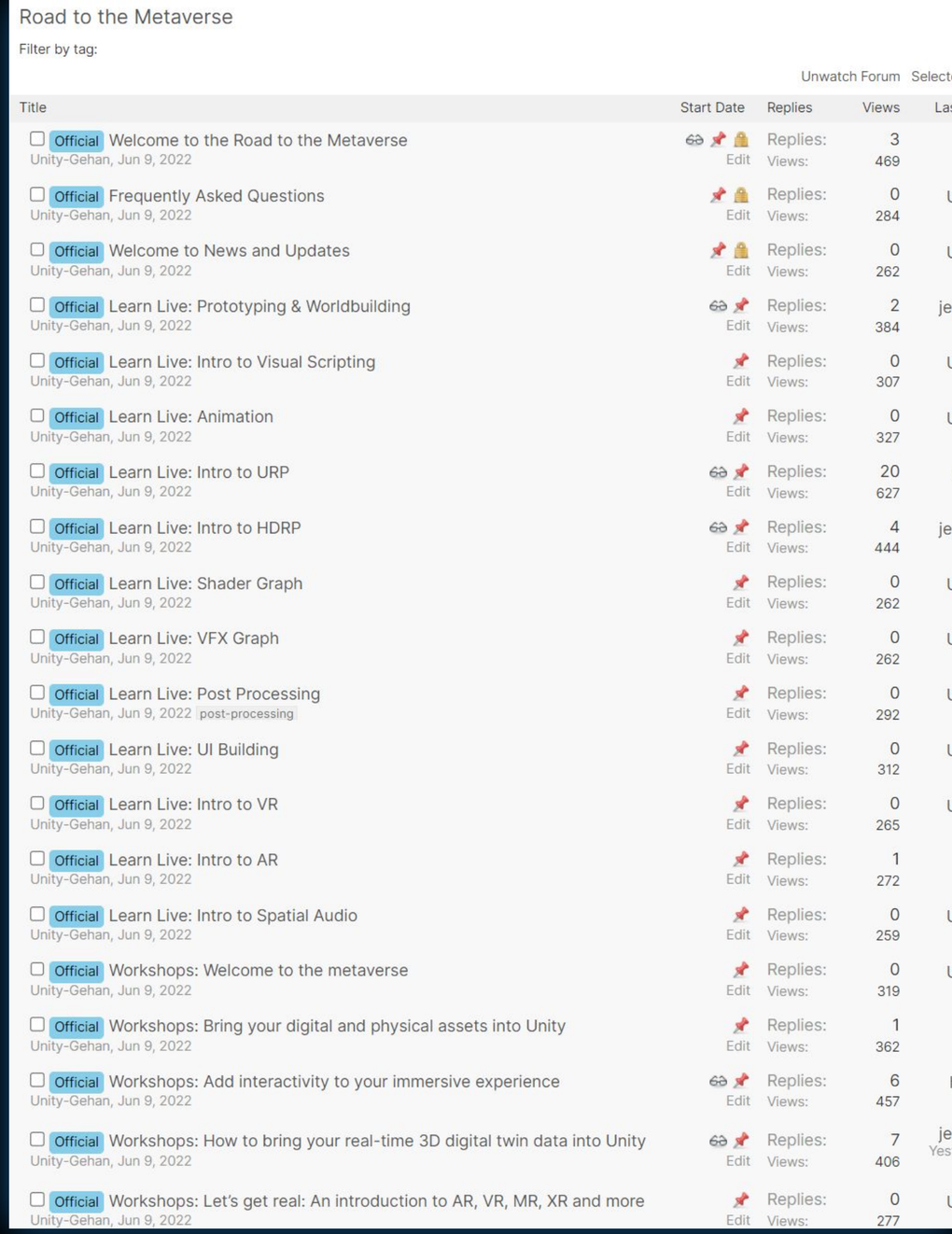
https://blog.unity.com/topic/metaverse-minute



Road to the Metaverse:

Forums!

https://forum.unity.com/forums/road-to-the-metaverse.798/



ROAD TO THE METAVERSE

CREATOR SERIES

Augmenting reality: Bring digital objects into the real world

What is XR?

What is XR? It's shorthand for a related set of new technologies that are changing the way we interact with the world and with each other: Virtual Reality, Augmented Reality, and Mixed Reality.

- **VR** - Virtual Reality
- **MR** - Mixed Reality
- **AR** - Augmented Reality
- **XR** - Umbrella term for all the above

Unity Learn Intro to XR

What is AR?

Let’s jump in!

What is AR?

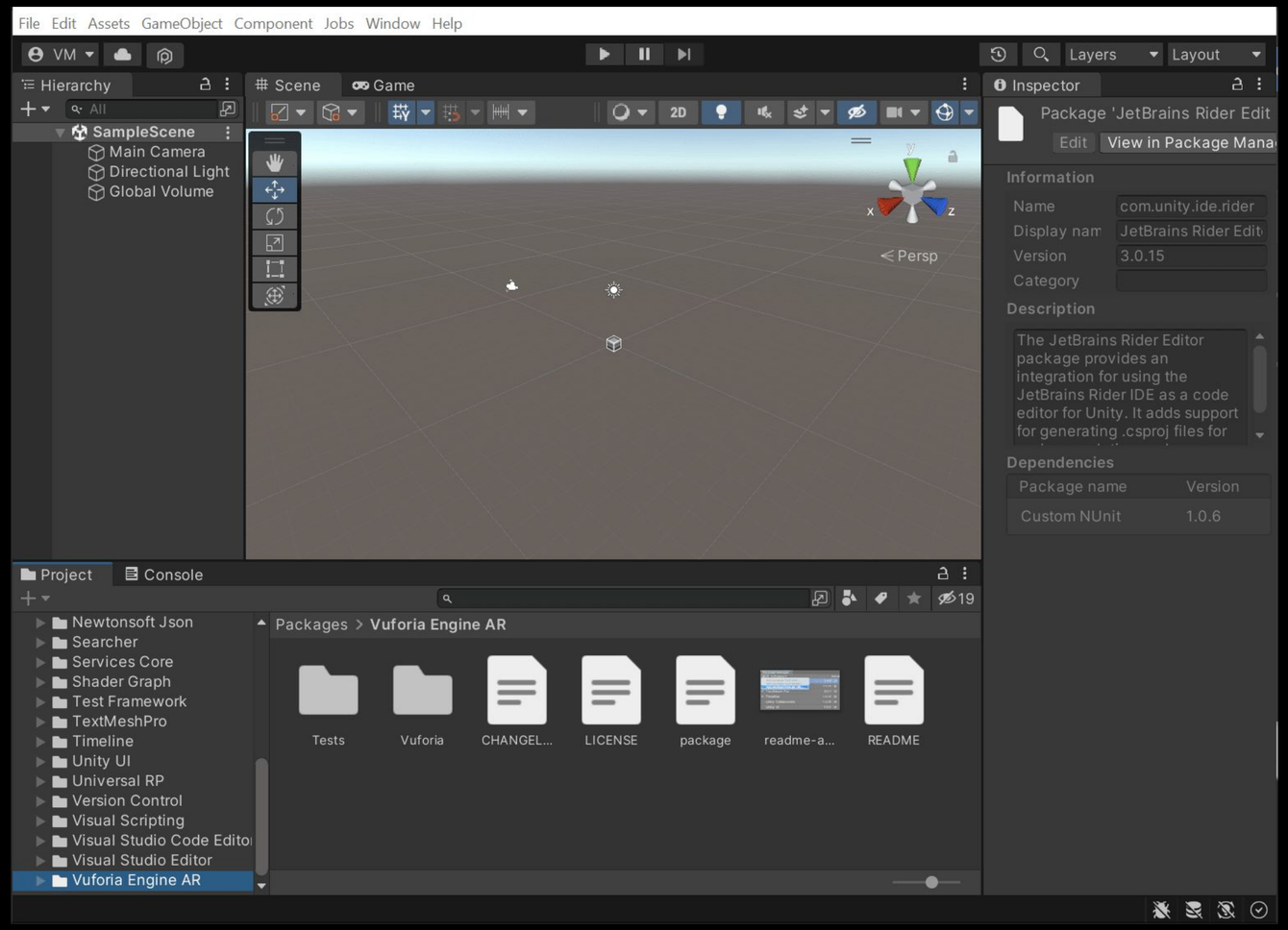
PTC

Vuforia

Download the **SDK** via the Asset Store

- Log in and select **Add to my Assets** Here
- Open a project in Unity
  - Navigate to **Window → Package Manager** and select *Packages: My Assets*
  - Select Vuforia Engine and **Download**, then **Import**
  - Now you can find Vuforia Engine **AR** into your *Packages*

- Add **ARCamera** Gameobject from the Vuforia Engine menu
  - Select the **ARCamera** and click *Open Vuforia Configuration*
  - Select **Add License** and add *Get Basic (* You will have to register*)*, copy and paste the license key into **App License Key** on the Inspector.



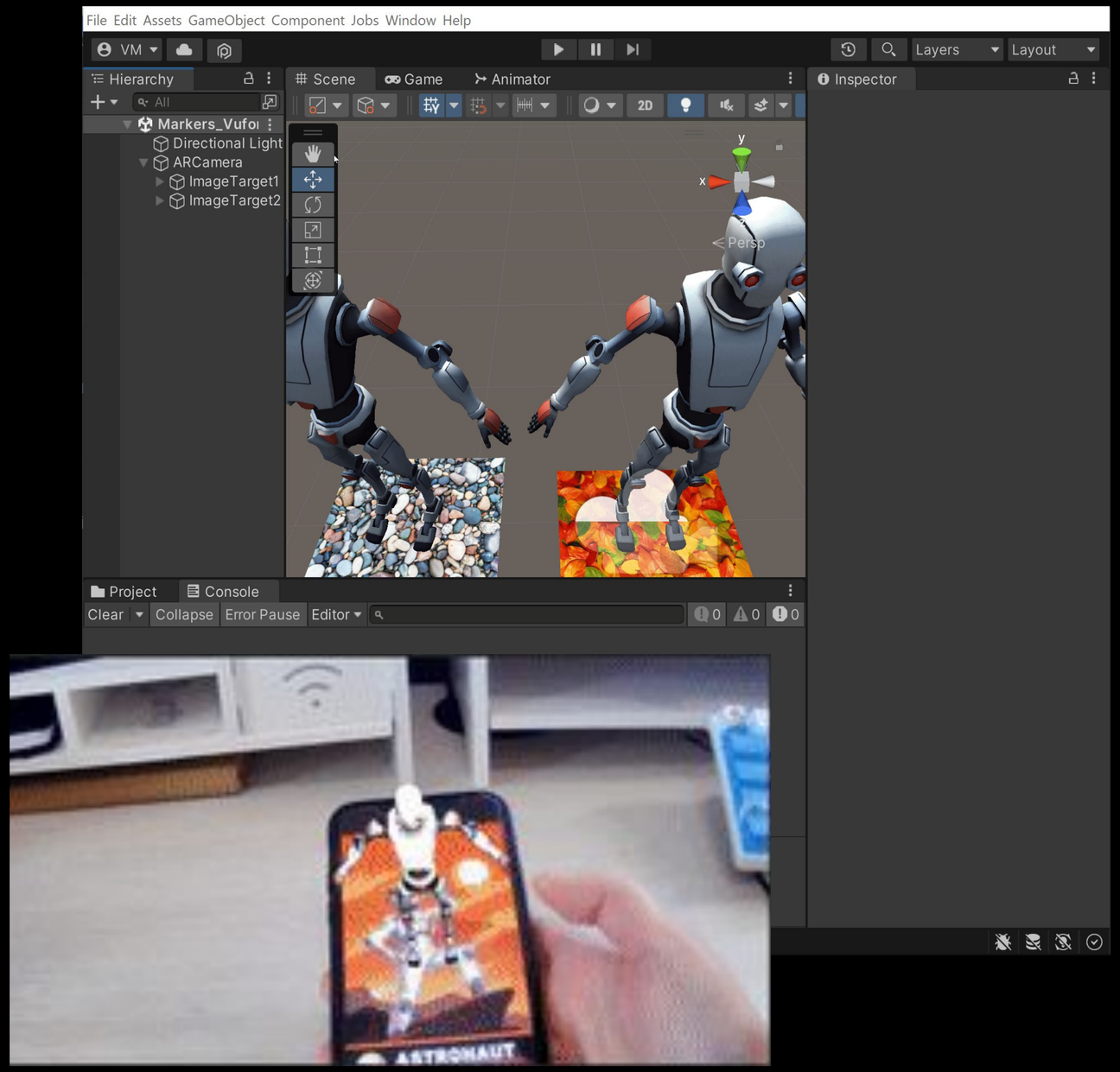
Getting Started with Vuforia

PTC

Vuforia

Set up a basic **AR** Scene - Image Target

- GameObject → Vuforia Engine → **Image Target**
  - Select Image Target and choose **Type : Database** on the Inspector and click Import
  - Download Robot Kyle from the Unity Asset Store here and Import
  - Add your asset as a **Child** of the Image Tracker and Scale it
  - Select Image Target → **Advanced** → Device Tracker Settings → PlayMode → Playmode Type → **Webcam**
  - **Print** or **Download** to a mobile device the Image Tracker
  - **Focus** your webcam into the Image Tracker and press Play



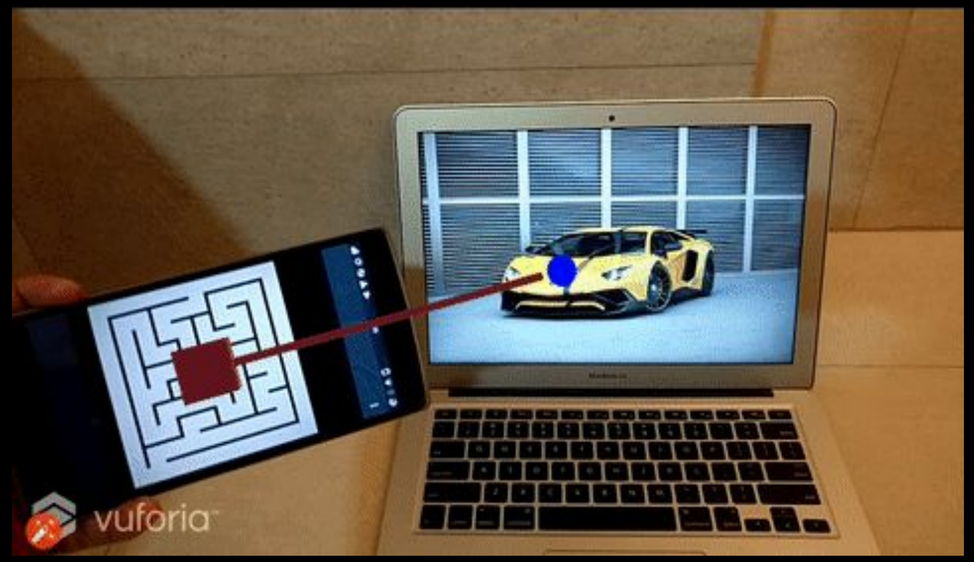
Features Overview

PTC

Vuforia

Other Examples

- Complex Image Targets
- Virtual Buttons
- Model Targets
- And more...



