The Unity Roadmap

As of November 19th, 2025
Latest version @ unity.com/roadmap

Safe Harbor Statement

The following is intended for informational purposes only, and may not be incorporated into any contract. No purchasing decisions should be made based on the following materials. Unity is not committing to deliver any functionality, features or code. The development, timing and release of all products, functionality and features are at the sole discretion of Unity, and are subject to change.

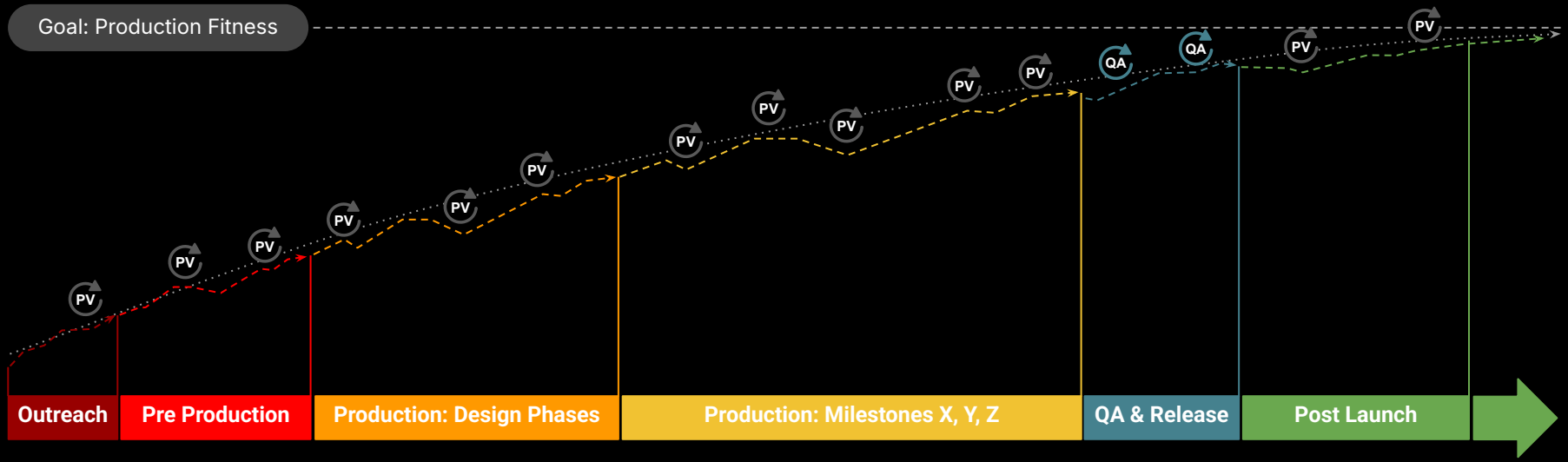
Developer Confidence

Production Verification (PV)

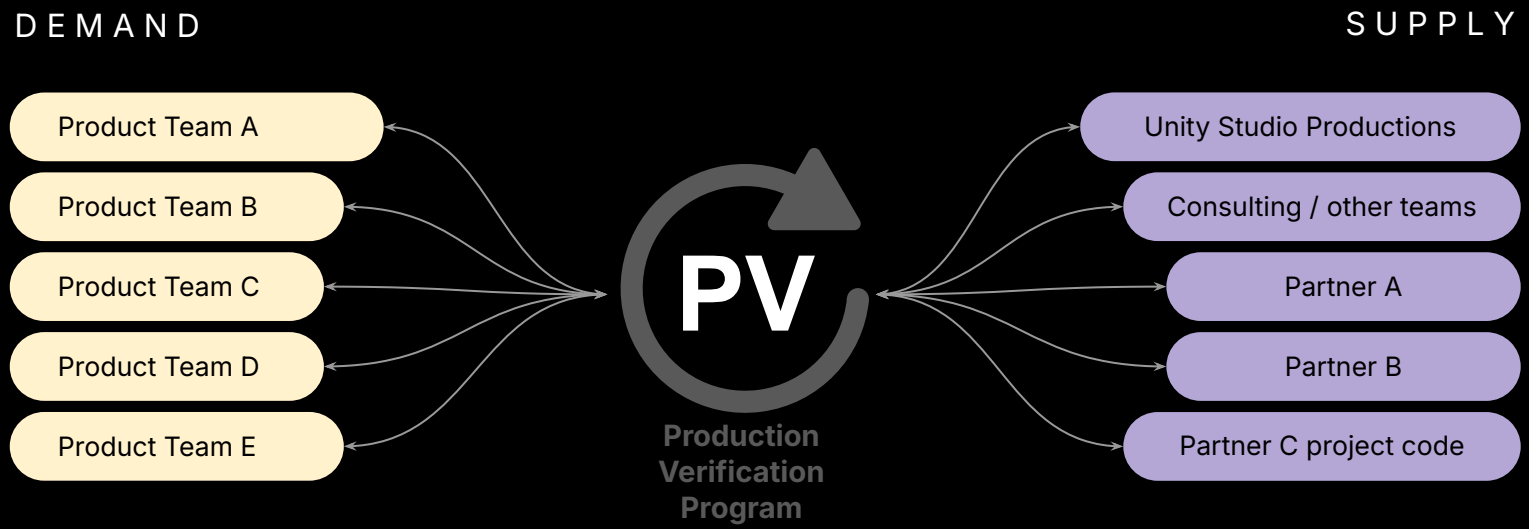
Unity 6.3 LTS production verified with:

- → **Survival Kids** by Unity and Konami
- → **VRising** by Stunlock Studios
- → **Phasmophobia** by Kinetic Games
- → **5 Supersonic** Mobile Titles
- → **Thrasher** by Puddle Games
- → **Den of Wolves** by 10 Chambers

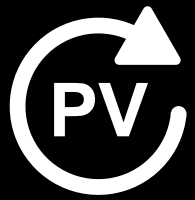
Production Verification (PV)



Production Verification (PV)

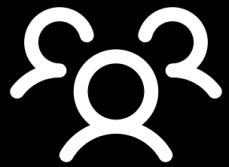


Production Verification (PV)



Production Verification

Unity Studio Productions
& Consulting



Partner Council

External partners
by invitation only

Code Sharing

External partners
by invitation only

Focus on Quality pays off

70%+

Internally reported issues

-43%

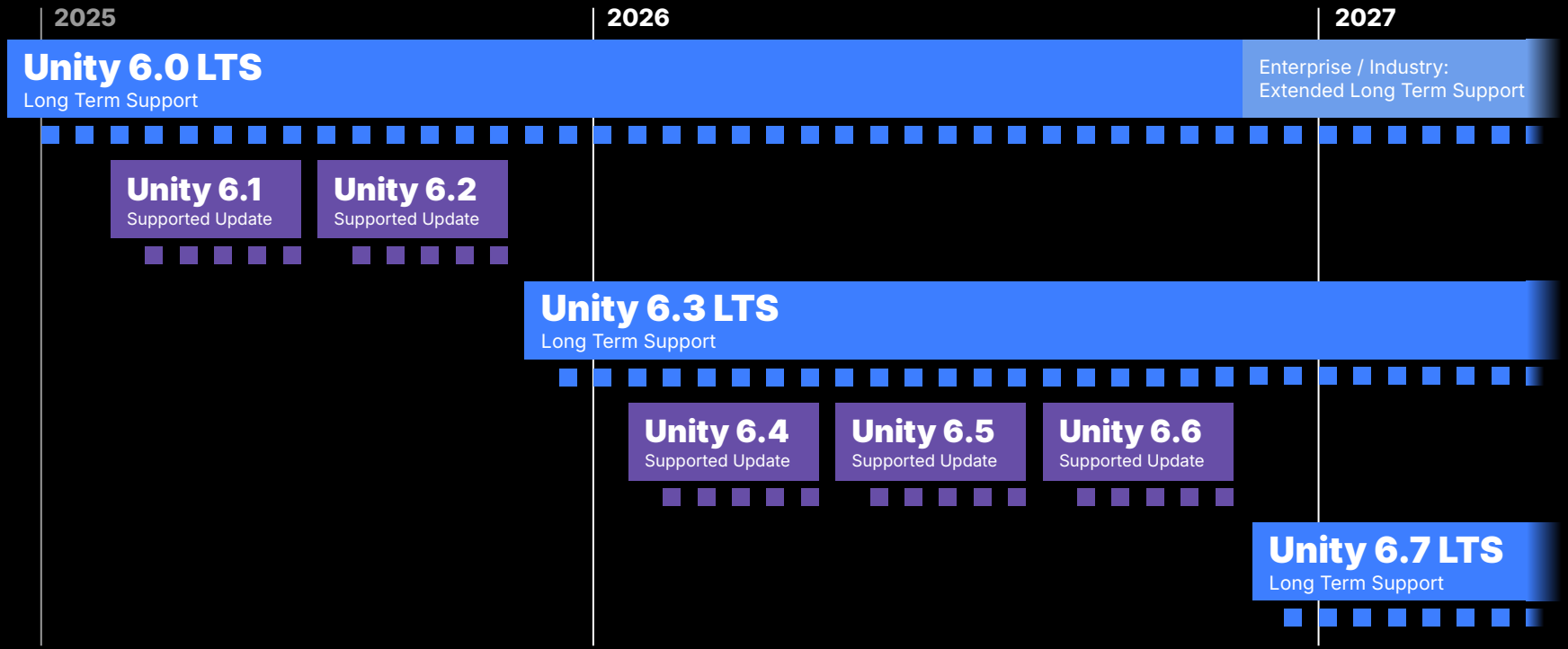
Median priority regression time to fix
(now around 8 days)

-54%

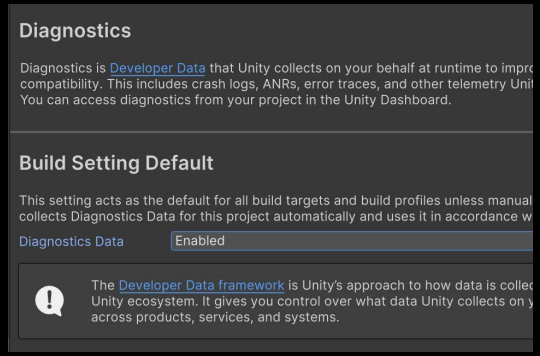
Regressions in active backlog

Based on Unity internal data from October 2023 to October 2025

Unity 6 quarterly releases



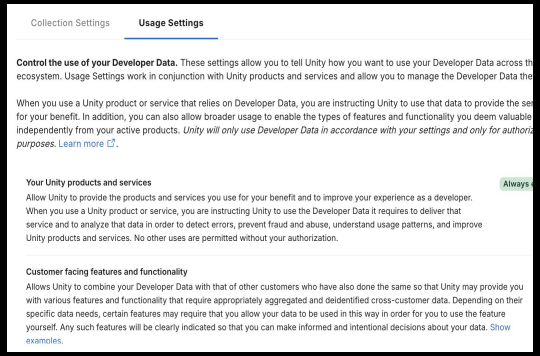
Developer Data Framework



Conservative defaults

Defaults ensure your data is used the way you and your players want.

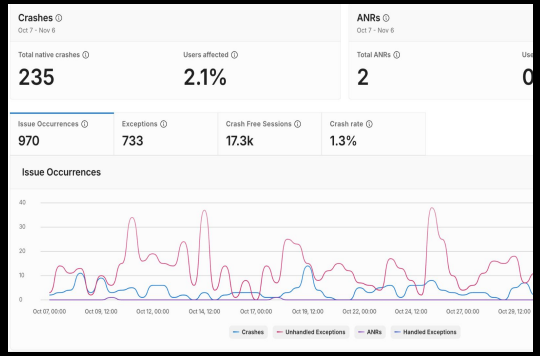
6.2



Flexible data collection controls

Centralized settings and remote controls that evolves with your game's strategy.

6.2



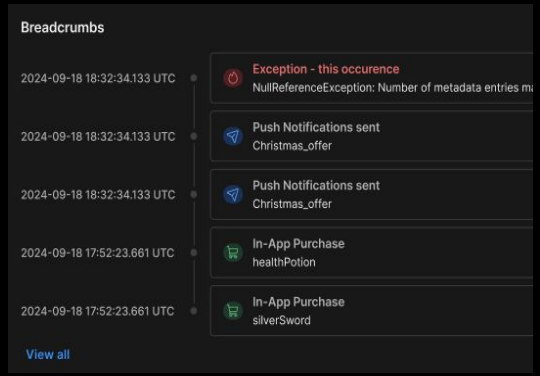
Help Unity's product quality

Engine Diagnostics enabled by default for monitoring stability and ANR* events.