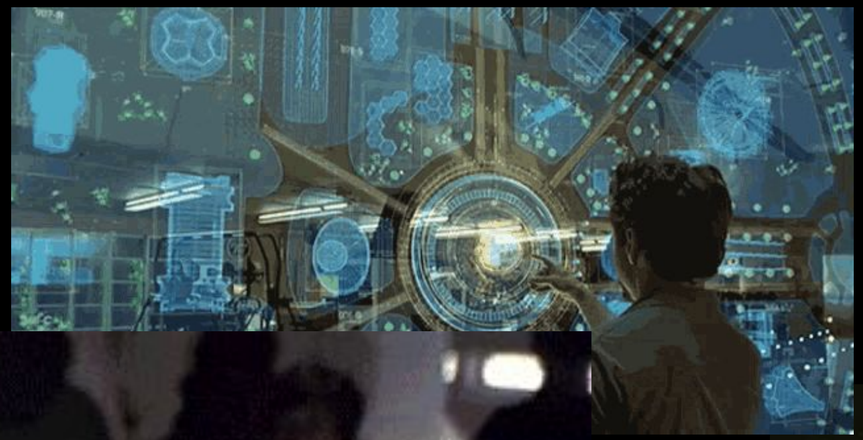
[Core Samples]

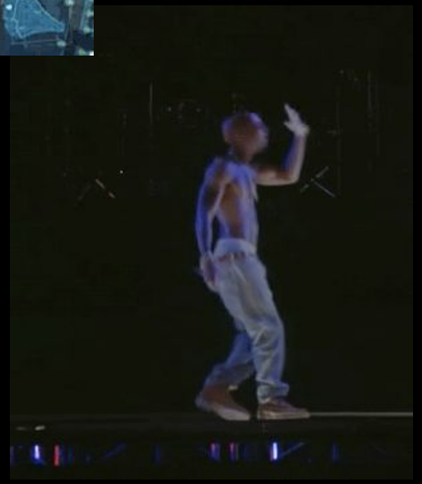
[Vuforia VFX Library]

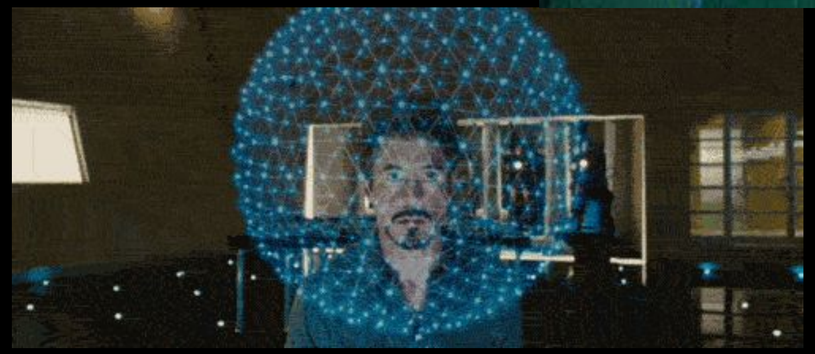
HYPER-REALITY

Keiichi Matsuda

[Watch Here!]





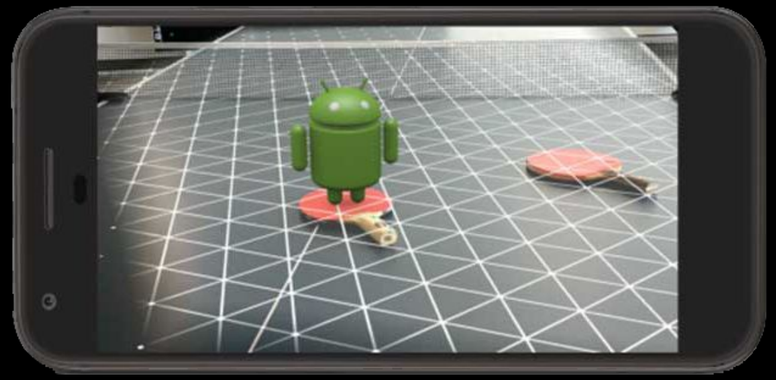


Augmented Reality - Handheld

a technology that superimposes a computer-generated image on a user's view of the real world, thus providing a composite view.

Handheld AR offers the most mainstream AR solutions today on social media. People may not even realize that the AR filters they use on Instagram are augmented reality experiences.

- **Google Android - AR Core**
- **Apple IOS - ARKit**

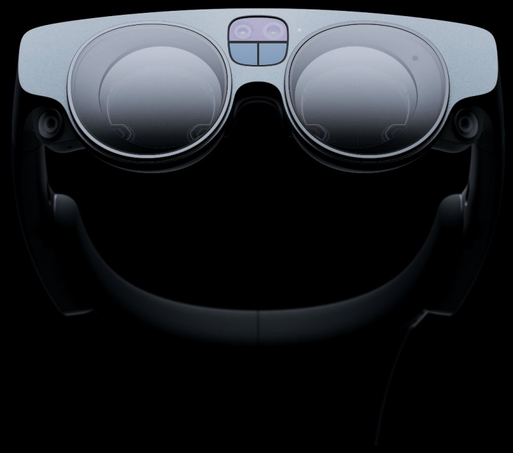


[Unity for AR]

Augmented Reality - Head-mounted

Augmented Reality HMDs also known as Optical head-mounted displays or OHMDs. The displays of these devices are transparent. They allow users to see through them while projecting images and information in front of the users' eyes.

- **Microsoft Hololens**
- **Magic Leap**
- **Nreal**
- **Lenovo A3**

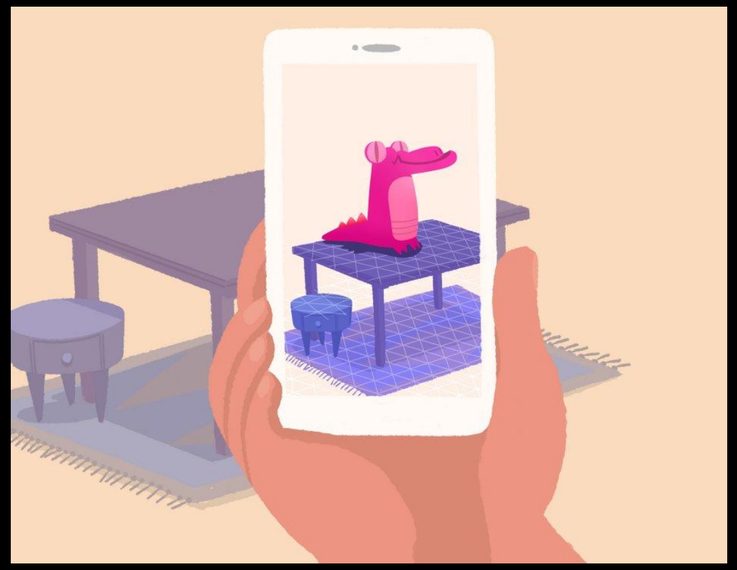


[Intro to Unity for Hololens]

[Intro Unity for Magic Leap]

AR Fundamental Concepts

Motion Tracking

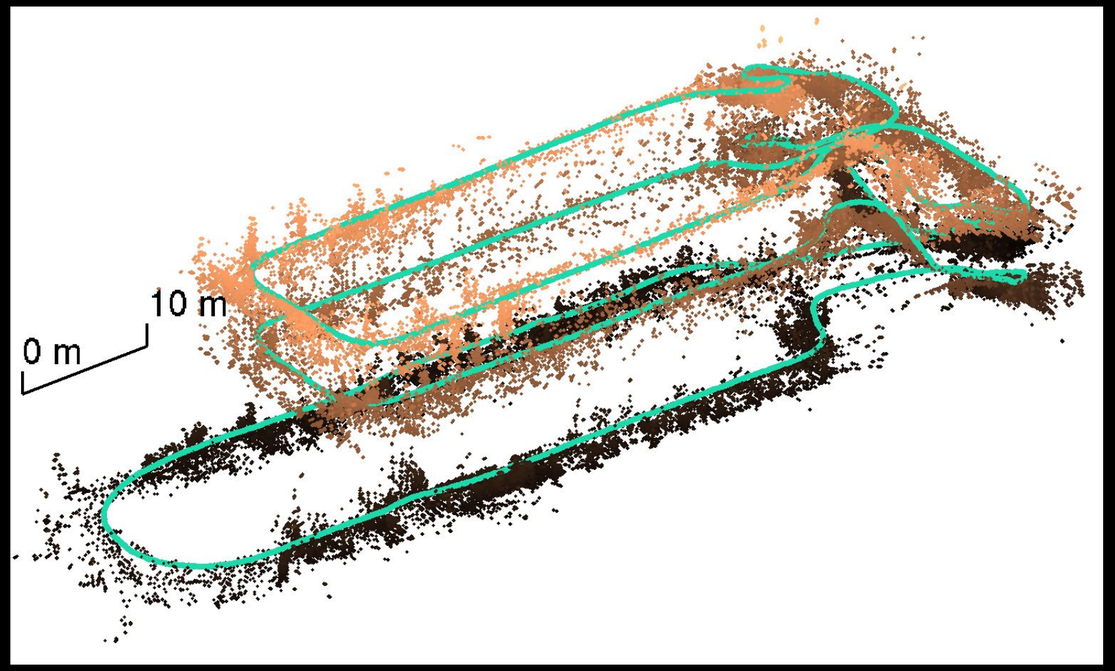


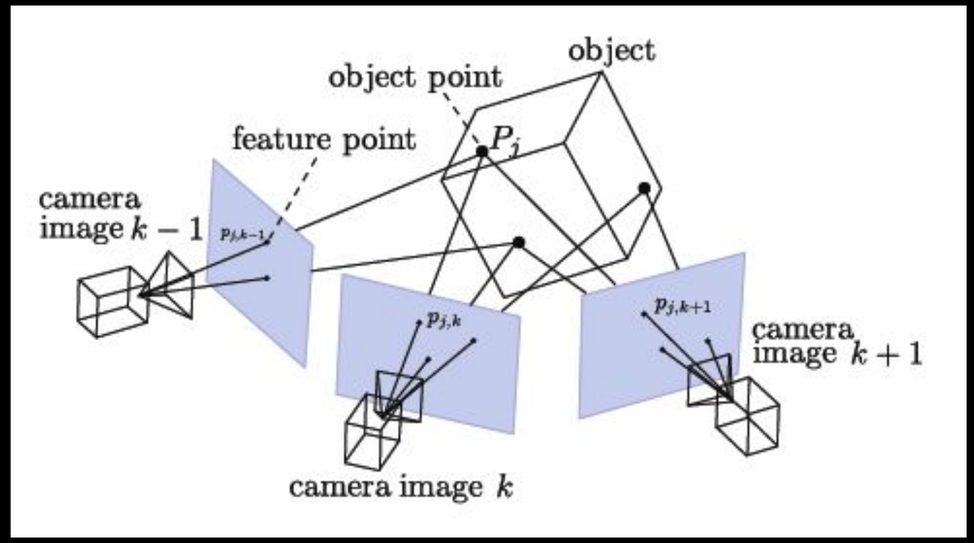
Environmental Understanding

Light Estimation

Let’s jump in!

SLAM Demo





PTC

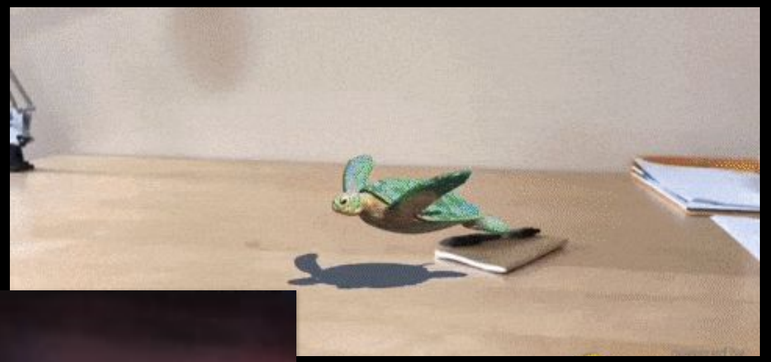
Vuforia

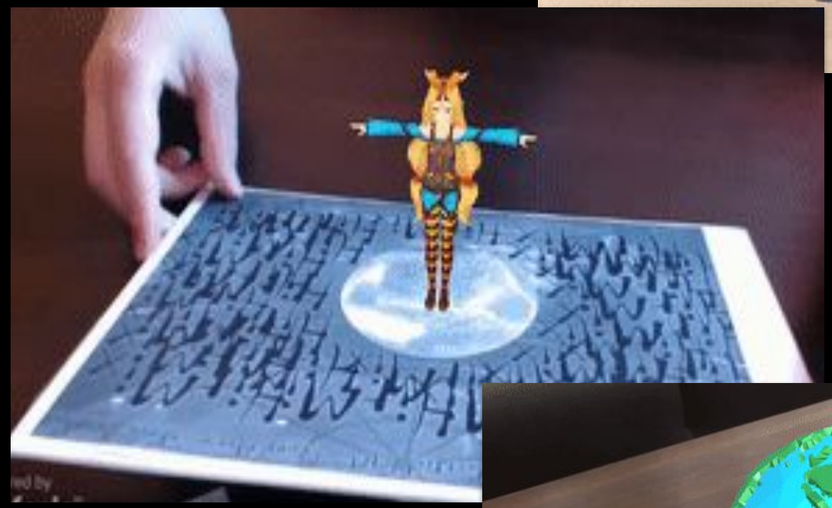
Vuforia Engine Package

SDK for mobile devices that enables the creation of **AR** applications. Available on the Unity Asset Store here

Vuforia Features:

- **Image** Tracking
- **Model** Tracking
- **Area** Tracking
- **Ground Plane** Detection





Resources

- Project Set Up
- Docs

Niantic Lightship ARDK

Create persistent and realistic experiences. ARDK's meshing feature translates the colors produced by the user's camera, runs them through a neural network, and builds a mesh of tessellated triangles that create a machine-readable representation of the physical world.

website | video

Niantic Lightship ARDK

Lightship Augmented Reality Developer Kit (ARDK) is the first product of the Niantic Lightship Platform. It is a cross-platform development kit that brings world understanding technology—depth, meshing, semantics—together with Lightship VPS and multiplayer services for more realistic and diverse AR experiences fused with the real world.

- **Depth** : Generate 3D depth information from a user's surroundings in real time, using just the device's camera.
- **Semantic Segmentation :** Easily and automatically classify objects, such as "ground" or "sky", from the user's camera data.
- **Meshing :** Build a 3D mesh representation of objects in the world, based on camera images.
- **GameBoard :** Enable smart object placement, procedural gameplay, and character navigation.
- **Shared AR :** Users can use your app to locate and orient themselves in the same AR space, and then use that shared space to collaborate in virtual activities.
- **Lightship VPS :** Use Lightship VPS to create immersive experiences in which your users interact with persistent, shareable AR content at real-world locations. With Lightship VPS you can place virtual objects that persist in the real-world and can be found and shared by others.

Microsoft

Mixed Reality Toolkit

Microsoft MRTK3 is distributed as a set of **packages** that are imported into Unity using the **Mixed Reality Feature Tool** for Unity and the Unity Package Manager (UPM).

Microsoft MRTK3 leverages the **Unity XR Subsystem Management** infrastructure for writing extensible modules that can help provide cross-platform support for features like speech and hand tracking.