**Application Non-Responsive*

6.2

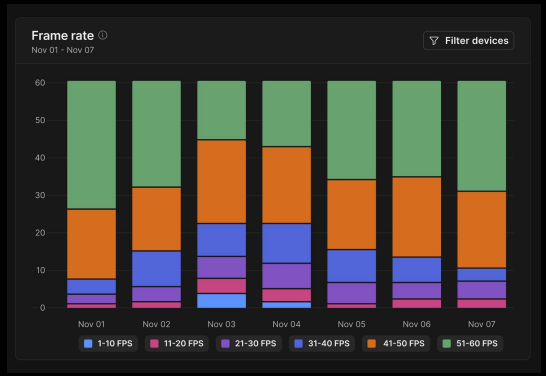
Engine Diagnostics



Accelerate issue resolution

Robust diagnostics: alerts, custom logs & breadcrumbs, aggregations, and more

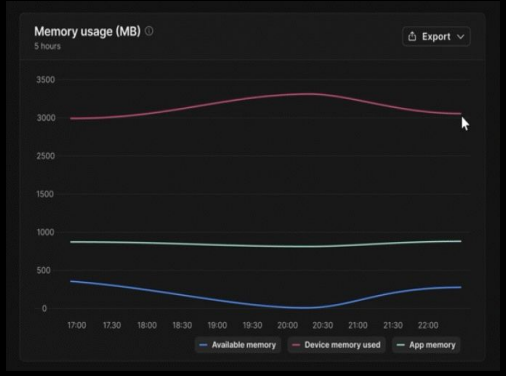
6.3 LTS



Sharper insights

Advanced filters, richer graphs, and actionable metrics.

6.4

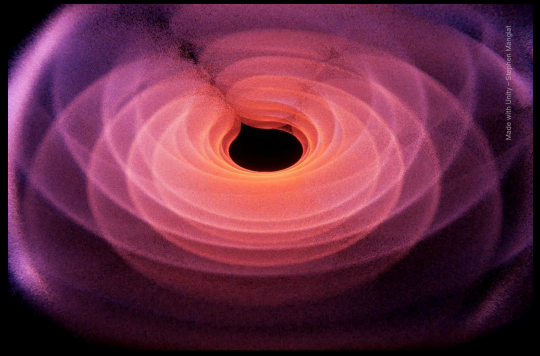


Diagnostics where you work

Ready for analysis in dashboards or your own tools via unified data export.

6.4

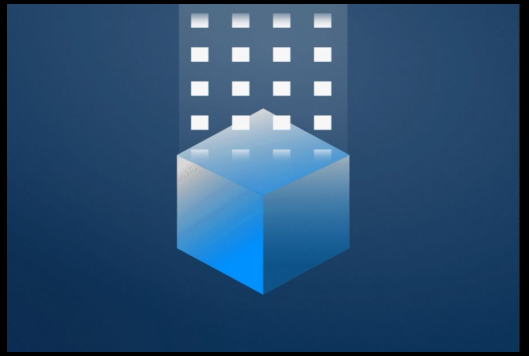
From Developer Data to Data Ecosystem



Supporting your live games

Diagnostics and Developer Data Framework backported in 2026.

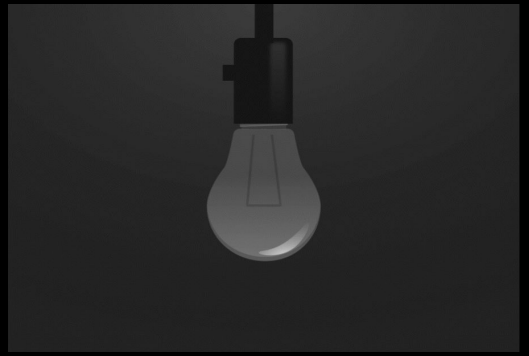
22 LTS | 6.0 LTS | 6.3 LTS



Unified data and pipeline

Unified object model and optimized data infrastructure to improve data quality.

6.4



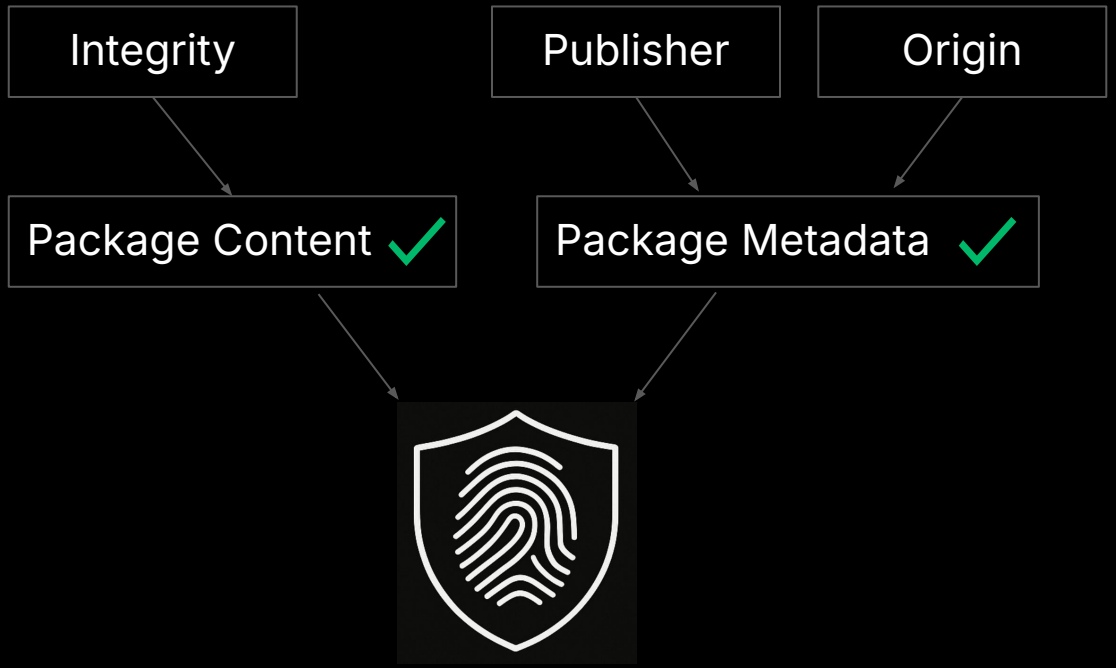
Turn signals to powerful insights

Expanding telemetry to 1st and 3rd party sources for richer insights.

6.4

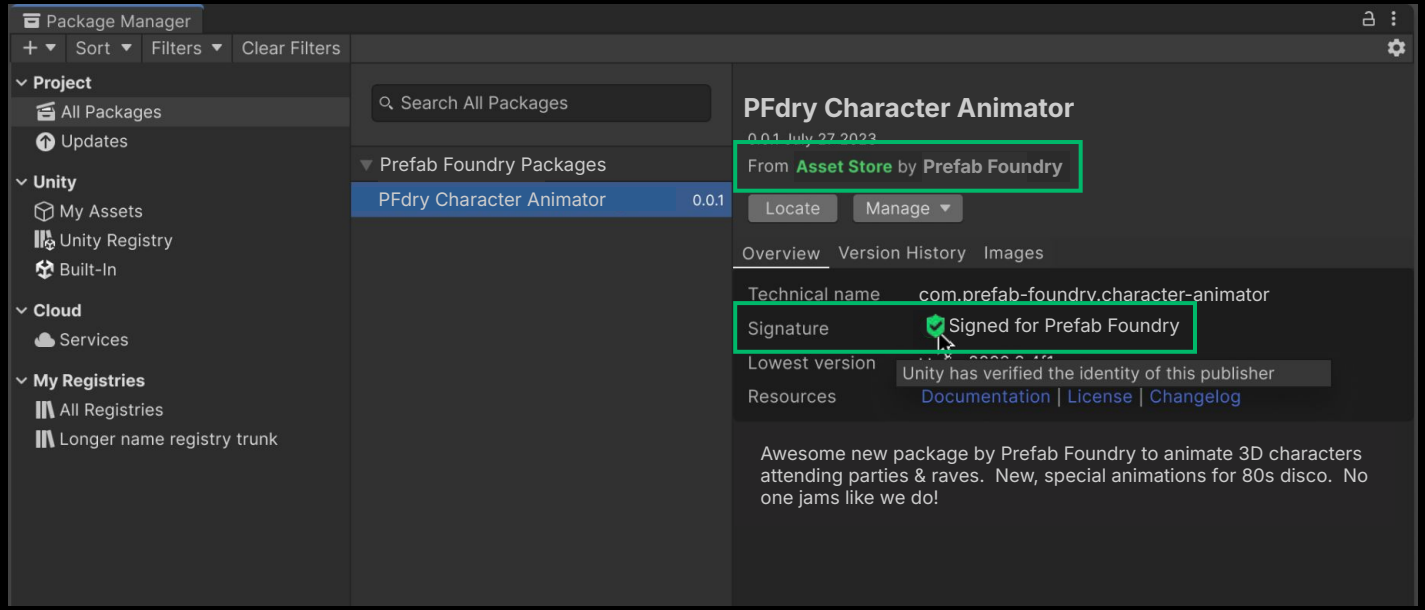
Unity Core Standards

Unity Core Standards



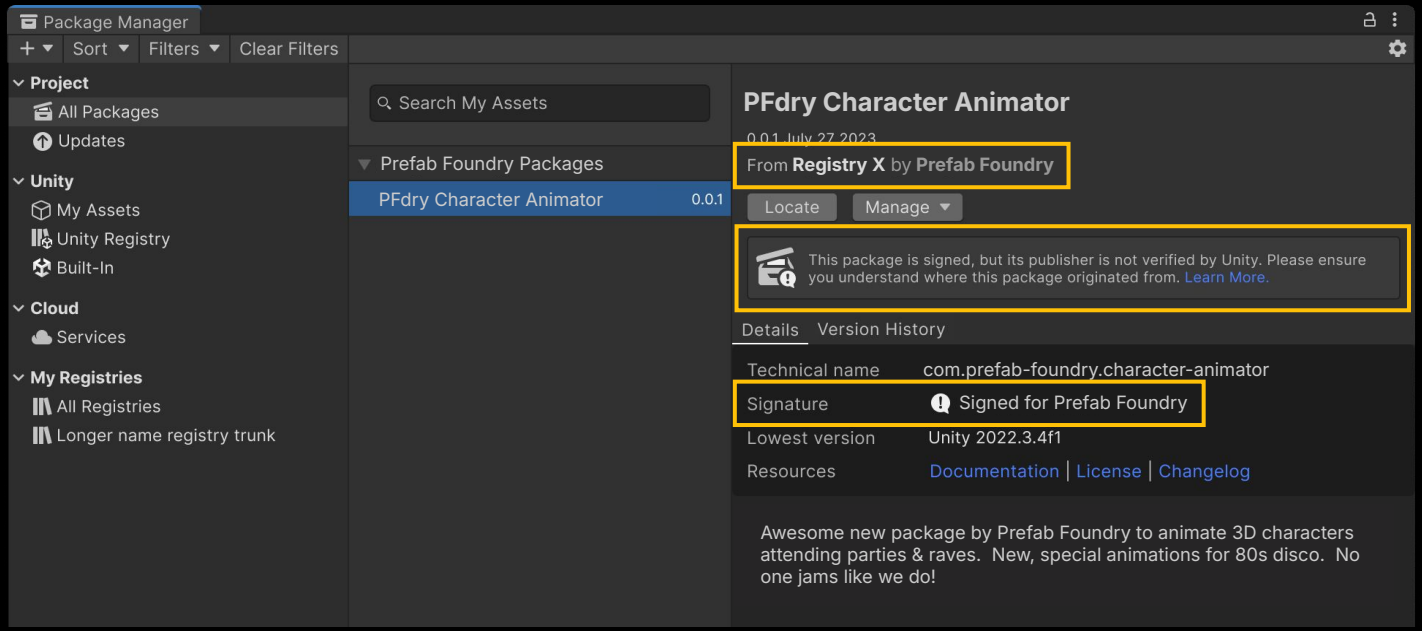
Unity Core Standards

Signed Controlled Distribution



Unity Core Standards

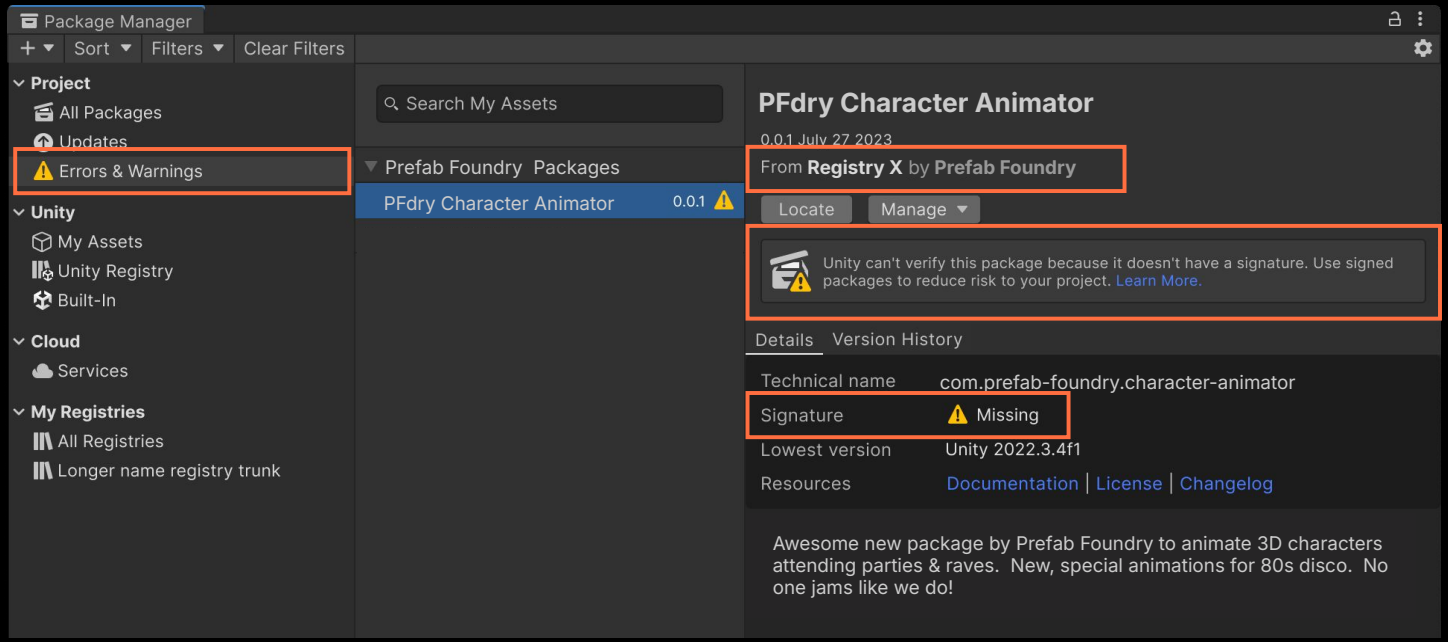
Signed and independently distributed



6.3 LTS

Unity Core Standards

Unsigned



6.3 LTS

Unity Core Standards

Unity Asset Store

BETA

Apply for
Early Access

Unity Made

Asset Store

Studio Internal

Signed Controlled Distribution

Smooth addition to the project. No questions asked.

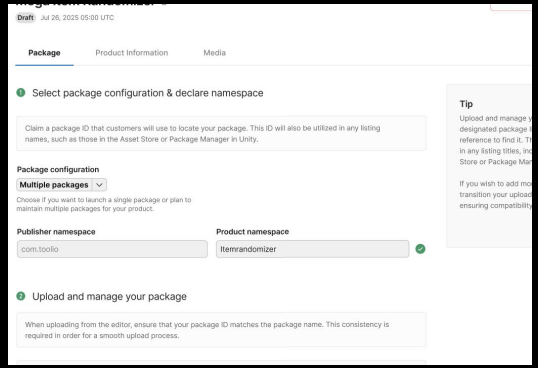
Open Source

Web Share

Signed and Independently distributed

Unity provides data points to guide project addition.

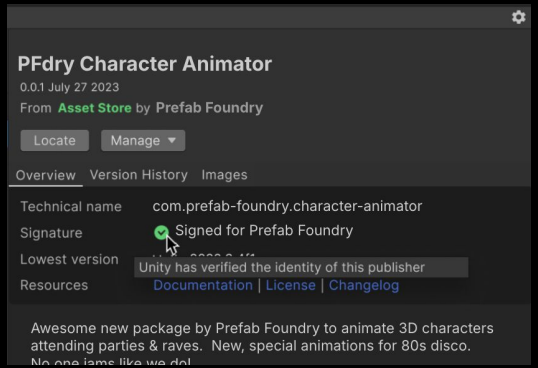
Package Manager Integration



Beta Package Authoring

Create multiple, modularized packages within a single product, with versioning

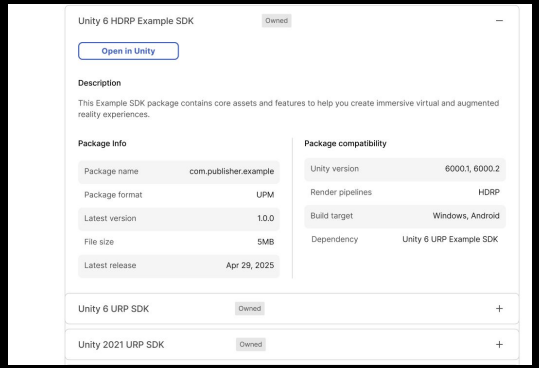
22 LTS | 6.0 LTS | 6.3 LTS



Package Signatures

Packages now have signatures to verify their origin, and that they hasn't been tampered with.

6.3 LTS



Streamlined Integration

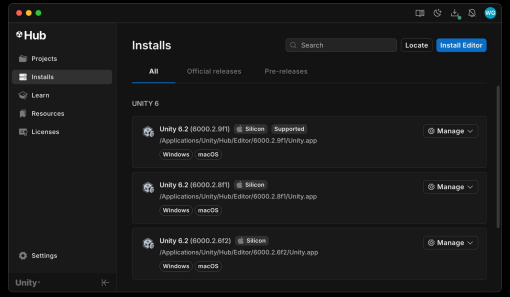
Packages state compatibility, provide version selection, acquire dependencies automatically, and exist outside of the Asset folder for reduced project size.

22 LTS | 6.0 LTS | 6.3 LTS

Develop
FAST
Deploy
EVERYWHERE
Grow
YOUR GAME

Develop Fast

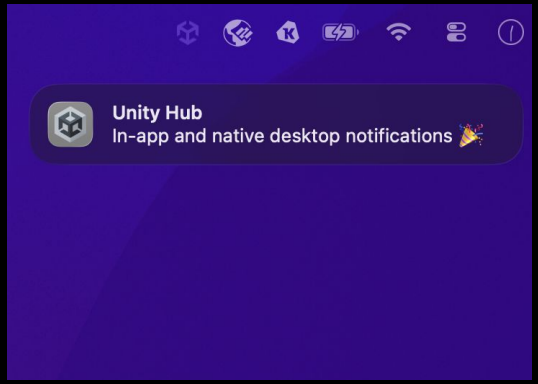
Unity Hub



Quality of Life

Improved Hub UX with UI refresh and greater customization.

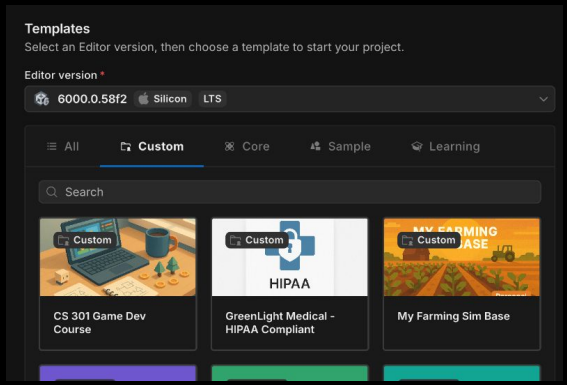
6.3 LTS



Notifications

In-app or on your desktop. Never miss an update.

6.3 LTS



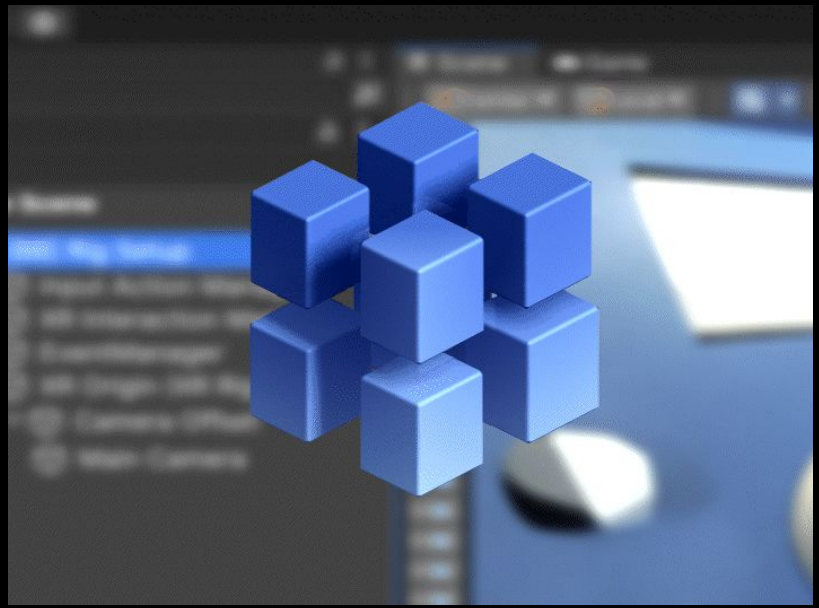
Custom Templates

Tailored to your needs. Launch projects pre-configured and ready.

6.4

Unity Building Blocks

- → Starting point to quickly add functional features to your project
- → Add to your project at any stage of development and customize
- → Available on the Asset Store with scripts and a test scene to help you get started



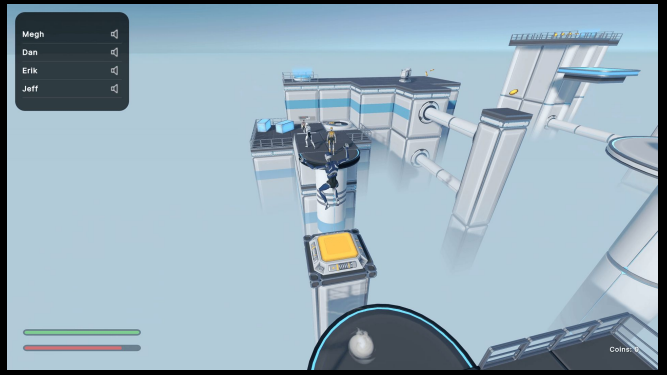
6.3 LTS

New 3D Multiplayer Templates

Multiplayer FPS

An extensible multiplayer FPS sample, networked by default. Bridges Netcode for Entities performance with GameObject systems.

6.3 LTS

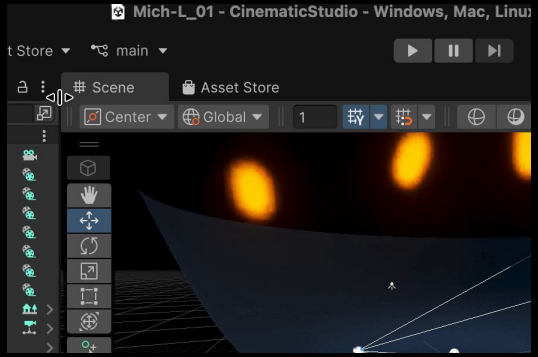


Multiplayer Third-person Gameplay

A multiplayer prototyping kit that uses Netcode for GameObjects, features Unity Building Blocks content for quick iteration.