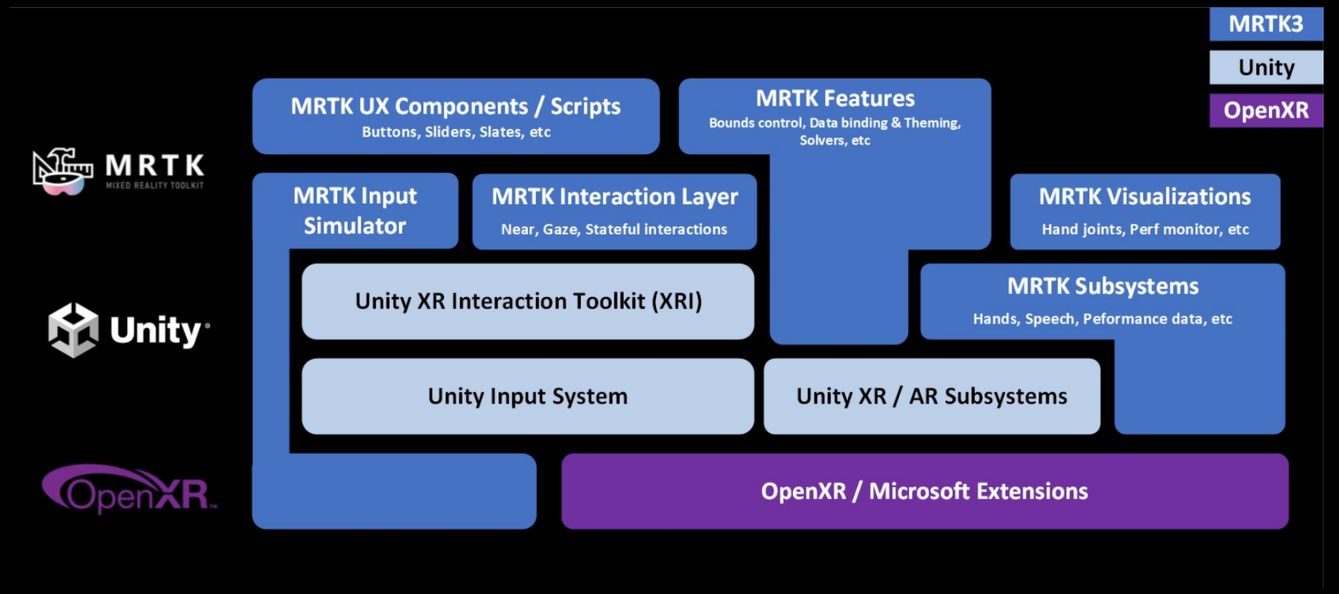
[Docs]

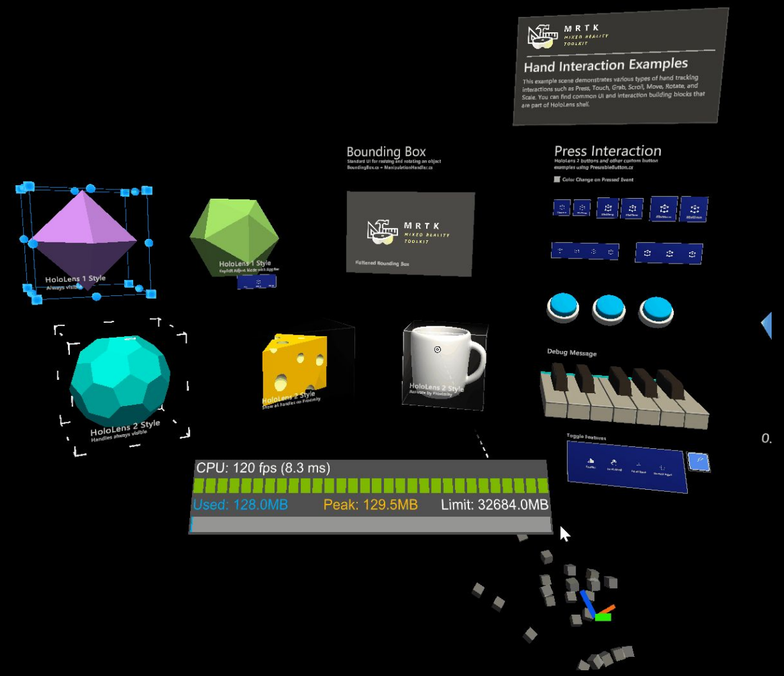
[Initial Setup for MRTK3]

[MRTK3 packages]

[Subsystems]

[MRTK2 to MRTK3 concept migration guide]

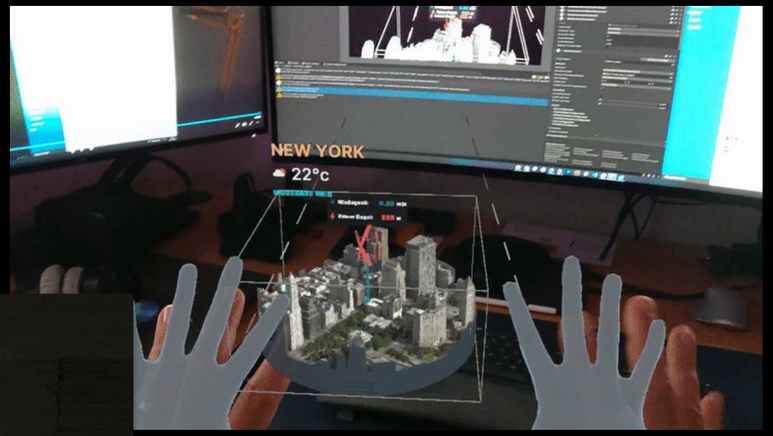


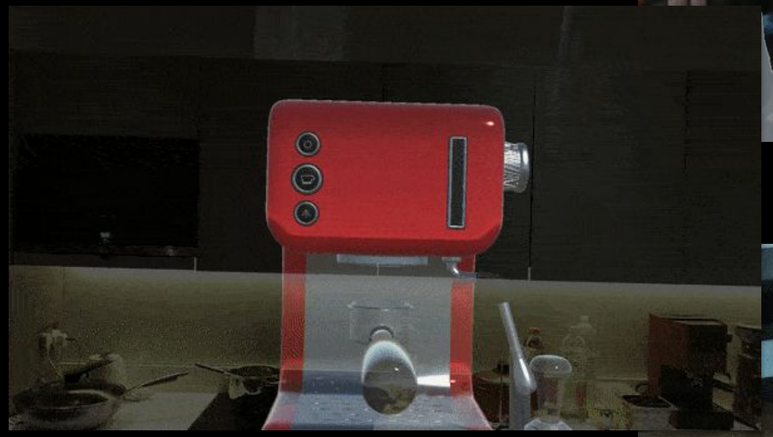


Microsoft MRTK

Lightship Augmented Reality Developer Kit (ARDK) is the first product of the Niantic Lightship Platform. It is a cross-platform development kit that brings world understanding technology—depth, meshing, semantics—together with Lightship VPS and multiplayer services for more realistic and diverse AR experiences fused with the real world.

- **Input System**
- **Hand Tracking**
- **Eye Tracking**
- **UI Controls**
- **Solvers**
- **Spatial Awareness**
- **Speech & Dictation**





Unity

AR Foundation

AR Foundation allows you to work with augmented reality platforms in a multi-platform way within Unity.

This package presents an interface for Unity developers to use, but doesn't implement any AR features itself. To use AR Foundation on a target device, you also need separate packages for the target platforms officially supported by Unity:

ARCore XR Plug-in on **Android**
ARKit XR Plug-in on **iOS**
OpenXR Plug-in on **HoloLens**

Docs

Feature support per platform

You can refer to this table to understand which parts of AR Foundation are relevant on specific platforms:

	ARCore	ARKit	OpenXR
Device tracking	✓	✓	✓
Plane tracking	✓	✓	
Point clouds	✓	✓	
Anchors	✓	✓	✓
Light estimation	✓	✓	
Environment probes	✓	✓	
Face tracking	✓	✓	
2D Image tracking	✓	✓	
3D Object tracking		✓	
Meshing		✓	✓
2D & 3D body tracking		✓	
Collaborative participants		✓	
Human segmentation		✓	
Raycast	✓	✓	
Pass-through video	✓	✓	
Session management	✓	✓	✓
Occlusion	✓	✓	

Note: To use ARCore cloud anchors, download and install Google's ARCore Extensions for Unity's AR Foundation.

AR Foundation Samples

Download sample project via the **github**

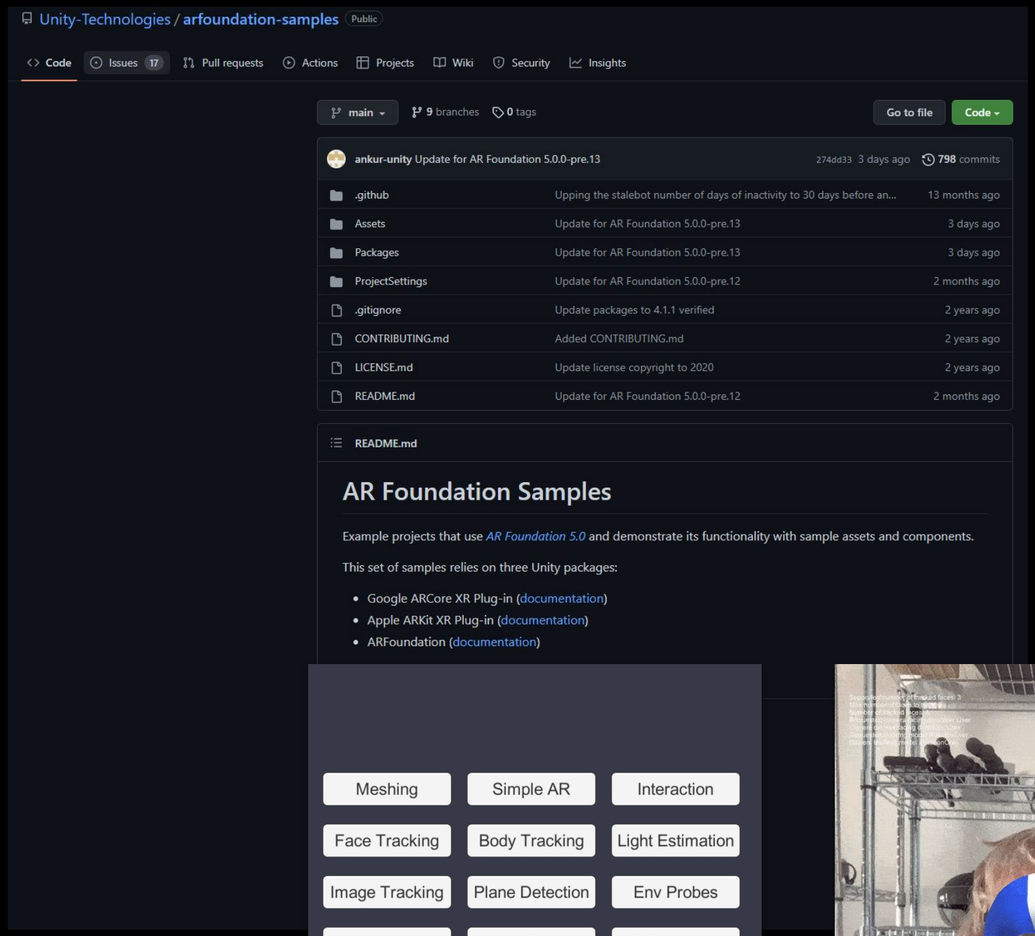
- Get Project Here
- Clone or download this repository to a workspace on your drive
  - Click the ↓ Code button on this page to get the URL to clone with Git or click Download ZIP to get a copy of this repository that you can extract
- Open a project in Unity
  - Download, install, and run Unity Hub
  - In the Installs tab, select Locate or Add to find or install Unity 2019.4 LTS or later
  - In the Projects tab, click Add
  - Browse to the VR folder within your downloaded copy of this repository and click Select Folder
  - Click the project which should now be added to the list to open the project
- To run the AR Foundation sample on a headset, go to File > Build Settings and build the app. 1

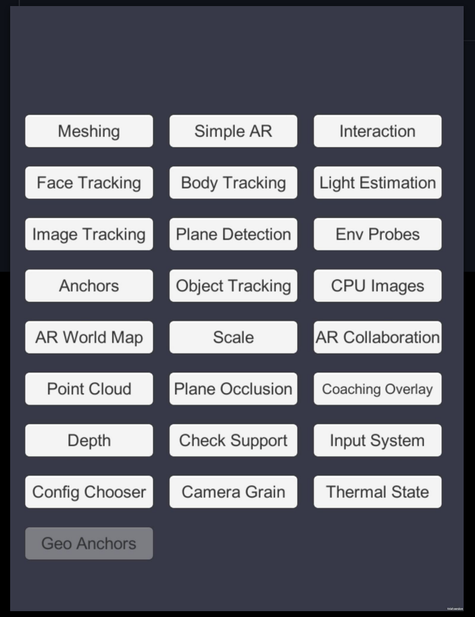
Samples (v4.2)

Samples (v5)

Building for iOS

Building for Android





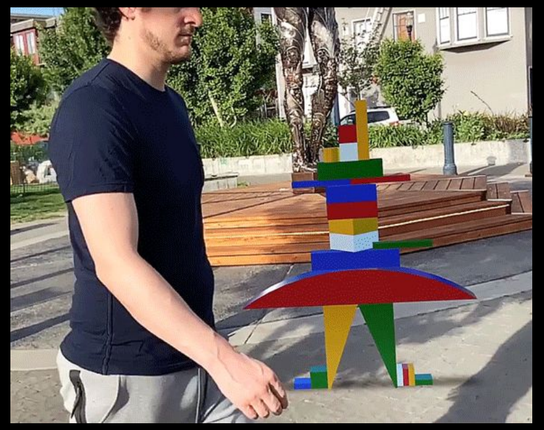
Let’s jump in!

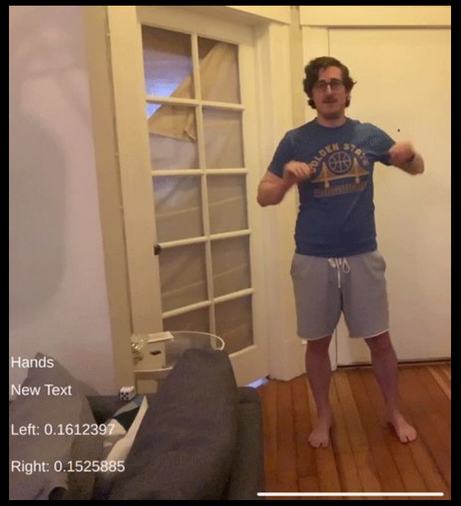
AR Foundations Demo

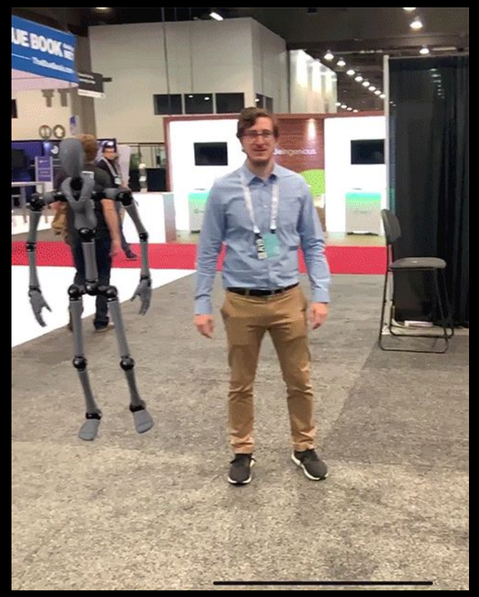
Project Assets:
https://bit.ly/R2MV_XpressoAR_assets

AR Foundation Samples

Meshing	Simple AR	Interaction
Face Tracking	Body Tracking	Light Estimation
Image Tracking	Plane Detection	Env Probes
Anchors	Object Tracking	CPU Images
AR World Map	Scale	AR Collaboration
Point Cloud	Plane Occlusion	Coaching Overlay
Depth	Check Support	Input System
Config Chooser	Camera Grain	Thermal State
Geo Anchors		







[Samples (v4.2)]

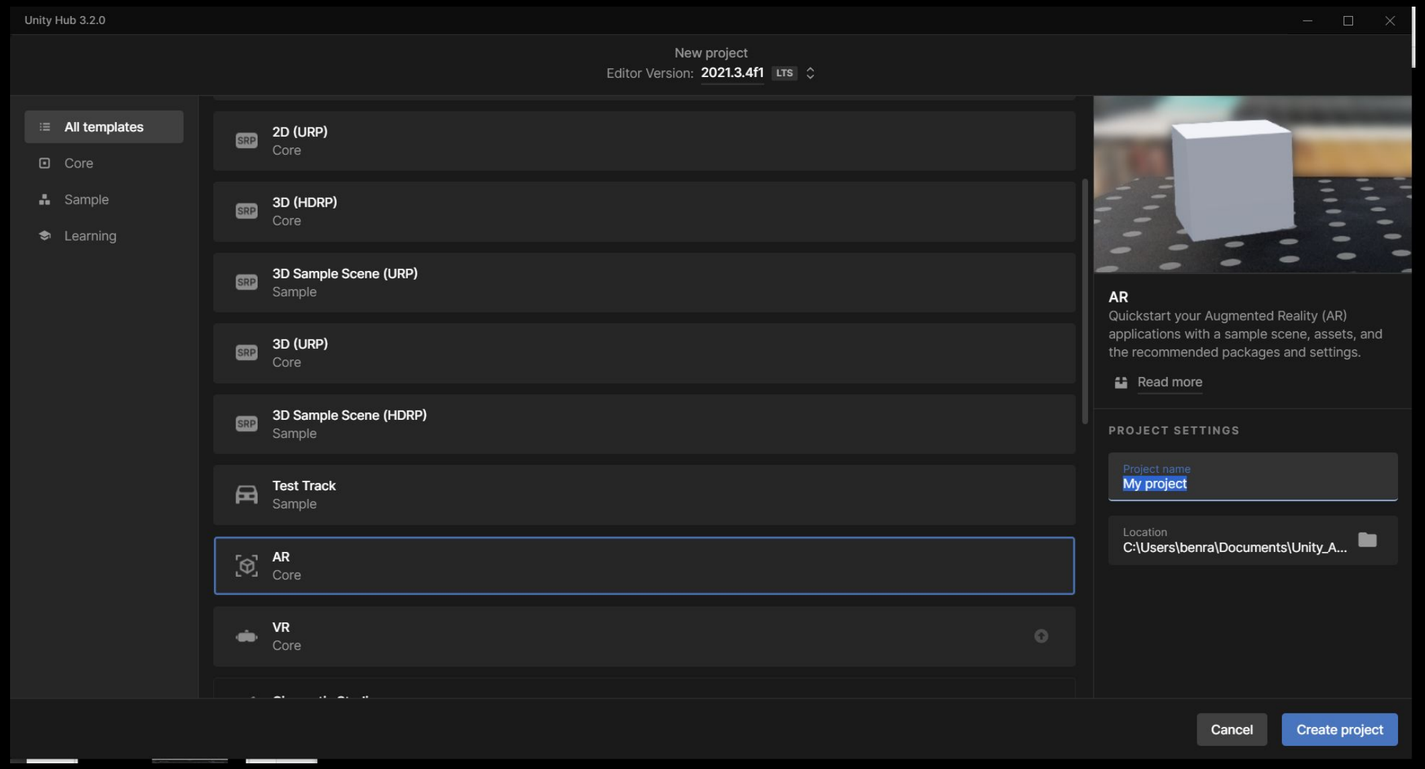
[Samples (v5)]

Unity AR project template

Unity's AR project template provides a starting point for augmented reality development in Unity. The template pre-installs the right packages needed for AR development, and the Scene hierarchy is pre-configured for AR.