6.3 LTS

Unity Version Control



Editor Toolbar Redefined

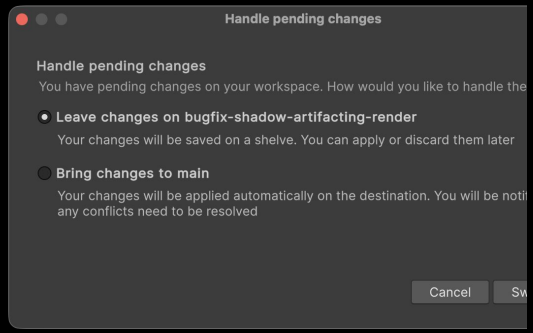
- → View project status
- → Track and manage changes
- → Perform key actions with a single click

6.3 LTS

Improved Hierarchy View

- → Now includes Unity Version Control actions and status icons
- → Prevent changes on locked scene

6.3 LTS

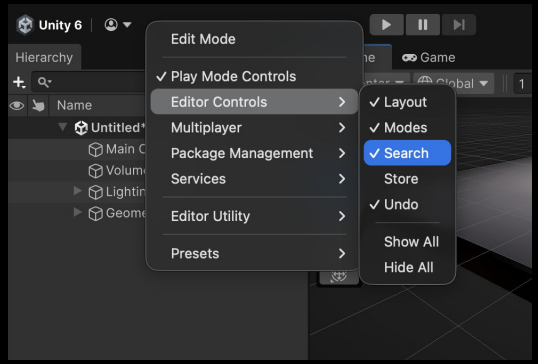


Streamlined Workflows

- → Switch branches without losing work
- → Use labels to tag and compare a project's versions

6.3 LTS

Unity Editor

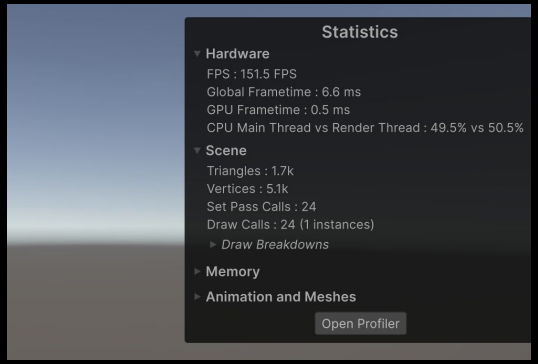


Customizable Main Toolbar

→ Show, hide and rearrange toolbars

→ Extensible API to add new functions

6.3 LTS



New Editor Stats Window

→ Identify bottlenecks, prioritize fixes

→ More accurate and detailed metrics

→ Quick access to the Profiler

6.4

Grid and Snapping Workflow

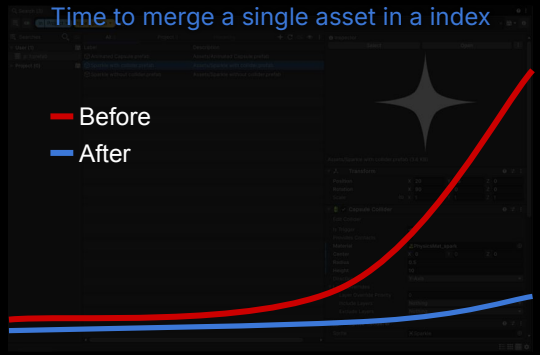
→ Faster access to settings

→ Clearer, consolidated controls

→ Fully customizable grids

6.4

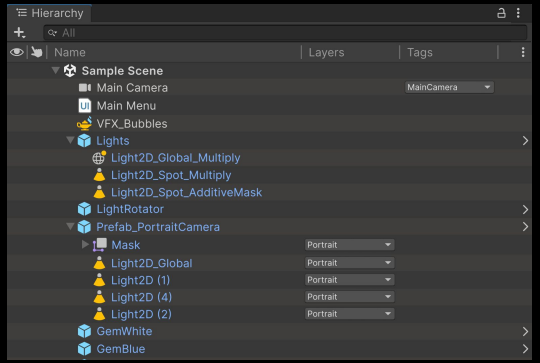
Unity Editor



Search Performance, Simplified

- → New backend for faster and more reliable search
- → Better scalability for large projects

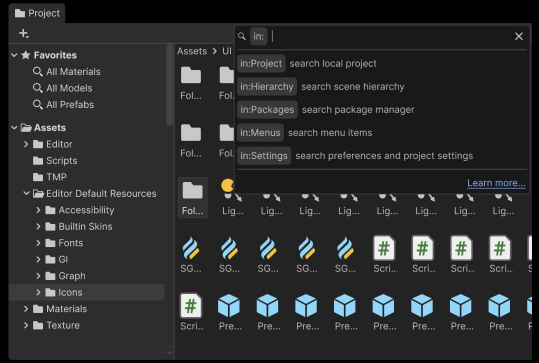
6.3 LTS



Faster, More Flexible Hierarchy

- → Efficiently handles millions of objects
- → Visual indicators for GameObject
- → Customizable columns

6.6



Unified Search Experience

- → Intuitive and simple to use
- → Accessible across the Editor

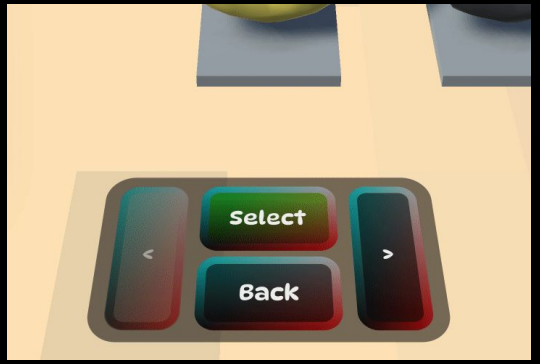
6.7 LTS

UI Toolkit - World space and advanced visuals

World Space UI

- → Place UI directly in 3D scenes
- → Crisp visuals at any scale
- → Fully compatible with XR

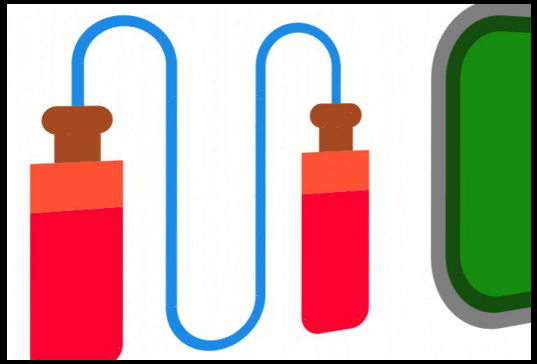
6.2



Custom Shaders & Filters

- → Customize UI using ShaderGraph
- → Apply predefined or custom post-processing effects

6.3 LTS



Vector Graphics

- → Import SVG files into the project
- → Resolution-independent visuals
- → Antialiased vector rendering

6.3 LTS

UI Toolkit - Additional improvements

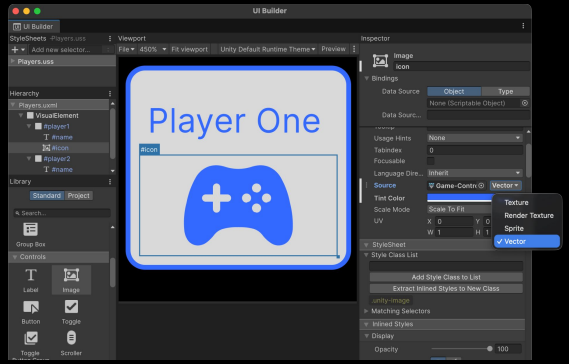
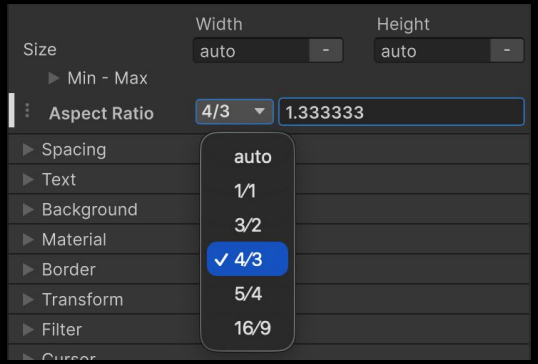


Image Control

→ Improved control over resizing and scaling

→ Now accessible in the UI Builder

6.3 LTS



Aspect Ratio Property

→ Consistent width-to-height proportions

→ Behaves like CSS aspect-ratio

6.3 LTS

New public APIs

→ Pseudo states

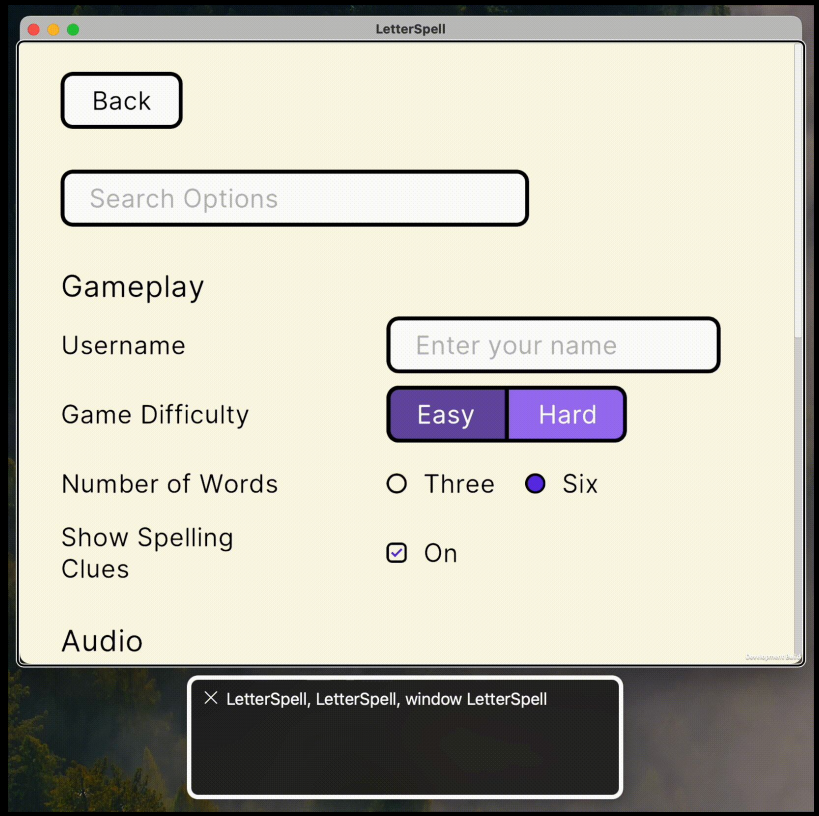
→ Control base classes and interfaces

→ Text navigation and manipulation

6.3 LTS

Accessibility

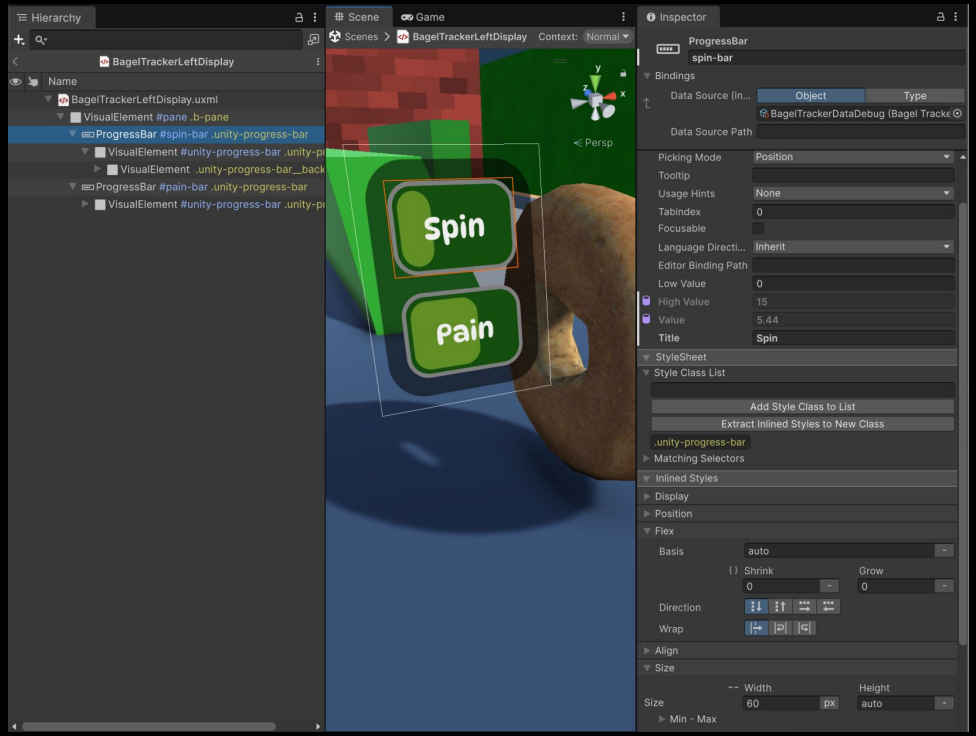
- → Windows and macOS screen reader
- → New accessibility roles
- → Mobile and desktop scrolling
- → Improved API & documentation



6.3 LTS

UI Toolkit - What's next

- → Complex animation
- → Custom shaders improvements
- → In-scene authoring
- → Performance profiling tools



6.7 LTS

Unity UI (uGUI)

→ uGUI remains critical feature

→ Fully supported in Unity 6 series

→ Production Verification unlocking :
- Quality of life improvements
- Performance optimizations

→ TextMesh Pro also supported in Unity 6
- Quality of life improvements
- Performance optimizations