Unity directly supports the following AR platforms:

- ARCore
- ARKit
- HoloLens
- Magic Leap



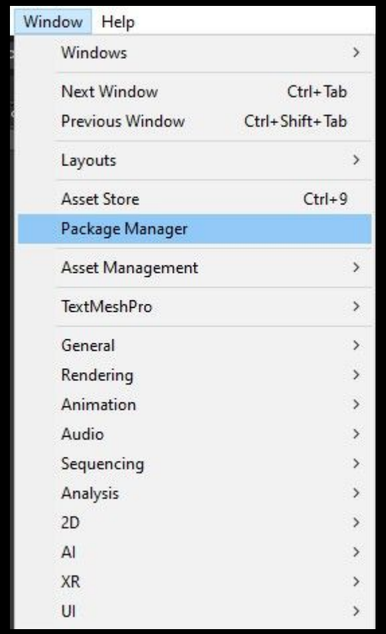
[Documentation]

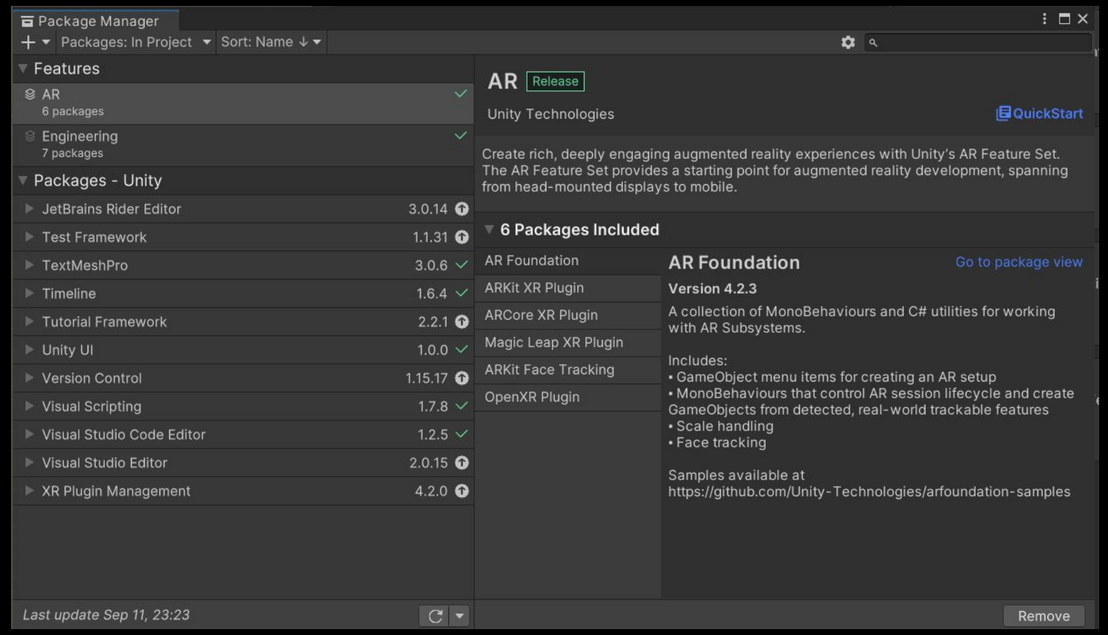
AR Foundation Packages

Create rich, deeply engaging augmented reality experiences with Unity's AR Feature Set. The AR Feature Set provides a starting point for augmented reality development, spanning from head-mounted displays to mobile.

- **AR Foundation**
- **ARKit XR Plugin**
- **ARCore XR Plugin**

- **XR Plugin Management**





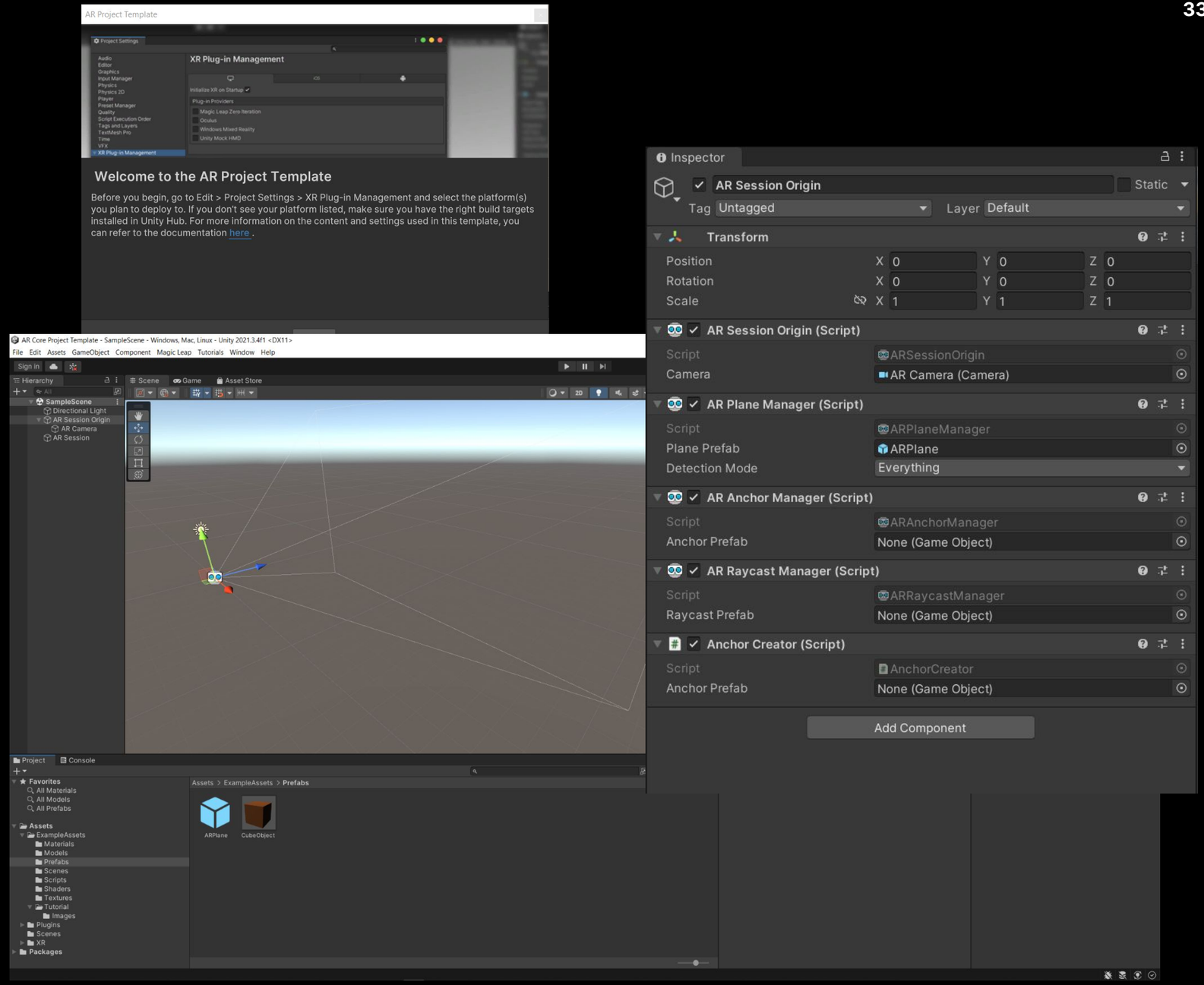
AR Sample Scene

The template contains a Scene named SampleScene. This scene contains sample materials and models. It additionally contains all the tutorials, and a script that allows you to spawn anchors through input.

You can use this Scene as a reference, or you can remove the example Assets from it and use it as a starting point for your own Project.

- **AR Session**
- **AR Session Origin**

[Documentation]



PlaceOnPlanes script

Listens for touch events and performs an AR raycast from the screen touch point. AR raycasts will only hit detected trackables like feature points and planes.

PlaceOnPlanes.cs

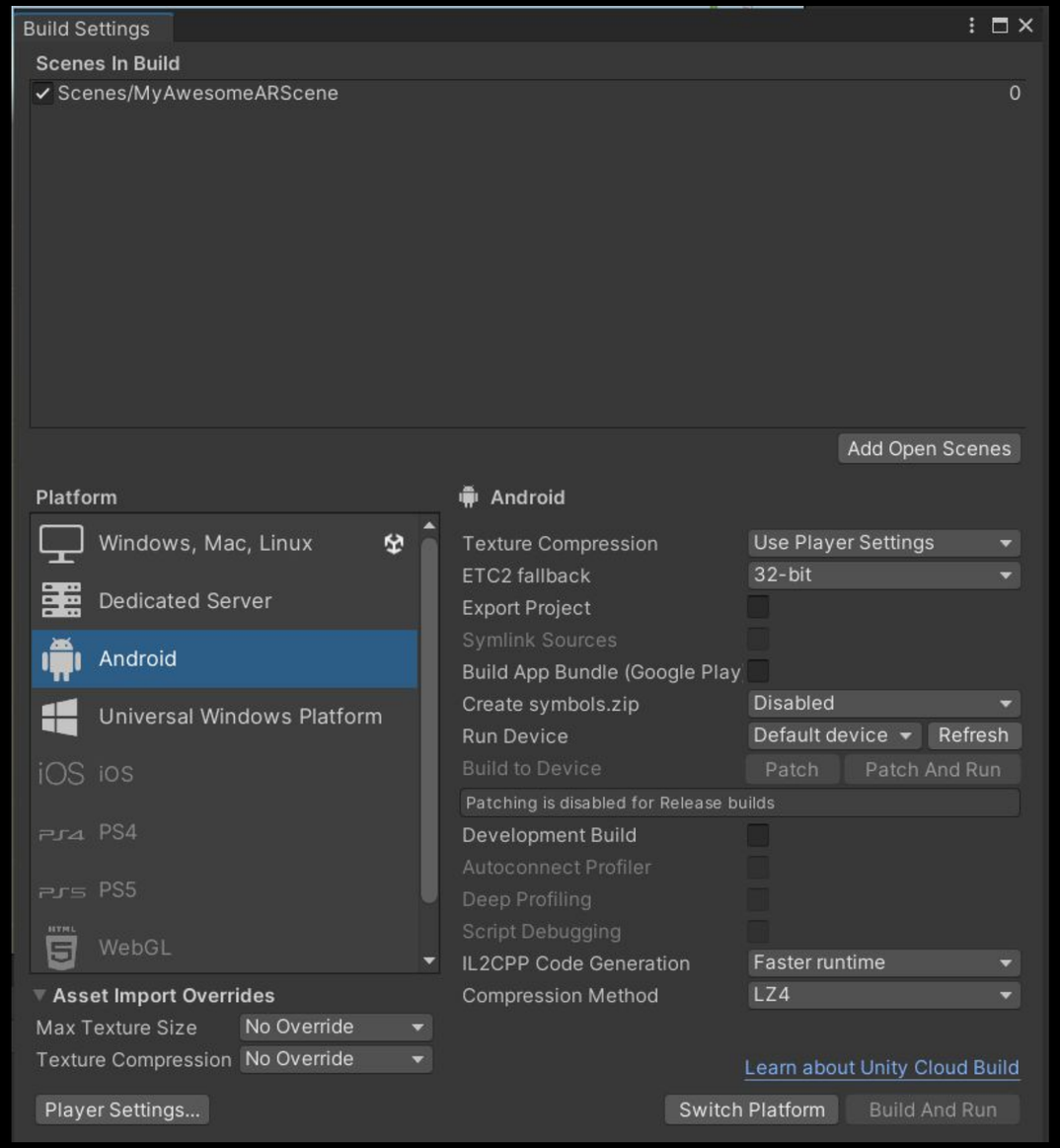
AR Foundation Samples

```
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.XR.ARFoundation;
using UnityEngine.XR.ARSubsystems;

namespace UnityEngine.XR.ARFoundation.Samples
{
    /// <summary>
    /// Listens for touch events and performs an AR raycast from the screen touch point.
    /// AR raycasts will only hit detected trackables like feature points and planes.
    ///
    /// If a raycast hits a trackable, the <see cref="placedPrefab"/> is instantiated
    /// and moved to the hit position.
    /// </summary>
    [RequireComponent(typeof(ARRaycastManager))]
    public class PlaceOnPlane : MonoBehaviour
    {
        [SerializeField]
        [Tooltip("Instantiates this prefab on a plane at the touch location.")]
        GameObject m_PlacedPrefab;

        /// <summary>
        /// The prefab to instantiate on touch.
        /// </summary>
        public GameObject placedPrefab
        {
            get { return m_PlacedPrefab; }
            set { m_PlacedPrefab = value; }
        }

        /// <summary>
        /// The object instantiated as a result of a successful raycast intersection with a plane.
        /// </summary>
        public GameObject spawnedObject { get; private set; }

        void Awake()
        {
            m_RaycastManager = GetComponent<ARRaycastManager>();
        }

        bool TryGetTouchPosition(out Vector2 touchPosition)
        {
            if (Input.touchCount > 0)
            {
                touchPosition = Input.GetTouch(0).position;
                return true;
            }

            touchPosition = default;
            return false;
        }

        void Update()
        {
            if (!TryGetTouchPosition(out Vector2 touchPosition))
                return;

            if (m_RaycastManager.Raycast(touchPosition, s_Hits, TrackableType.PlaneWithinPolygon))
            {
                // Raycast hits are sorted by distance, so the first one
                // will be the closest hit.
                var hitPose = s_Hits[0].pose;

                if (spawnedObject == null)
                {
                    spawnedObject = Instantiate(m_PlacedPrefab, hitPose.position, hitPose.rotation);
                }
                else
                {
                    spawnedObject.transform.position = hitPose.position;
                }
            }
        }

        static List<ARRaycastHit> s_Hits = new List<ARRaycastHit>();

        ARRaycastManager m_RaycastManager;
    }
}
```

Build Settings

Unity can build your application for different platforms and with different settings. This documentation describes how to define the scenes, target platform, and settings for your build.