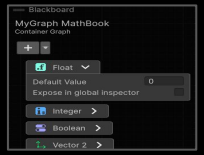
6.7 LTS

Graph Toolkit

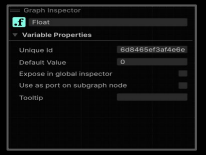
→ Becomes a core module

→ What's next:

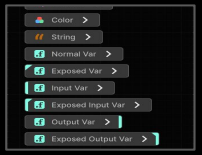
- Vertical Ports
- API for Graph Creation
- Node Customization



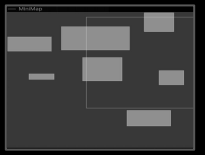
Blackboard



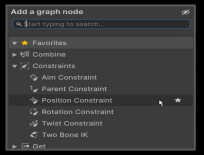
Local Inspector



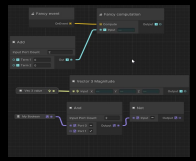
Variables



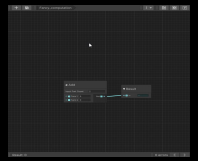
Minimap

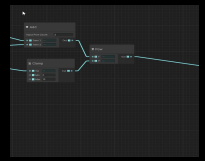


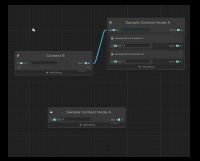
Item Library

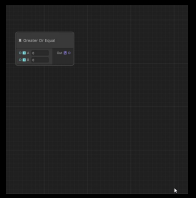

Nodes

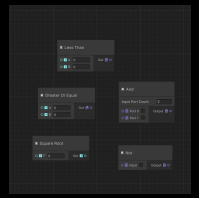
Ports & Wires

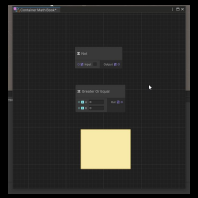

Toolbars

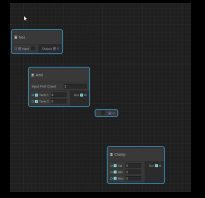

Subgraphs

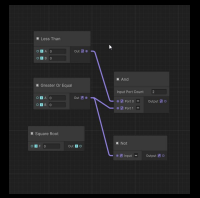

Contexts & Blocks

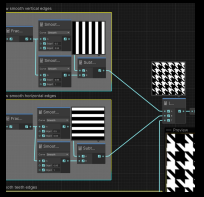

Basic commands

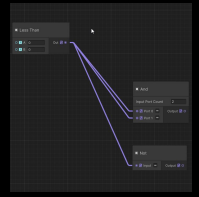

Selection

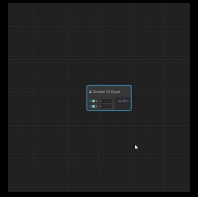

Auto Pan

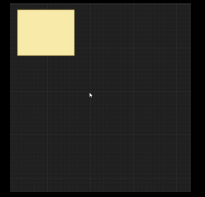

Placement Options

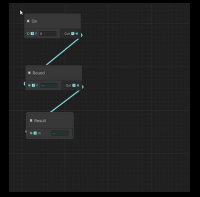

Deletion


Previews

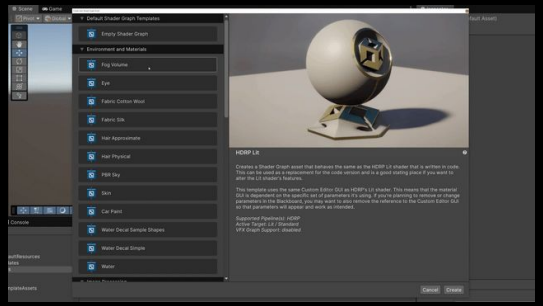

Portals


Automatic LODs


Sticky Notes


Placemats

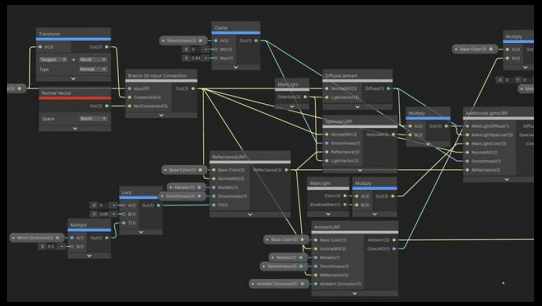
Shader & VFX authoring improvements



Graph templates

→ Start from what you want to achieve
→ Make your own team templates

6.3 LTS



UI Toolkit, Terrain, Custom Lit

→ Author without code in Shader Graph

6.3 LTS

Top requested functionalities

→ Nested properties, improved inspector customization, dynamic keywords, 8 Tex. Coords support,... + Stencil settings (6.5)
→ VFX GPU events instancing
→ New samples & learning content

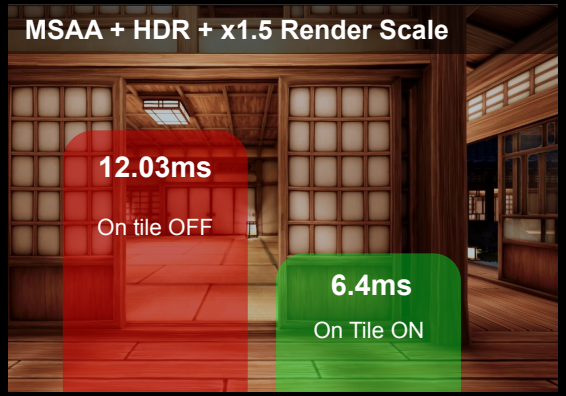
6.3 LTS

Graphics performance

Improved batching

→ Renderer Shader User Value (RSUV) for SRP Batcher

6.3 LTS



XR post process (URP)

→ XR tile-based Optimizations

6.3 LTS

Improved culling

→ Small objects shadows GPU occlusion culling

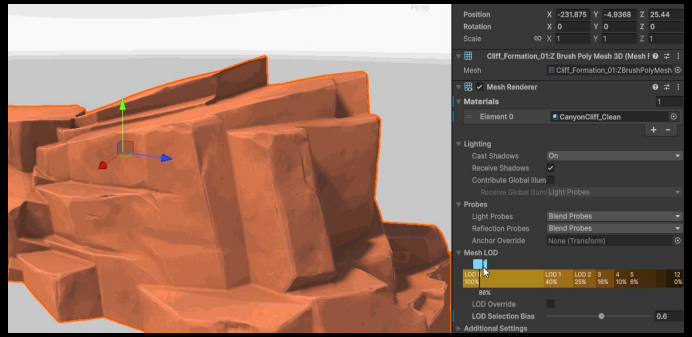
6.6

Level of Detail (LOD)

Mesh and Skinned Mesh LOD

- → Optimized memory
- → In-Editor generation
- → Mesh LOD API
- → Scene labels

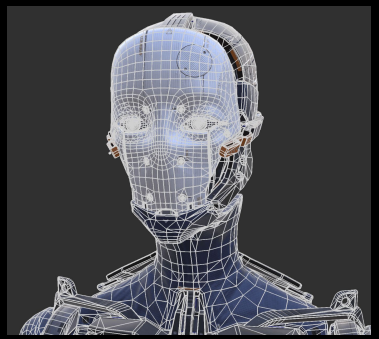
6.3 LTS



Mesh LOD Debug

- → Inspector sliders

6.4

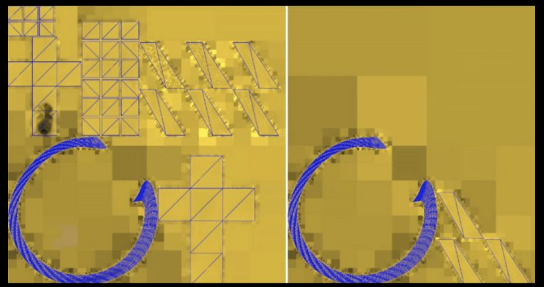


Expand LOD generation

- → Control over decimation
- → More LOD generators
- → Asset Transformer SDK (‘Pixyz’)

6.7LTS

Improved lighting quality, performance and workflow



Ease of use and performance

- → Lightmap UV packing with xAtlas
- → Low-level unified Ray Tracing API
- → Improved mobile bloom performance

6.3 LTS

High quality lighting in URP

- → Screen Space Reflections

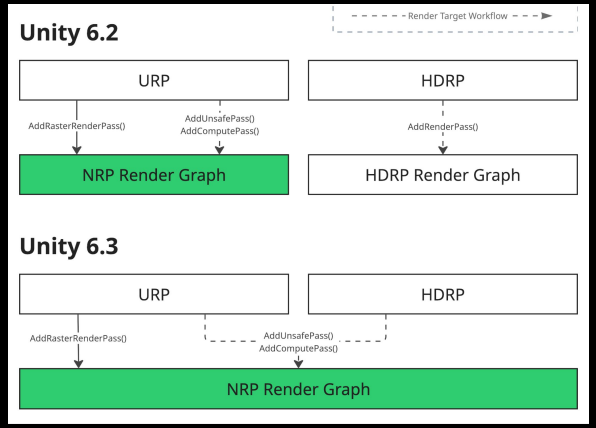
6.7 LTS

Fully dynamic diffuse GI

- → Faster authoring with Surface Cache GI
- → Dynamic: destruction, procedural content...

6.7 LTS

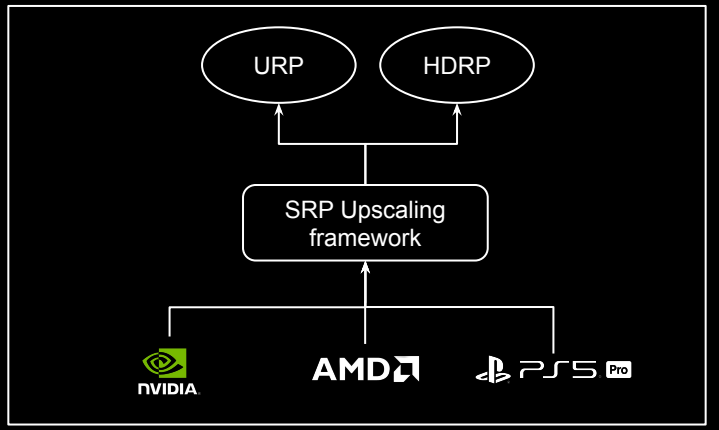
Unification and Extensibility



Render Graph for all!

→ Single backend for all SRPs
→ Easier and optimized customization

6.3 LTS



Cross pipeline upscaling framework

→ Plug-in your own upscaler
→ NVIDIA DLSS 4 Super Resolution
→ AMD FidelityFX(TM) Super Resolution 3 Technology
→ PlayStation® Spectral Super Resolution (PSSR)

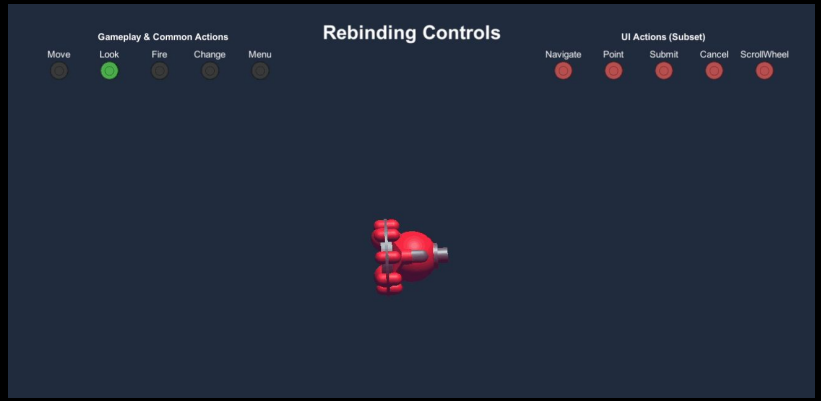
6.5

Input System rebinding sample

End User Rebinding

- → Result of a key feature request
- → UI for any supported device
- → Observability of input actions

6.3 LTS



Gameplay Example

- → Actual game with in-game actions
- → Observability of input actions
- → Showing dynamic binding labels