- **Android**
- **iOS**



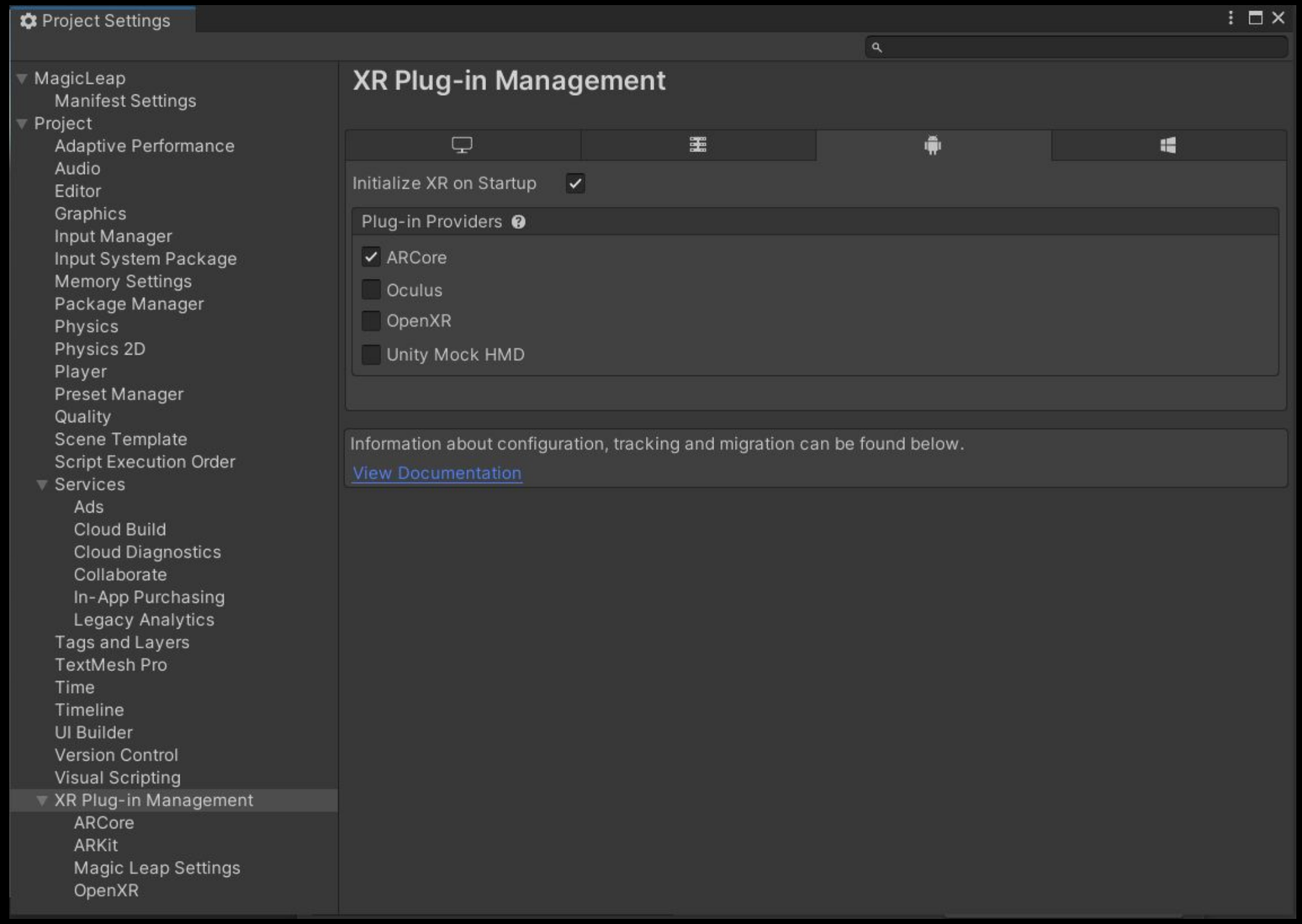
[Build Settings Documentation]

XR Plug-in Management

Support for XR hardware is delivered via plug-ins. Plug-ins allow you to develop for a universal XR platform so you don't need to modify your project for different XR headsets. Supporting new XR hardware is as easy as adding the Plug-in Providers in Project Settings.

Currently, XR plug-ins are available for Oculus, Windows Mixed Reality, and Magic Leap headsets. Also included is the MockHMD plug-in, which allows for XR development without a device. It's possible to install multiple XR plug-ins to support multiple headsets. In this scenario, the application will look for the headset supported by the first listed loader (covered later in this tutorial). If the headset is not available, the application will search for the next in the list, continuing through all listed loaders in order.

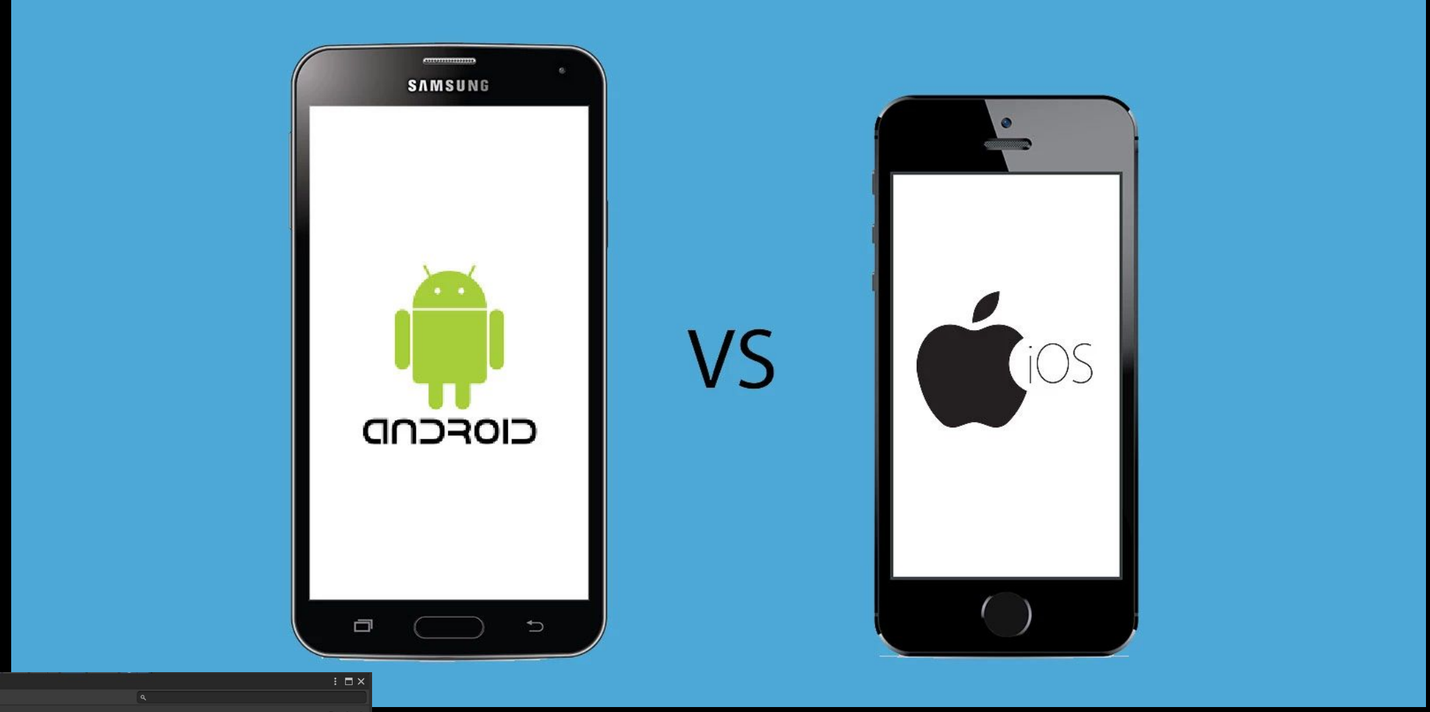
- **ARCore**
- **ARKit**
- **Oculus**
- **OpenXR**
- **WindowMixedReality**

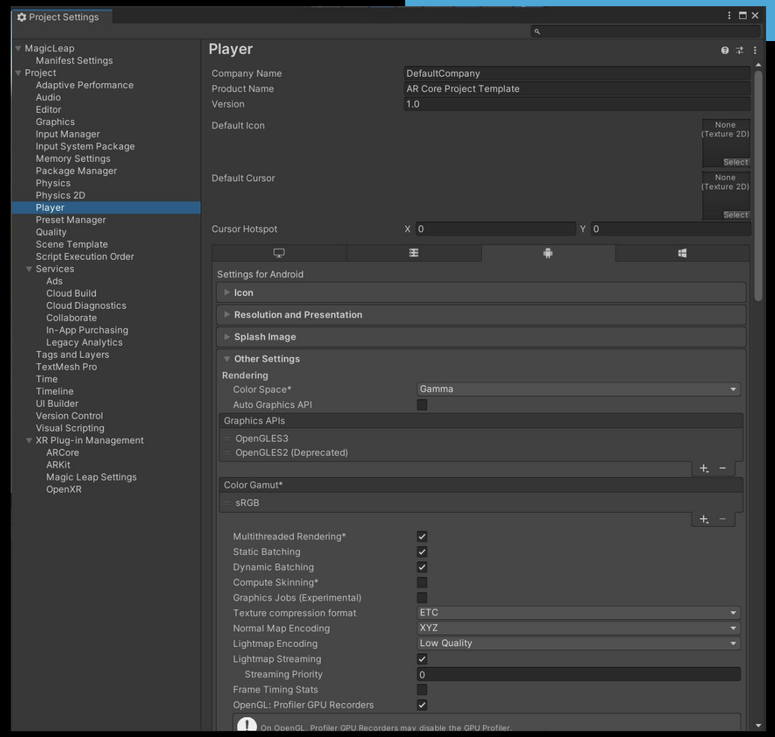


Setting Up the Development Environment for AR Applications

There are a few tools we must set up before developing Unity applications for either iOS or Android devices. If developing for iOS, this involves setting up Xcode (Apple's integrated development environment (IDE)) on your development machine. Xcode is required to deploy our Unity builds onto our iOS devices.

[Unity Learn Tutorial]

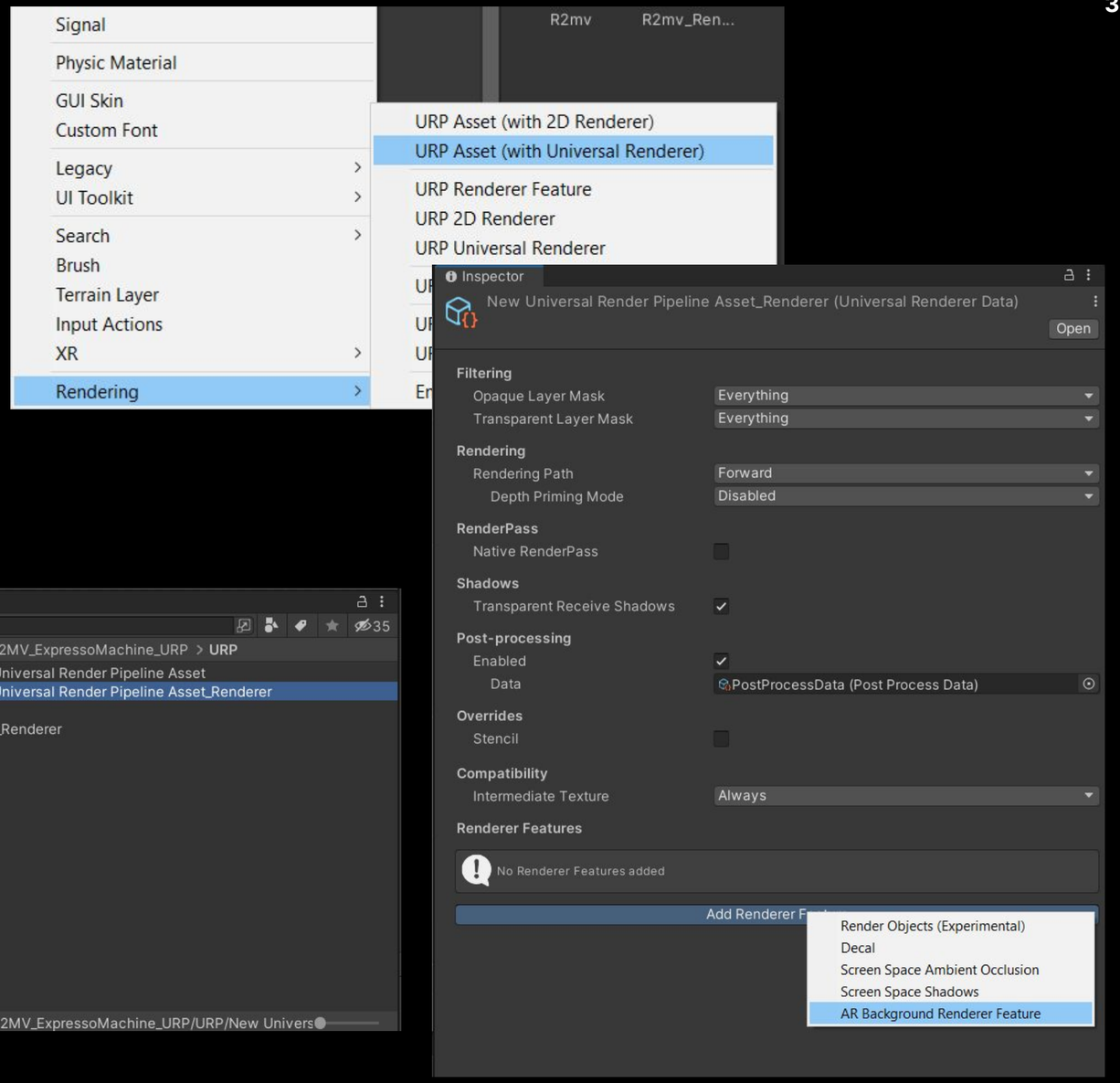




Universal Render Pipeline

AR Foundation supports the Universal Render Pipeline (URP) versions 7.0.0 or later.

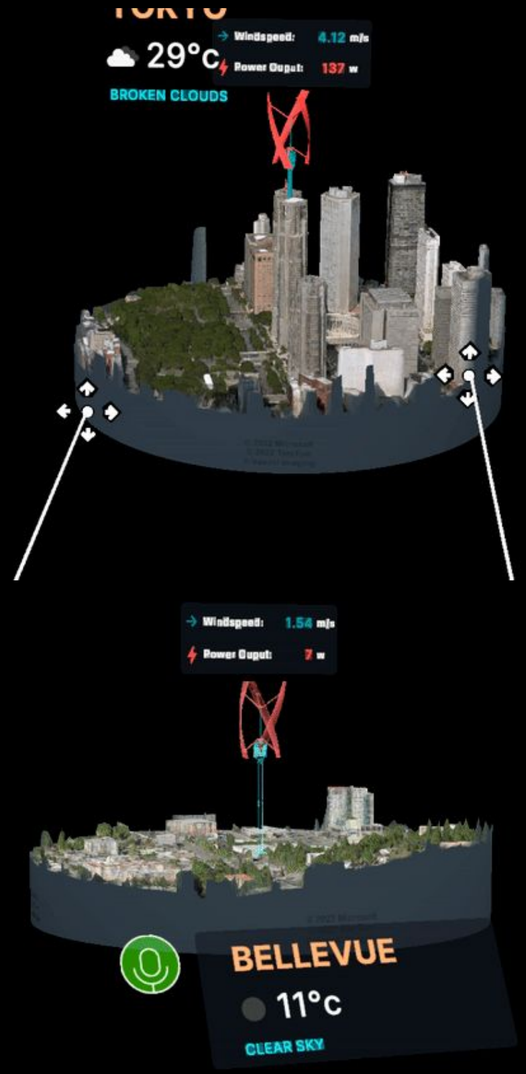
Note: Projects made using URP are not compatible with the High Definition Render Pipeline or the built-in Unity rendering pipeline. Before you start development, you must decide which render pipeline to use in your Project.