6.3 LTS

Physics

Swappable physics backends

→ Switching physics engines becomes possible

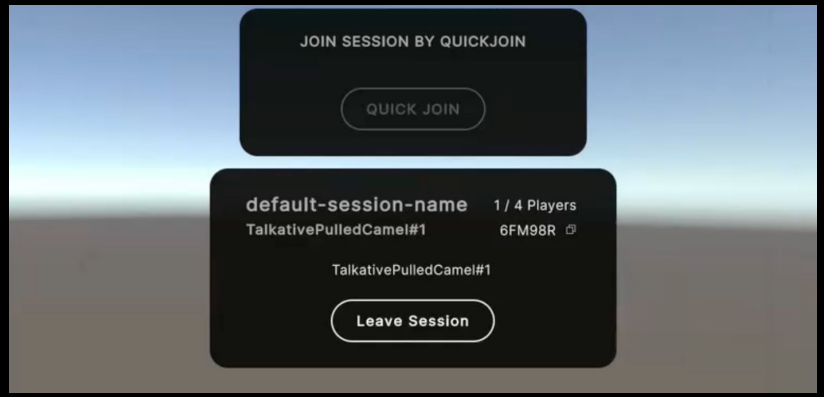
6.4

Unity Physics improvements

→ Direct solvers available in Unity
→ More versatility for complex interactions

6.4

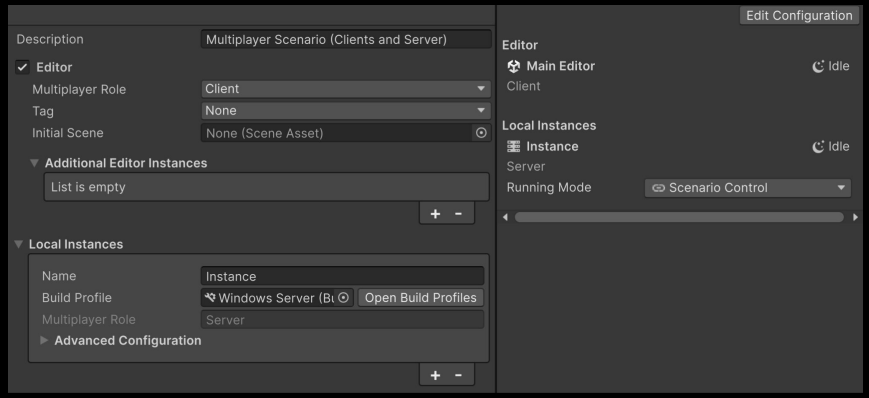
Get started faster with multiplayer



Multiplayer Building Blocks

→ Go fast from concept to playable

6.3 LTS

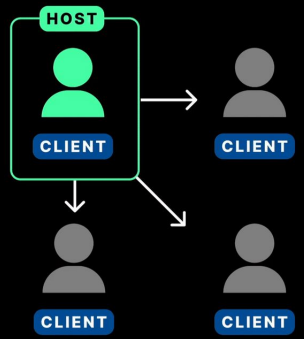


Further with Multiplayer Play Mode

→ Test quickly with multiple players and host directly in Editor
→ Independent instances provide flexibility and faster iteration

6.3 LTS

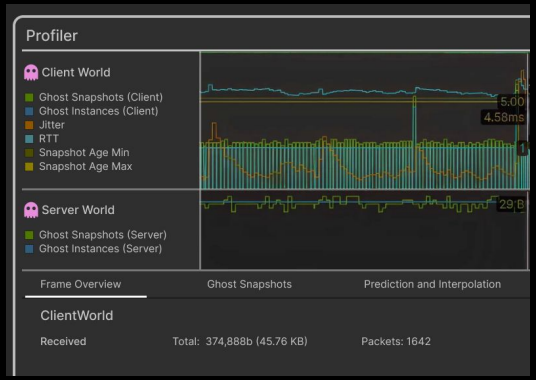
Host migration and Netcode updates



Host Migration

→ Handle disconnections with client host migration for Netcode for Entities

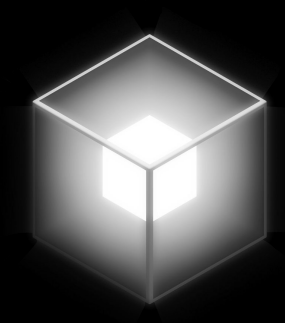
6.3 LTS



Troubleshooting Tools

→ Debug network traffic with the Network Profiler for Netcode for Entities

6.3 LTS

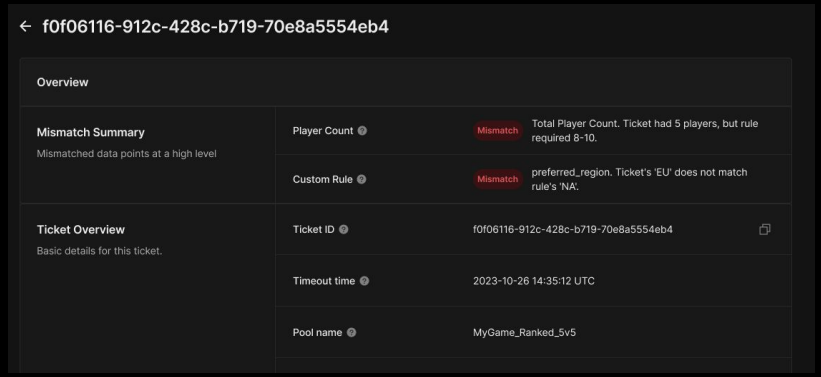


ECS for all

→ Bring features of Netcode for Entities to all GameObject projects

6.7 LTS

Flexible insight-driven Matchmaking



Get insights and troubleshoot

→ Greater insights into matchmaking issues

6.4

Backfill your teams with precision

→ Conditional backfilling for custom needs

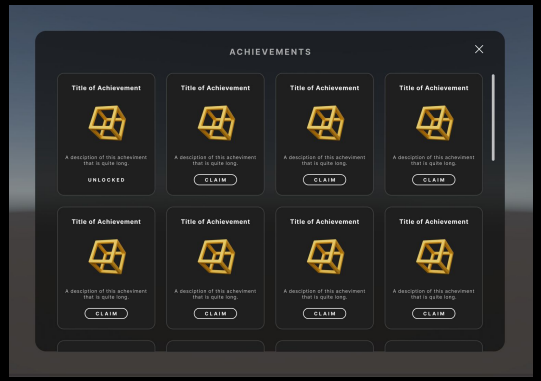
6.4

Cloud-code optimized for turn-based multiplayer

- → Event-driven for Card, Puzzle, and Idle games
- → Bridge Unity gameplay code with cloud services
- → Ready-to-use and cost-effective solution

6.7 LTS

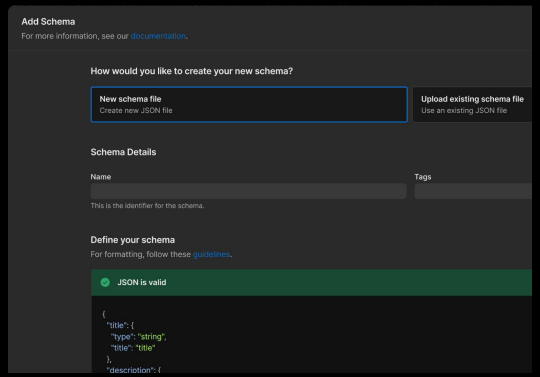
Game backend



LiveOps Building Blocks

→ Off-the-shelf LiveOps and Multiplayer features

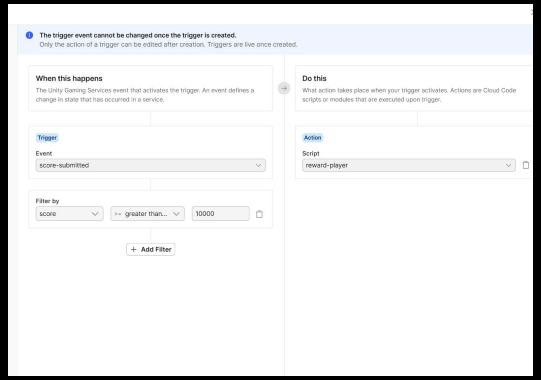
6.3 LTS



Structured game data

→ Standardized LiveOps data with Schema-driven validation

6.5



Setup game-driven workflows

→ Triggers and Secret Manager opens up extensions to 3rd party integration

6.5

2D Graphics

Rendering 3D as 2D

- → Mesh / Skinned Mesh Renderers in 2D URP
- → Mesh renderers lit by 2D Lights, interact with Sprite Masks
- → Sorted like Sprites

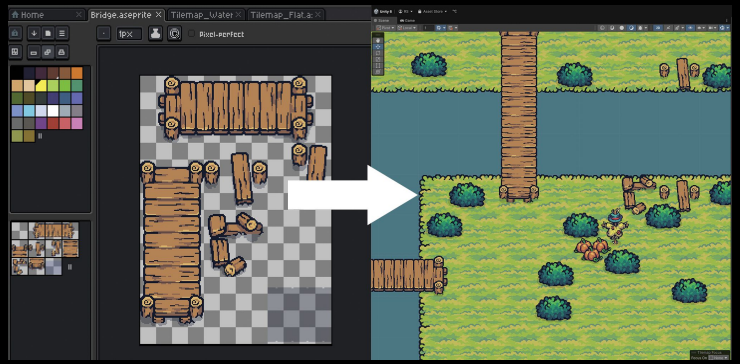
6.3 LTS

2D URP Extensibility

- → APIs and injection points enable custom rendering
- → New graphical objects like custom 2D lights & pipelines
- → Improves project upgradability and scalability

6.7 LTS

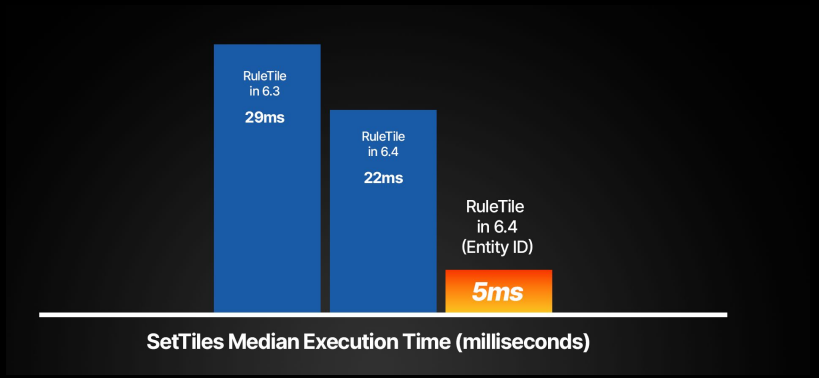
Aseprite importer & Tilemaps



Aseprite Importer - Tilemap Support

→ Aseprite ‘Tile Set’ Import Mode
→ Generates sprites, sheet, tiles, and a Tile Palette

6.1

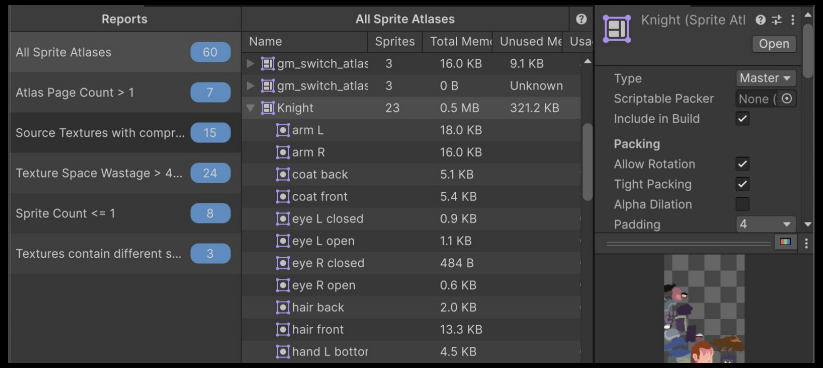


Tilemap Optimization with Entity IDs

→ New, opt-in tile system for Tilemaps
→ Solves performance bottleneck caused by managed objects
→ Compatible with Jobs for parallel, high-performance updates

6.4

Sprite Atlas



Sprite Atlas Analyzer

→ Find common atlasing issues: compressed source textures, multi-page atlases and underutilized texture space

6.3 LTS

```
            objectData.asset = inputSprites[atlasFit.index].GetEntityId();
            objectData.packInfo = new Vector4(atlasFit.position.x, atlasFit.positi
            atlasPage.assets[i] = objectData;
        }
        var textureData = new TextureData[1];
        Scale(packedTexture, (int)((float)packedTexture.width * multiplier), (int)
        textureData[0].texture = packedTexture.GetEntityId();
        textureData[0].mapName = "_MainTex";
        atlasPage.packedTextures = textureData;
        atlasPacked[pageIndex++] = atlasPage;
    }

    var config = new SpriteAtlasRuntimeConfig();
    config.scaleMultiplier = multiplier;
    SpriteAtlasManager.CreateSpriteAtlas(nameTag, config, atlasPacked);
}
```

Sprite Atlas Runtime API

→ Enable runtime Sprite Atlas creation and management
→ Optimized for dynamic content