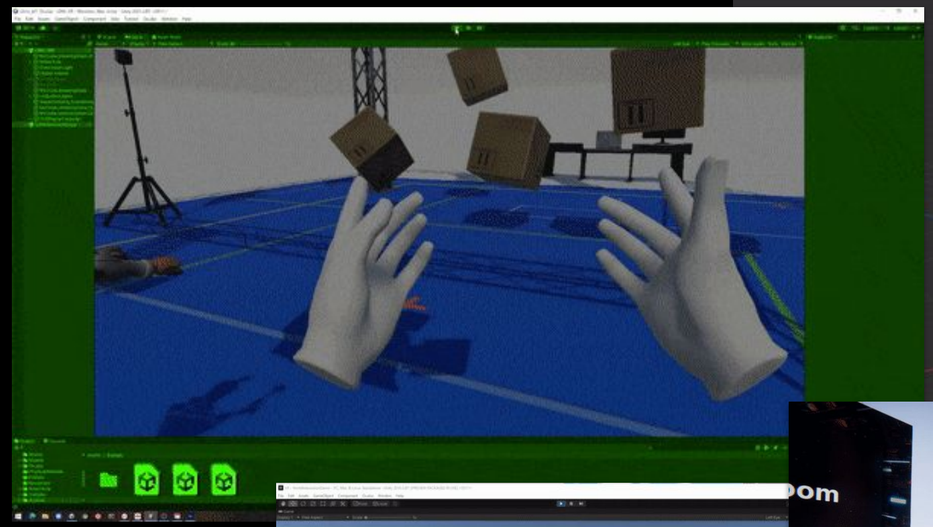


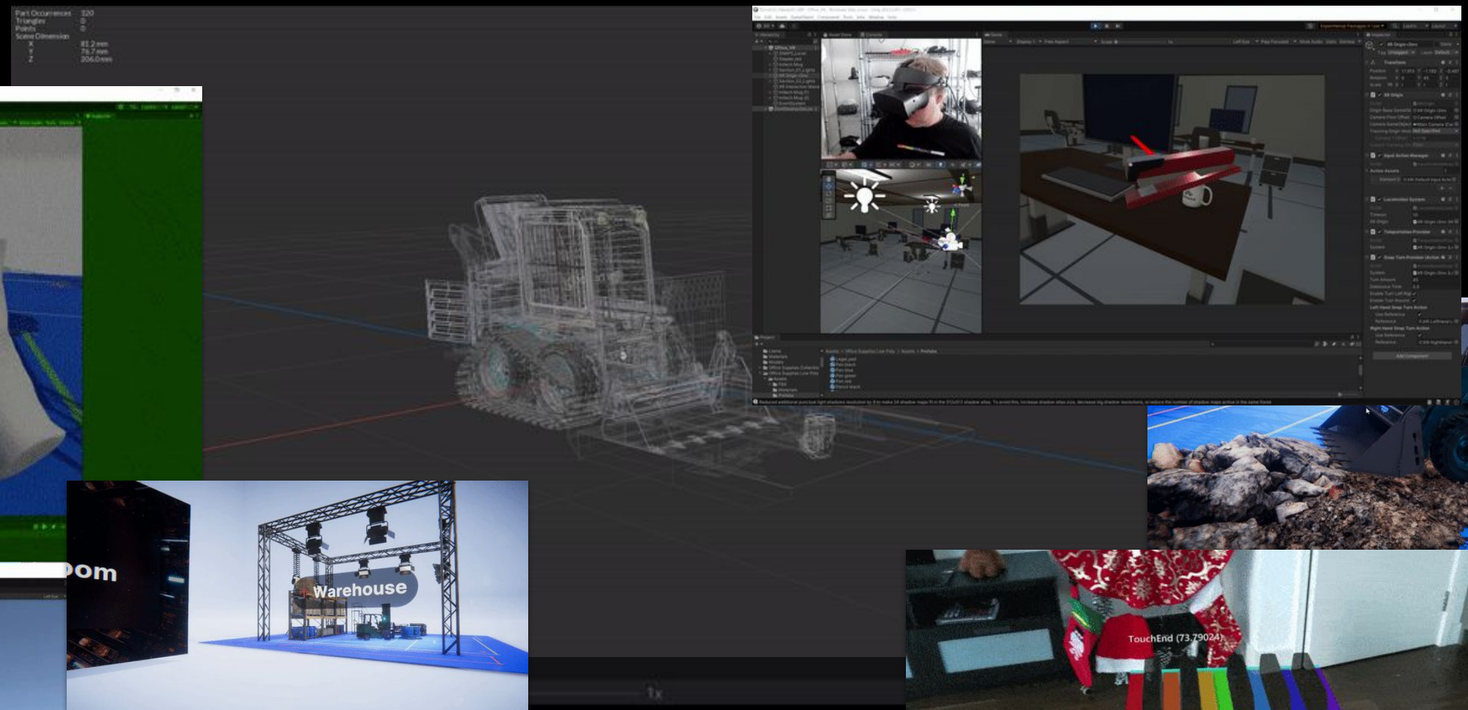
[Configuring AR Camera Background]

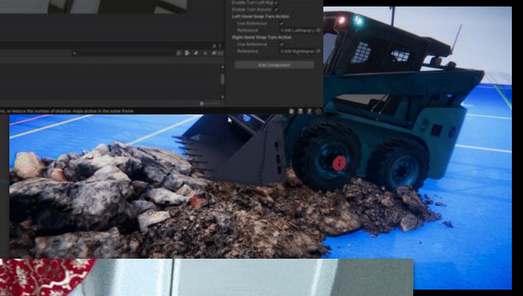
Wind Turbine Simulation

→ Porting to Hololens
→ Adding speech using Microsoft Azure!
→ Creating a custom Visual Scripting node



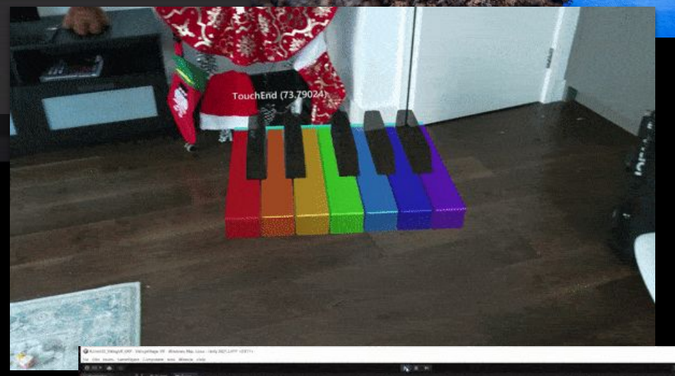


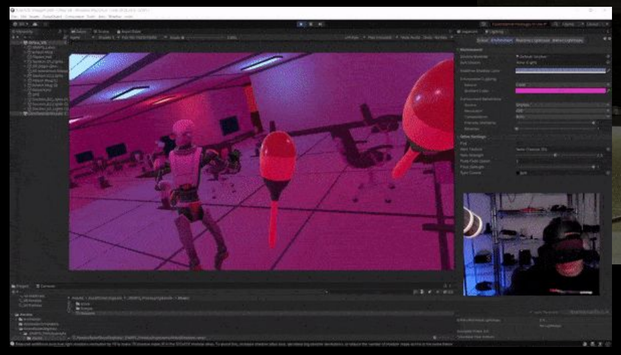


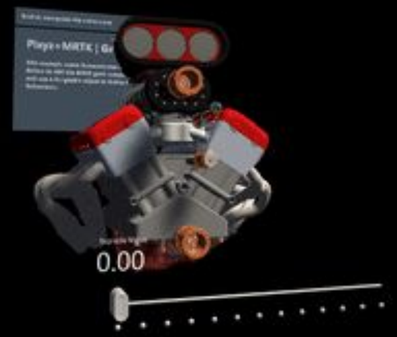


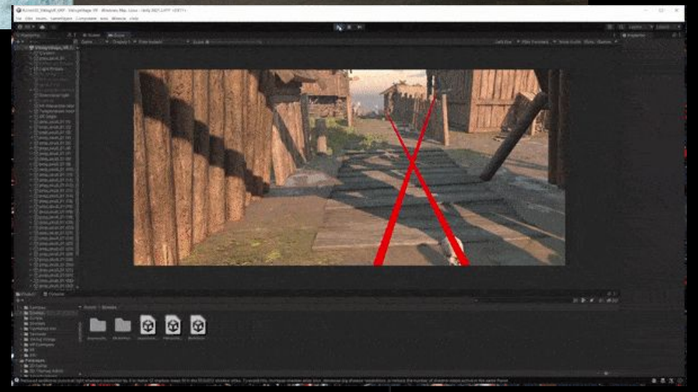
Warehouse

Complex Grab Interactions
Snap Interactions


TouchEnd (73.79024)




0.00

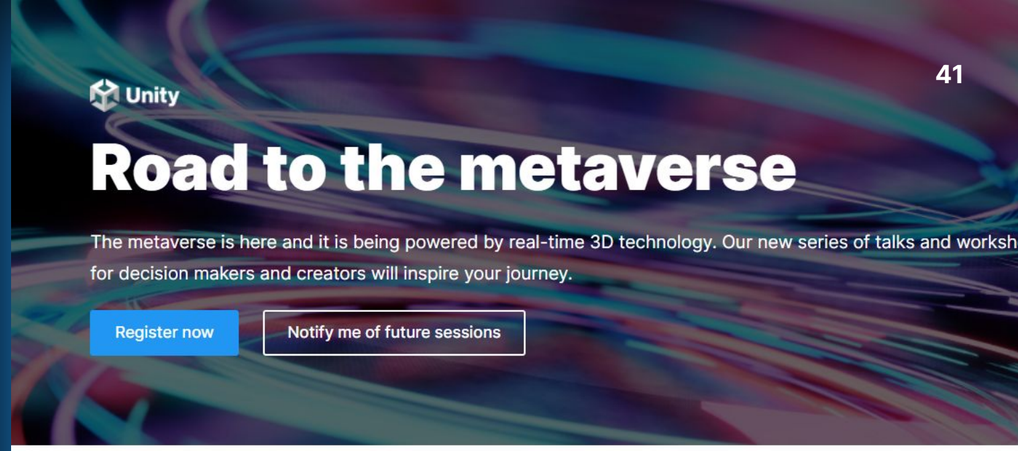


Unity®

Workshops

https://create.unity.com/road-to-metaverse

April	**Welcome** to the Metaverse!
May	Bring your **digital** and **physical assets** into Unity
June	Add **interactivity** to your **immersive** experience
July	How to bring your **real-time 3D digital twin data** into Unity
August	Let's get real: An introduction to AR, VR, MR, XR and more
September	Augmenting reality (AR): Bring digital objects into the real world
October	**Build immersive worlds in virtual reality (VR)**
November	Improve learning retention with **immersive training** experiences
December	Build a Microsoft HoloLens experience for **training in AR**
January	Integrate **cloud-based IoT data** into your XR experience
February	Masterclass: Build a **collaborative multi user** experience
March	Build **geolocated** and **social** experiences in **AR**



Monthly sessions for creators and decision makers

Get access to curated sessions designed for creators and decision makers.

As a leader in real-time 3D technology, Unity is poised to prepare its community for new modes of interaction t
enable data discovery and insight. Explore strategies and get training on the tools required to build connected
immersive experiences across industries.

Creator series

Build a strong foundation and develop your skills with Unity experts during these monthly sessions. Ranging from beginner to advanced, these deep-dives are curated specifically for creators looking to gain the knowledge and skill to succeed in the metaverse.

View all sessions

Executive series

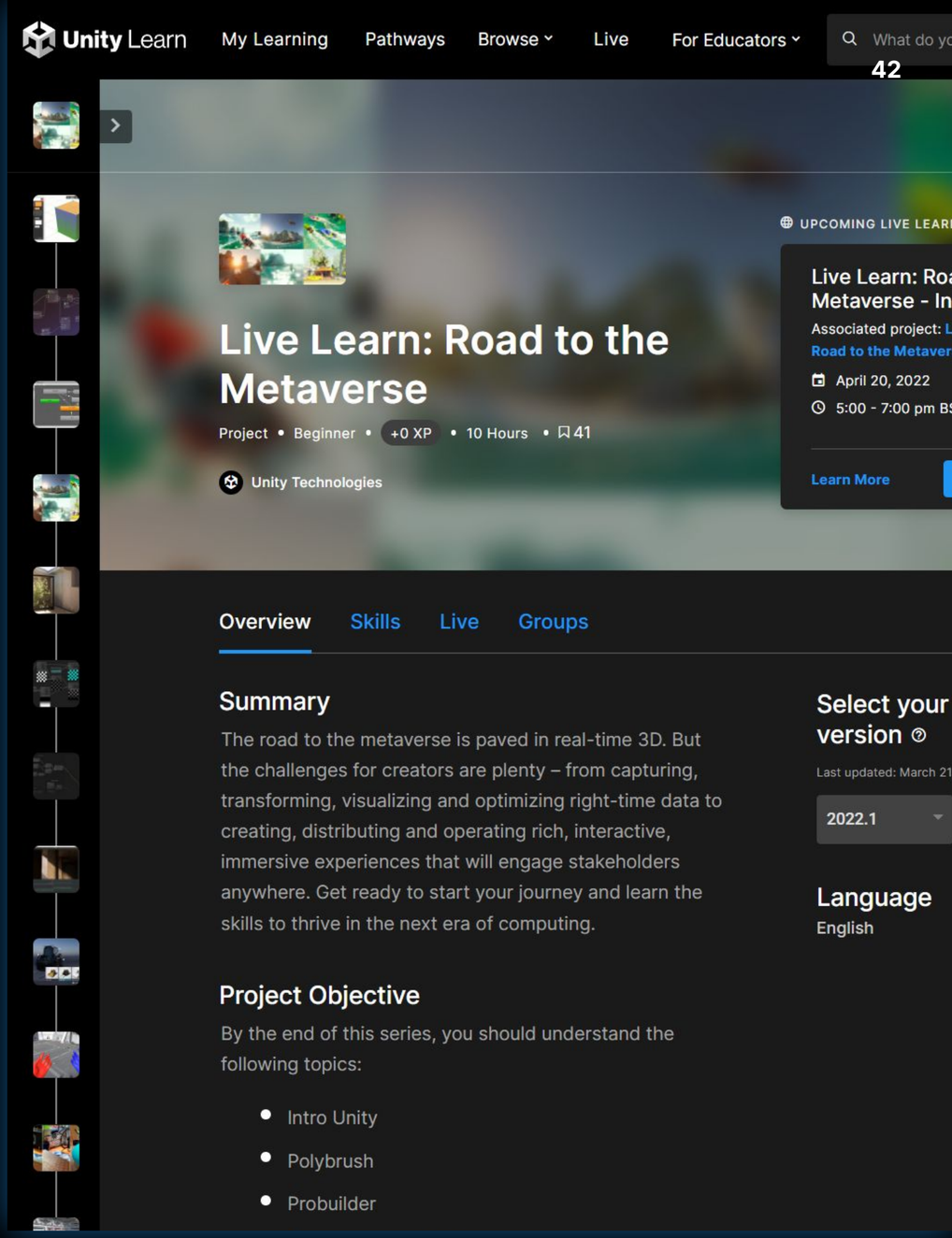
Journey with Ryan Peterson, Unity's VP of professional services, as he and special guests tackle the why and ho
behind this new era of consumer and social experience. These intimate chats are designed to spark your imagination, and will end with a live Q&A.

View all sessions

Learn Live

https://learn.unity.com/project/road-to-the-metaverse-live

March	**Prototyping** and world building.
April	Intro to **Visual Scripting**
May	**Animation**
June	Intro to **Universal Render Pipeline**
July	Intro to High Definition Render Pipeline
August	Shader Graph
September	**VFX Graph**
October	**Post Processing**
November	**UI** Building
December	Intro to **VR**
January	Intro to **AR**
February	Intro to **Spatial Audio**



Executive series

https://create.unity.com/road-to-metaverse#executive

EXECUTIVE SERIES

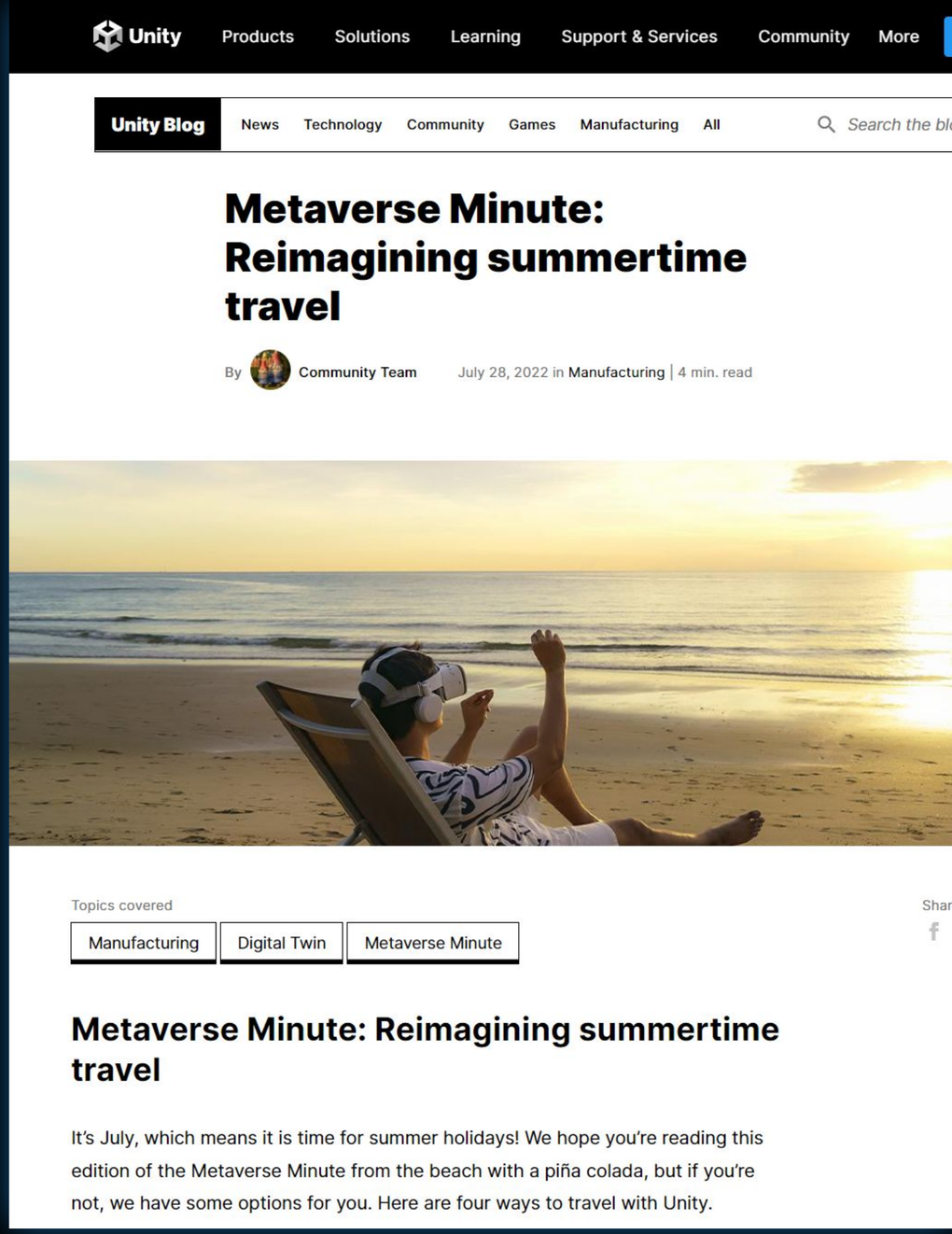
Building successful B2C brands in the metaverse

August 25, 2022
9 am PT / 12 pm ET / 5 pm BST
Duration: 60 minutes

Join Ryan and Media.Monks' SVP, Tim Dillon, as they discuss what it's going to take for B2C brands to make it in the metaverse. Hear from Tim as he shares lessons he's picked up from working with large consumer brands and walk away with strategic tips on how to get started, how to leverage real-time 3D technology to make an impact, and much more.

The Metaverse Minute

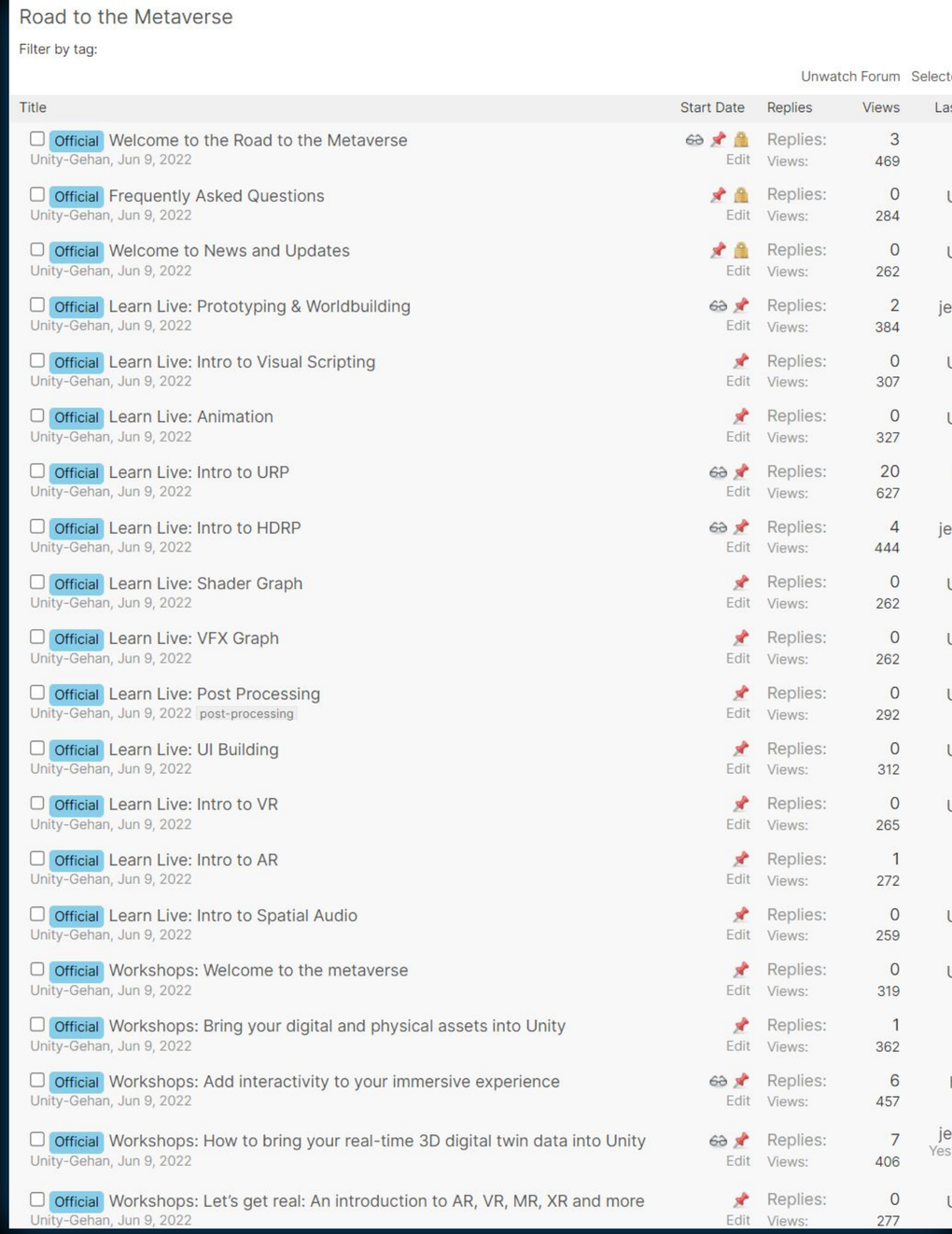
https://blog.unity.com/topic/metaverse-minute



Road to the Metaverse:

Forums!

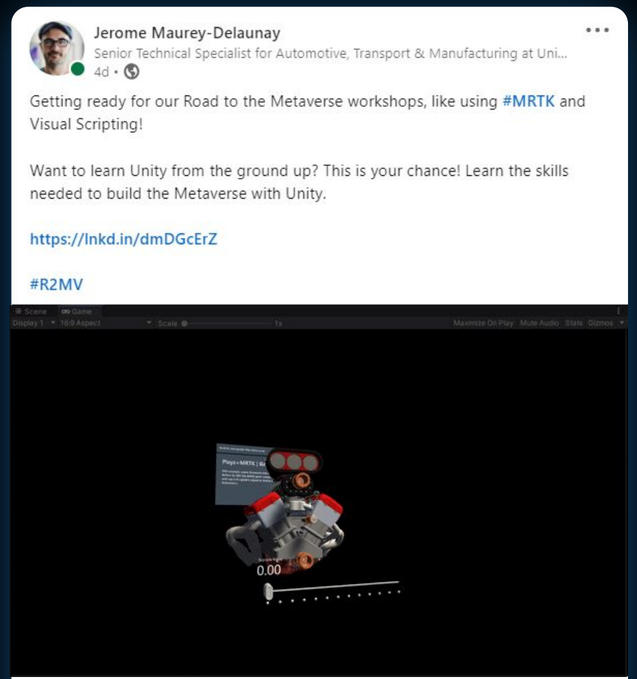
https://forum.unity.com/forums/road-to-the-metaverse.798/



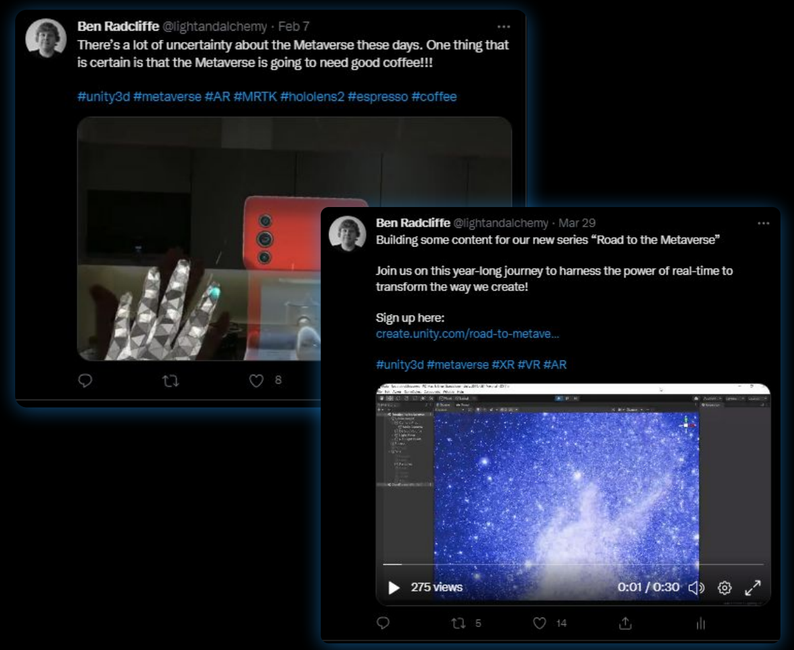
Social

→ Please use **#R2MV**

Jerome Maurey-Delaunay
@jeromemaurey



Ben Radcliffe
@lightandalchemy



Q&A

Any questions?

Unity®

Thank You

2022

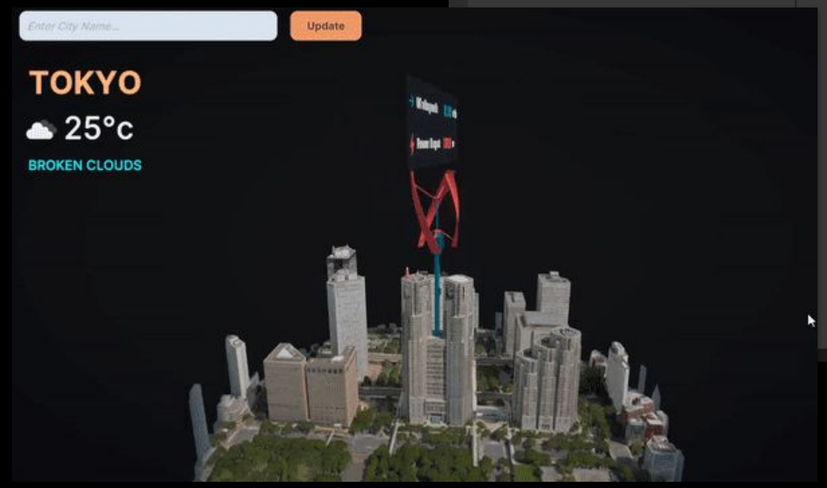
Survey available here!

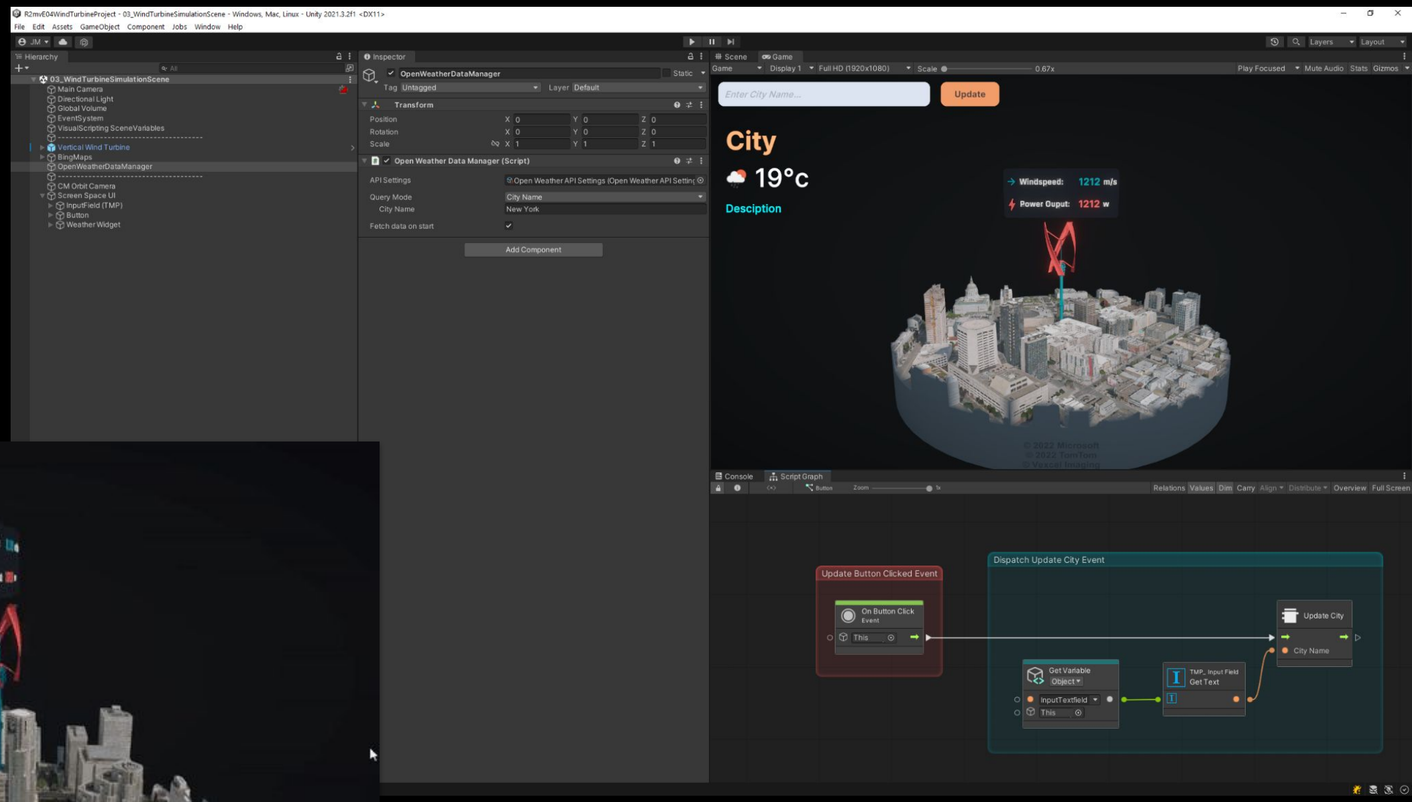
Overtime

Building a Wind turbine Simulation, Part II

Hololens Edition!