6.4

2D Bone animation performance

- → Multi-threaded
- → Faster Shadows
- → Bone data uses less memory
- → Cached deformed sprites
- → Deformation calculations only on move

Coming next

- → Faster AABB (Axis-Aligned Bounding Box)

6.3 LTS

2D Low-level physics APIs

- → Highly parallelized up to 64 cores
- → Independent and parallel 2D physics worlds
- → Compatible interoptibility with GameObjects
- → Supports XY, XZ & ZY 2D planes
- → Powerful query system
- → Opt-in API, doesn’t affect existing projects

6.3 LTS

6.7 LTS

Compiler optimization

Burst Determinism

→ Floating Point Determinism across CPU architectures
→ Galaxy Game sample demonstrates ECS determinism

6.3 LTS

```
agnostic push
agnostic ignored "-Winvalid-offsetof"
agnostic ignored "-Wunused-variable"


IL2CPP_METHOD_ATTR LinearColor_t6096 LightmapperUtils_ExtractIndirect_m5776 (Light_t1E684* ___0_1
method)


Light_t1E68* L_0 = ___0_1;
NullCheck(L_0);
Color_tD001 L_1;
L_1 = Light_get_color_mE7EB(L_0, NULL);
Light_t1E6847* L_2 = ___0_1;
NullCheck(L_2);
float L_3;
L_3 = Light_get_intensity_m8FA2(L_2, NULL);
Light_t1E6847* L_4 = ___0_1;
NullCheck(L_4);
float L_5;
L_5 = Light_get_bounceIntensity_m5350(L_4, NULL);
LinearColor_t6096 L_6;
L_6 = LinearColor_Convert_m0E220(L_1, ((float)il2cpp_codegen_multiply(L_3, L_5)), NULL);
return L_6;


N_C IL2CPP_METHOD_ATTR float LightmapperUtils_ExtractInnerCone_m8B2B(Light_t1E68*    0_1, const Runtim
```

IL2CPP Improvements

→ Build Size Improvements targeting mobile / web
→ Runtime Performance Improvements

6.7 LTS

Faster, incremental architecture updates

.NET

Internal Milestone

- → Unity 6 proof of concept

6.4

Internal Milestone

- → Support for CoreCLR GC
- → Internal testing
- → Burst as a module

6.5

Validation

- → Production Verification with internal productions and selected customers

6.6

Experimental

- → CoreCLR Desktop Player Technical Preview

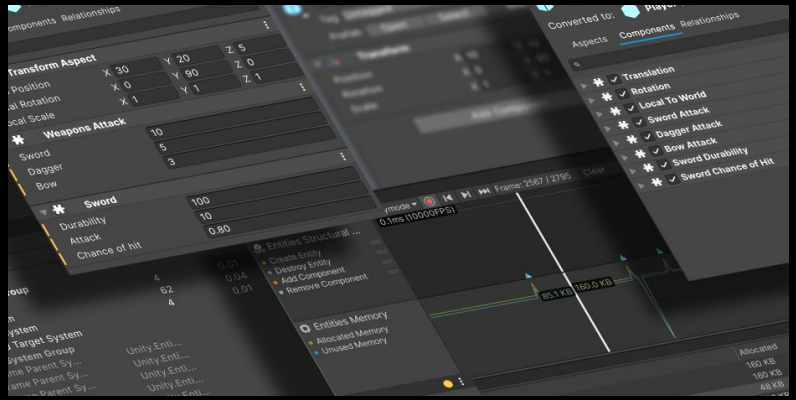
6.7 LTS

.NET Modernization

- → CoreCLR Desktop Player release
- → CoreCLR Editor
- → Updated IL2CPP

6.x

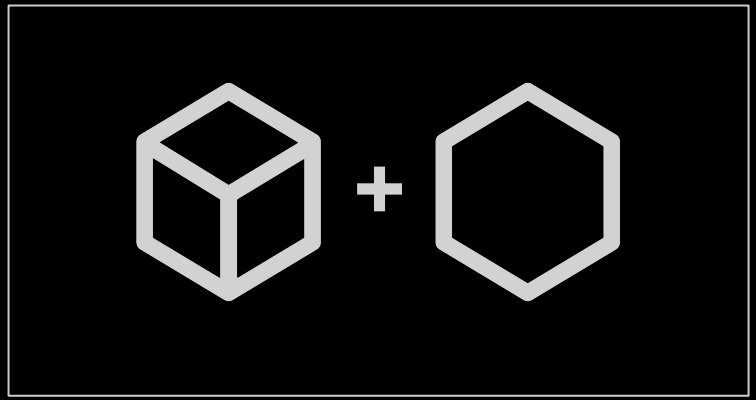
Faster, incremental architecture updates



ECS for all in Unity 6

→ Entities as a core package
→ Faster release cycle

6.4



Unified Runtime

→ ECS can access/modify GameObject Transforms
→ Common IDs between Entities and GameObjects

6.6

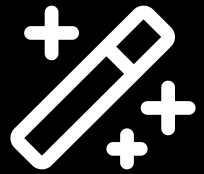
Agentic Development with Unity AI

Infrastructure

A stronger back-end for thorough, performant answers and code

Context

More understanding for increased relevance in complex scenarios



Workflows

Embedded agents & generations meet you where you are

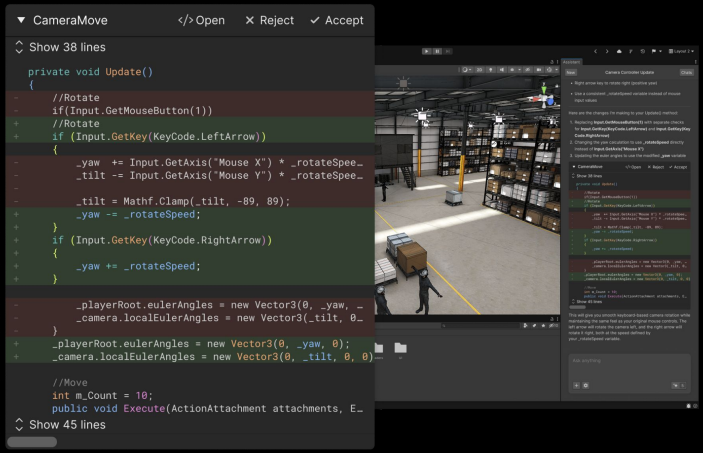
In Beta

Agentic Infrastructure

Orchestration Handles Complex Prompts

- → Improved planning and reasoning
- → Multi-step resolution
- → Routing to tools and agents

6.4

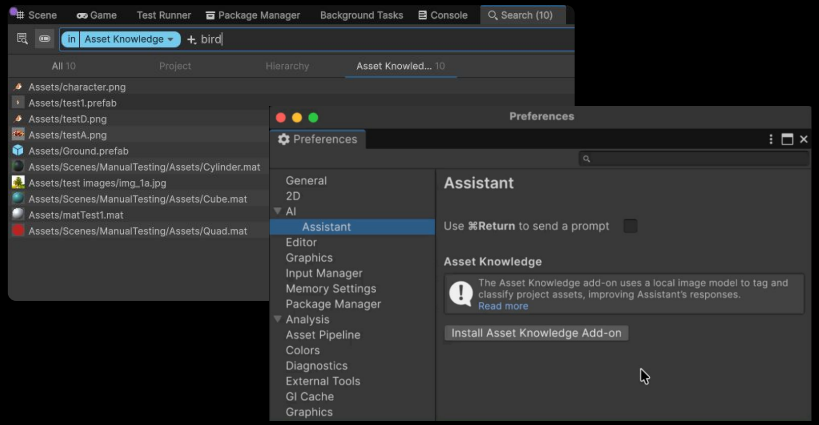


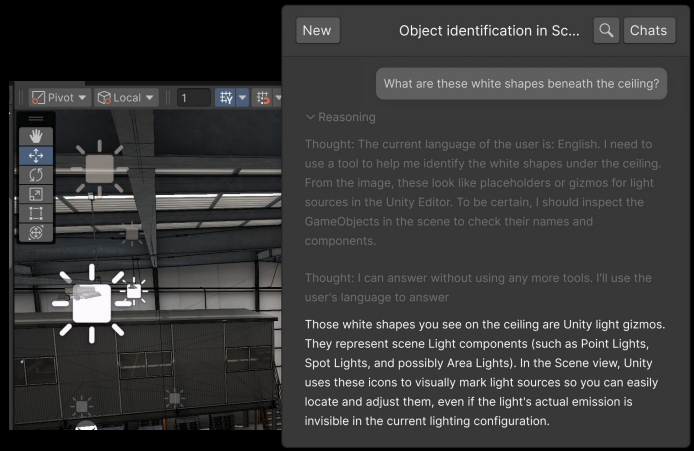
Production-Safe Code, Transparency, Full Control

- → Dependency-aware code changes
- → git integration
- → Visual diff tool

6.4

Context





Better Focused Answers

- ➔ Project Indexing and search for larger context
- ➔ Intelligent Context Retrieval for focused context
- ➔ Stronger game comprehension

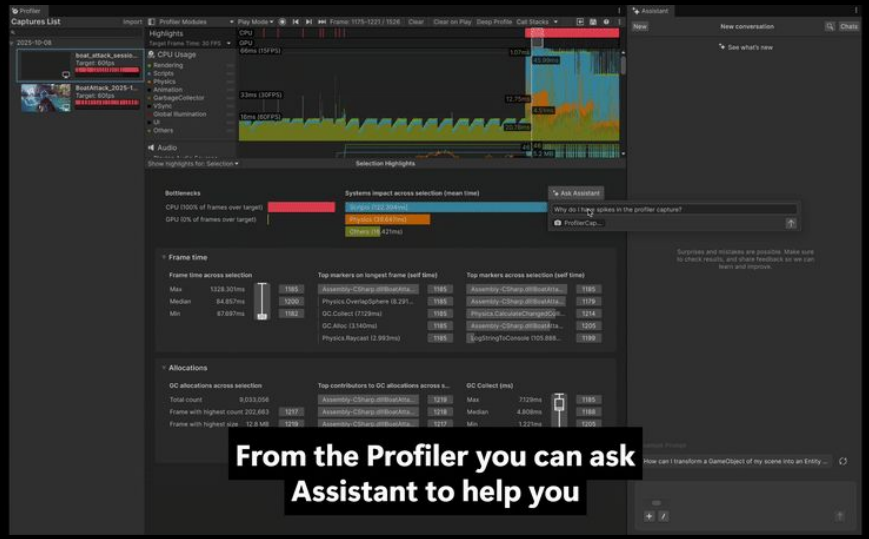
6.4

Visual Debugging & Composition

- ➔ Vision models can now receive images from your project, scene, or references
- ➔ Stronger game comprehension

6.4

Context

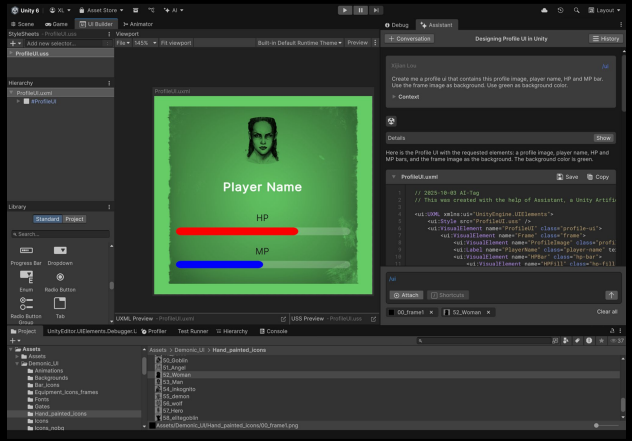


Models Can Analyze Profiler Data

- ➔ Assistant analyzes Profiler data to find bottlenecks and suggest optimizations
- ➔ Easier to make a more performant game