Previously...


Enter City Name...
Update
TOKYO
25°c
BROKEN CLOUDS

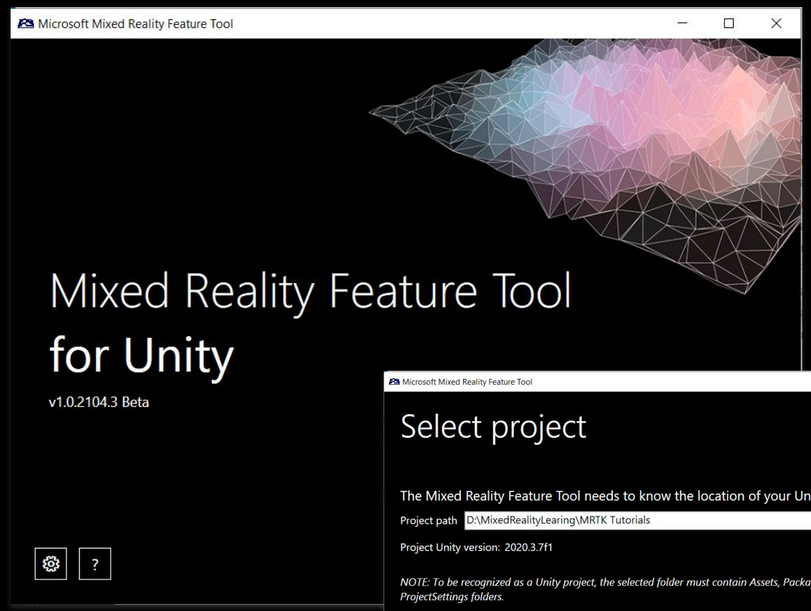

Update
City
19°c
Desciption
Windspeed: 1212 m/s
Power Ouput: 1212 w
Update Button Clicked Event
On Button Click
Dispatch Update City Event
Update City
City Name
Get Variable
inputTextfield
TMP_ Input Field
Get Text

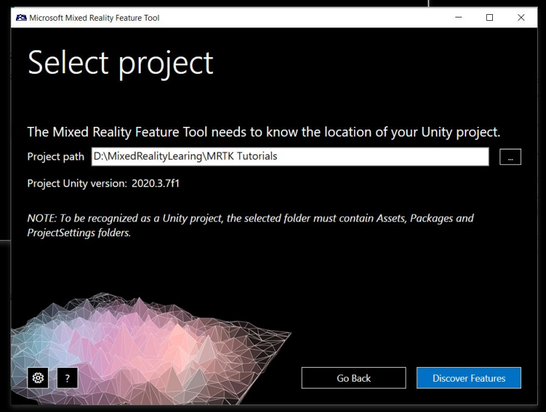
Mixed Reality Feature Tool

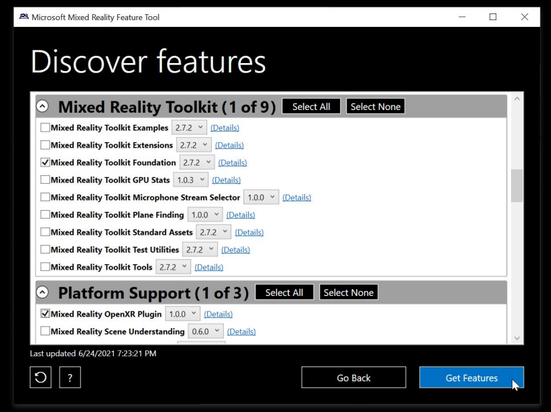
Use Mixed Reality Feature Tool to discover, update, and add Mixed Reality feature packages into Unity projects.

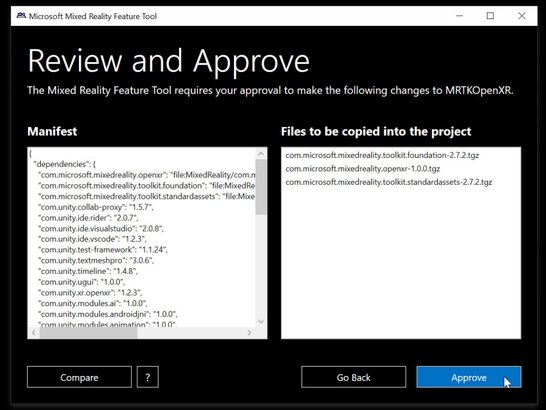
You can search packages by name or category, see their dependencies, and even view proposed changes to your projects manifest file before importing.

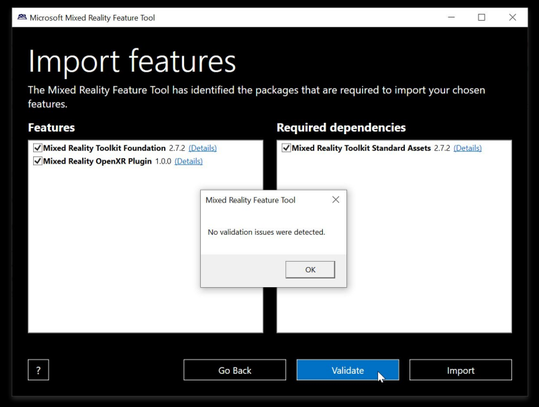
Download the latest version from the Microsoft Download Center.











Let’s dive in!

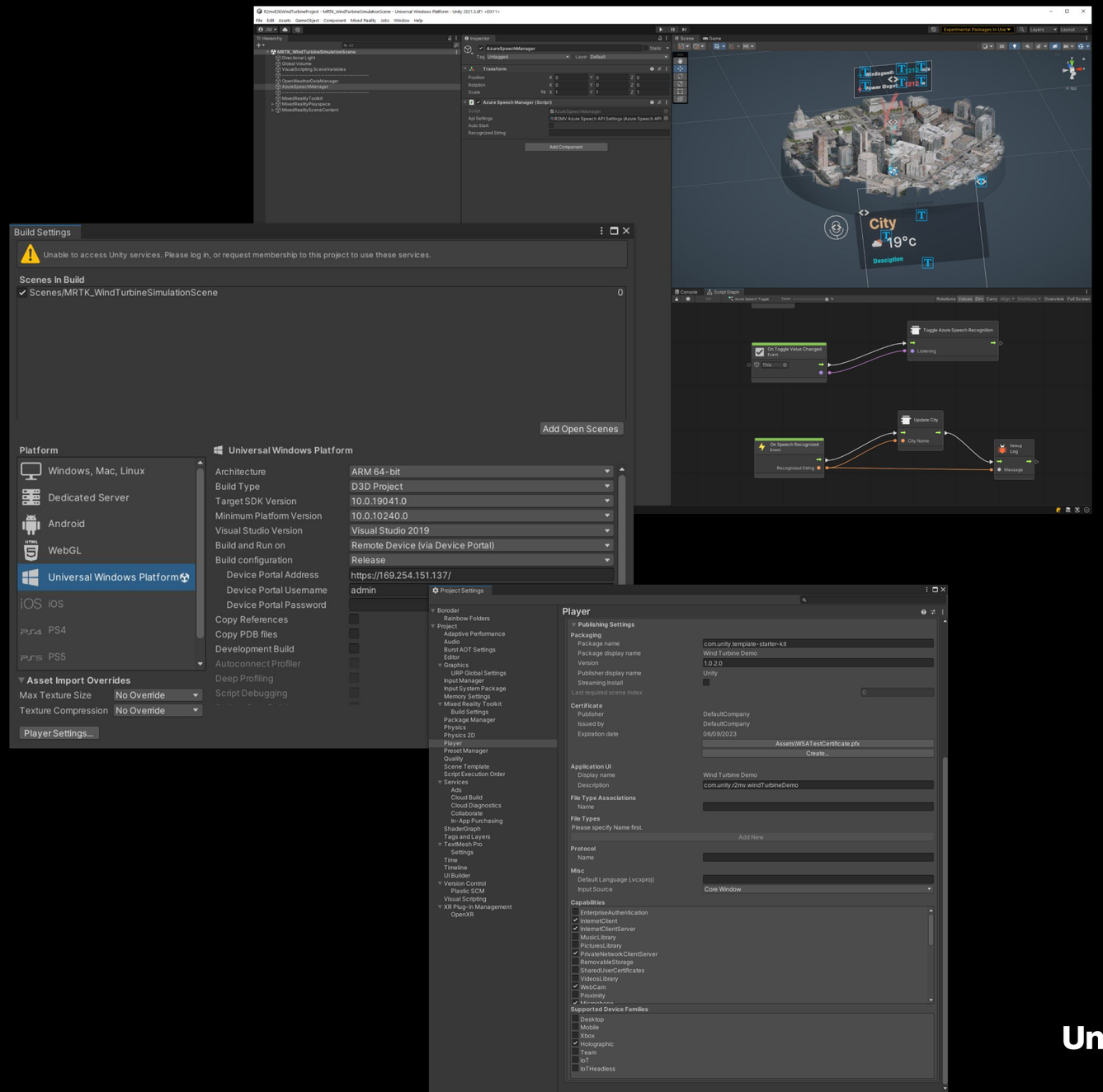
Project Setup

Build settings

- UWP
- ARM 64-bit
- Use **Visual Studio 2019**
  - Install build tools **v142**
- Build and run via **Device Portal**

Project settings

- Make sure to check the required capabilities under Player → Publish Settings
- Use **.NET Framework** for the latest features
- Make sure to install the **Mixed Reality OpenXR Plugin**!



Holographic Remoting

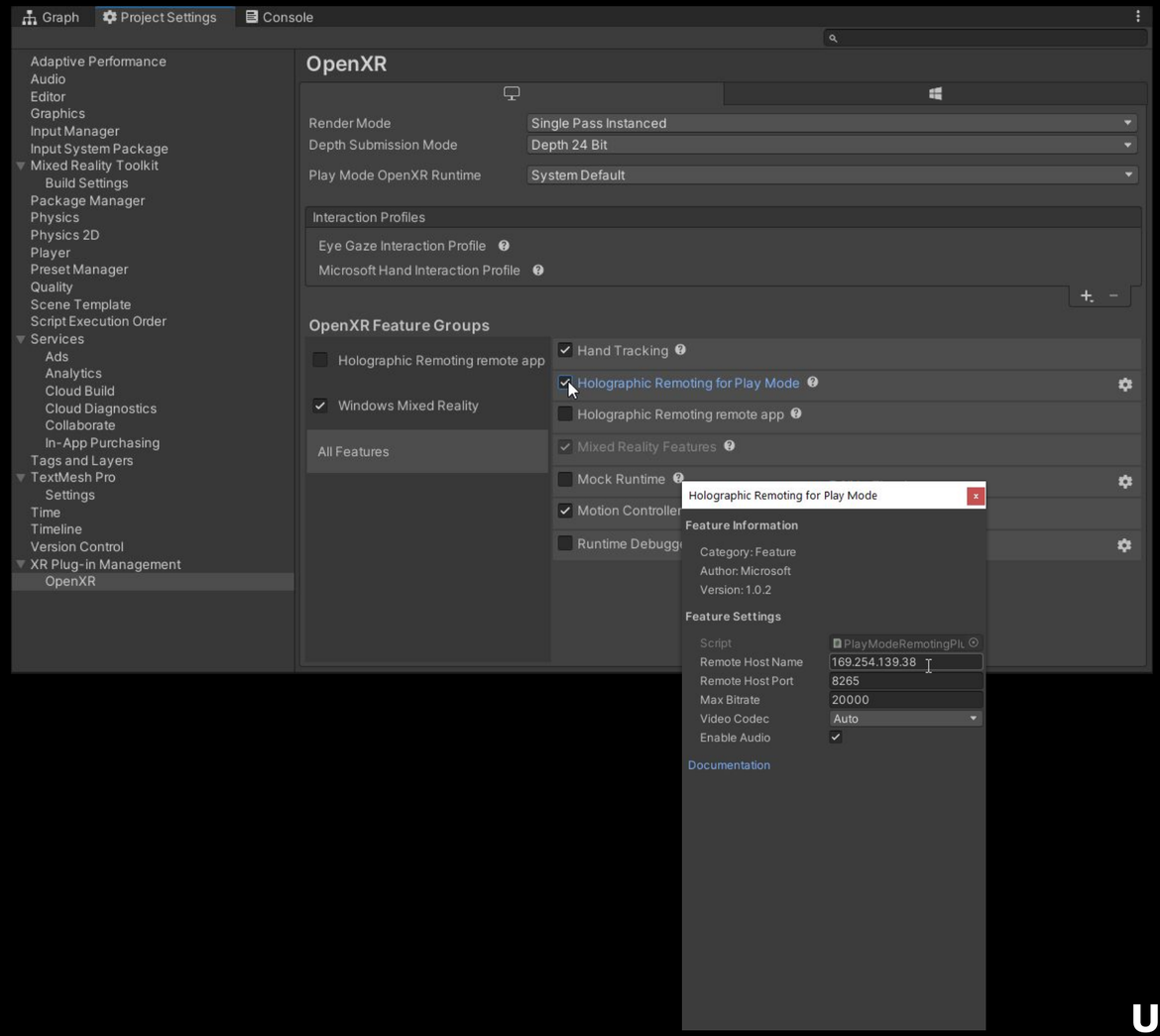
Stream holographic content to your HoloLens in real time, quickly test and debug your app, **without building** and **deploying** a full project**.**

1. **Download** the Holographic Remoting app from the Windows Store on your Hololens
2. **Enable** Holograhic Remoting for Play Mode in Project Setting → XR Plug-in Management → OpenXR
3. Click the ⚙️icon to reveal the **settings**

There are two options for connecting:

- Wifi for wireless freedom
- USB for speed and stability

🚨 When connected via a USB cable, make sure to disable Wifi on the HoloLens.



Docs

Let’s jump in!

Live demo ahead...

Resources used

Vertical Turbine with Solar Panels

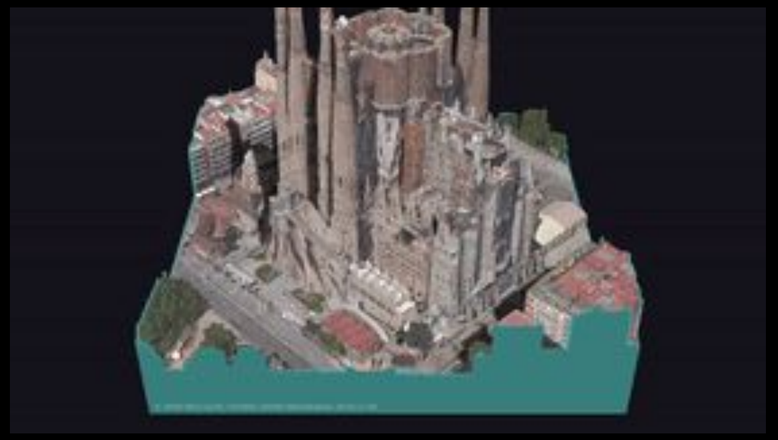
Vertical turbine concept by **Rifky Zaidani**. Created in **Solidworks**, imported with **Pixyz Plugin**.

GrabCAD

Azure Cognitive Services

Import and learn about Azure Speech Services to enhance your learning experience.

Tutorial



Maps SDK

A **Microsoft Garage** project provides the ability to visualize a 3D map within Unity-based projects.

Github

Questions?