6.5

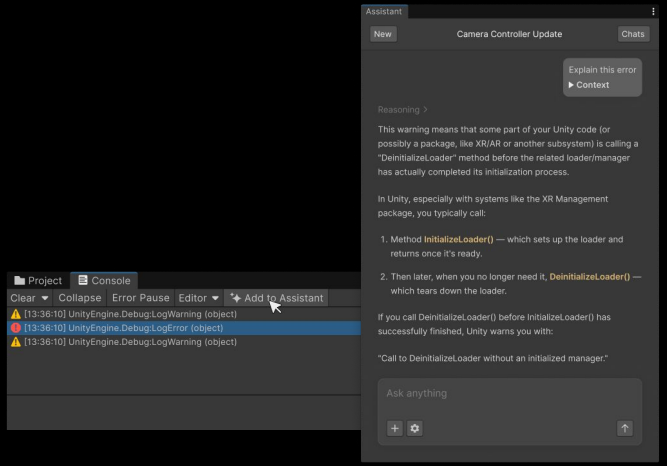
Workflows: Embedded Agents



UI Builder Agent

- ➔ Text to UI Toolkit layouts (USS & UXML)
- ➔ Rapid iteration on UI designs

6.4

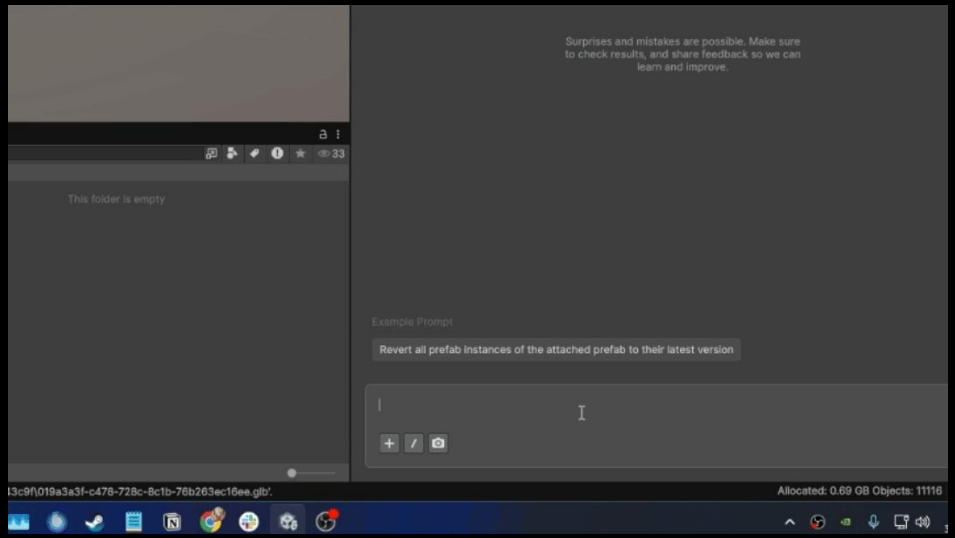


QA Agent

- ➔ Helps validate your builds
- ➔ Monitors console logs
- ➔ Creates simple play mode tests

6.6

Workflows: Generative



Embedded, Richer Asset Generation

Produce a growing list of asset types, seamlessly inside the Assistant to support rapid prototyping.

New:

- → UI Toolkit layouts (USS & UXML)
- → Realistic Skyboxes
- → Performant 3D meshes & textures
- → Effects
- → And more

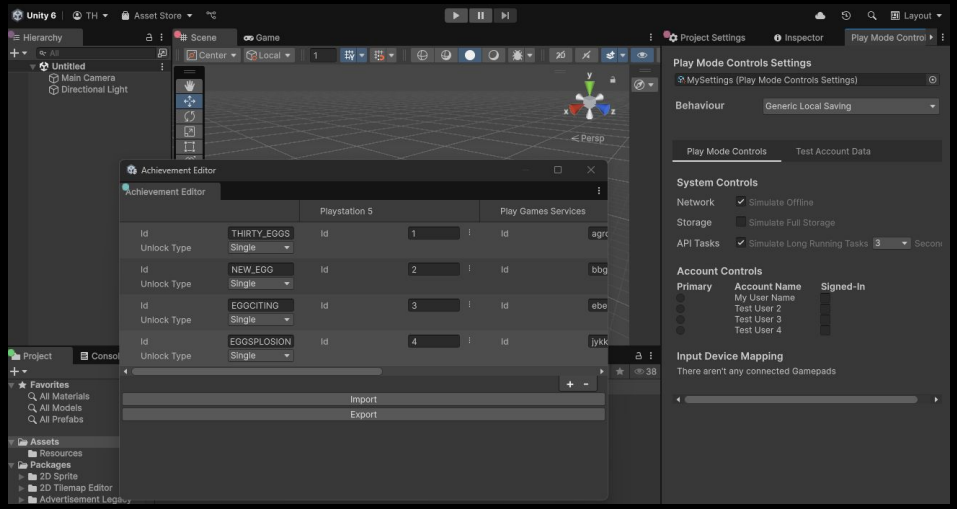
6.4

Deploy Everywhere

Platform Toolkit

Cross Platform API

- → Abstract common features
- → Pass certification more reliably
- → Test all features in-editor
- → Integrate with a single package

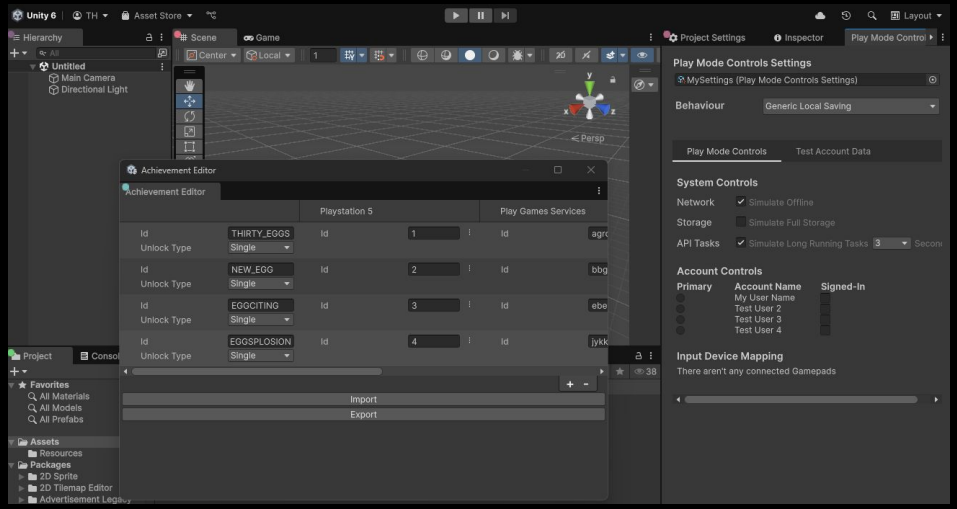


6.3 LTS

Platform Toolkit

Benefits

- → Write platform integrations once
- → All features supported in editor
- → Configuration in editor
- → Validation and testing tools
- → Fast integration
- → Fast iteration



6.3 LTS

Platform Toolkit

Supported features

- → Accounts
- → Achievements / Trophies
- → Storage
- → Data Store
- → Controller Ownership
- → Capabilities

Supported Platforms

- → Android™ / Google Play Game Services
- → iOS / Gamekit
- → Nintendo Switch™
- → Nintendo Switch™ 2
- → PlayStation®5
- → Windows / GDK
- → Windows / Steam
- → Xbox One
- → Xbox Series X | S

6.3 LTS

*Nintendo Switch is a trademark of Nintendo

Nintendo Switch 2

- → Day one support
- → Proven certification pass through in-house developed titles
- → Enhance your existing titles

Skate Story - Sam Eng / Devolver Digital

LEGO Voyagers - Light Brick Studios / Annapurna Interactive

6.0 LTS 6.3 LTS

**Nintendo Switch is a trademark of Nintendo*

XR

Expanding capabilities on Android XR

- → Facial expressions
- → Augment real-world objects
- → Automated Dynamic Resolution
- → 3D Surface Reconstruction

Unlocking innovative spatial experiences on Quest

- → Environment Raycast
- → Room Mesh
- → Faster post-processing

6.3 LTS

Mobile

- DisplayConnection.swift
- DisplayController.swift
- DisplayLinkCommon.swift
- GraphicsDevice.swift
- MetalRenderer.swift
- Renderer.swift
- UnityBackbuffer.swift

Apple

→ Modernization of the Unity x Apple OS glue layer

→ Moving to Swift

→ New Public APIs for lifecycle management

6.6

Android

→ Loading times improvements & thinLTO support

→ Safe Area & Insets improvements

→ Extended support for Foldable devices

6.6

HTTP/2 & gRPC

→ HTTP/2 enabled by default for UWR to improve server and client-side load

→ Available now for ALL platforms

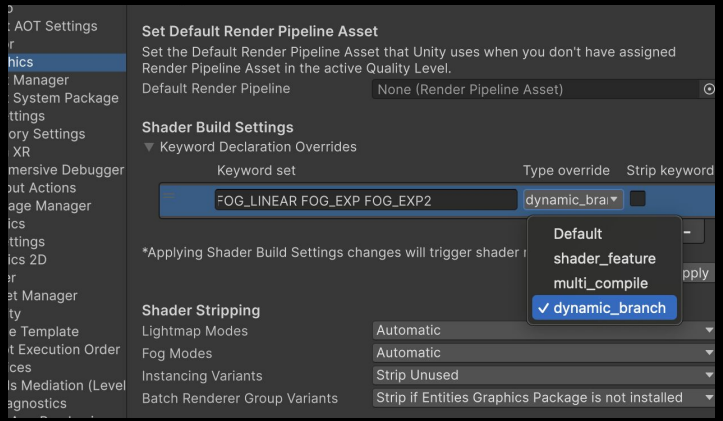
6.3 LTS

→ UWR APIs to support gRPC integrations

6.5

	HTTP 1.1	HTTP 2
Download time	16 sec	14.5 sec
Server CPU peak	69%	38%
Client CPU peak	17%	17%
Client CPU load	15-16%	13-14%

Per-profile shader settings



Shader Build Settings

→ Exclude and convert shader keywords to reduce the project compilation time. Without scripting!

6.3 LTS

URP 3D Sample (Highest Quality) - Shader Compilation Time

Build Profile Override

→ Override the shader build settings on a Build Profile basis for fast development builds, while maintaining optimal performance

6.3 LTS

Modern graphics APIs

Gear.Club Unlimited 3 - Eden Games

Performance, stability, memory

- → DX12 as default on Windows
- → Split Jobs for multithreading in Editor
- → Stability with Device Filtering
- → Graphics State Collection API

6.2

No Rest for the Wicked - Moon Studios

Optimized memory for DX12

- → Tight Buffer Alignment
- → Scratch Buffer Optimizations

6.3 LTS

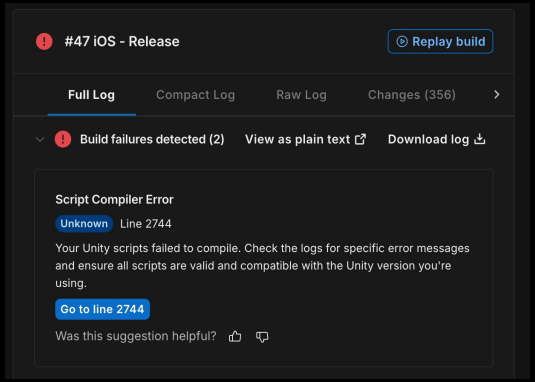
URP 3D Sample - Unity

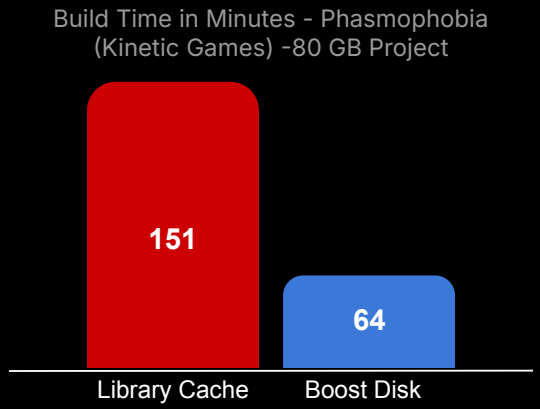
Smoother rendering and gameplay

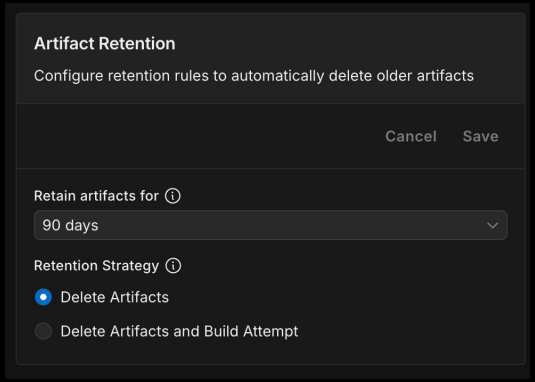
- → Utility functions & WebGPU support
- → VFX Graph compatibility
- → Editor UI and workflow automation

6.5

Unity Build Automation







Build Failure Categorization

→ Troubleshoot faster with improved insights with new categorization

6.3 LTS

Boost Disks

→ Reduce avg incremental build time by up to 58% on Phasmophobia, scales with project size

6.4

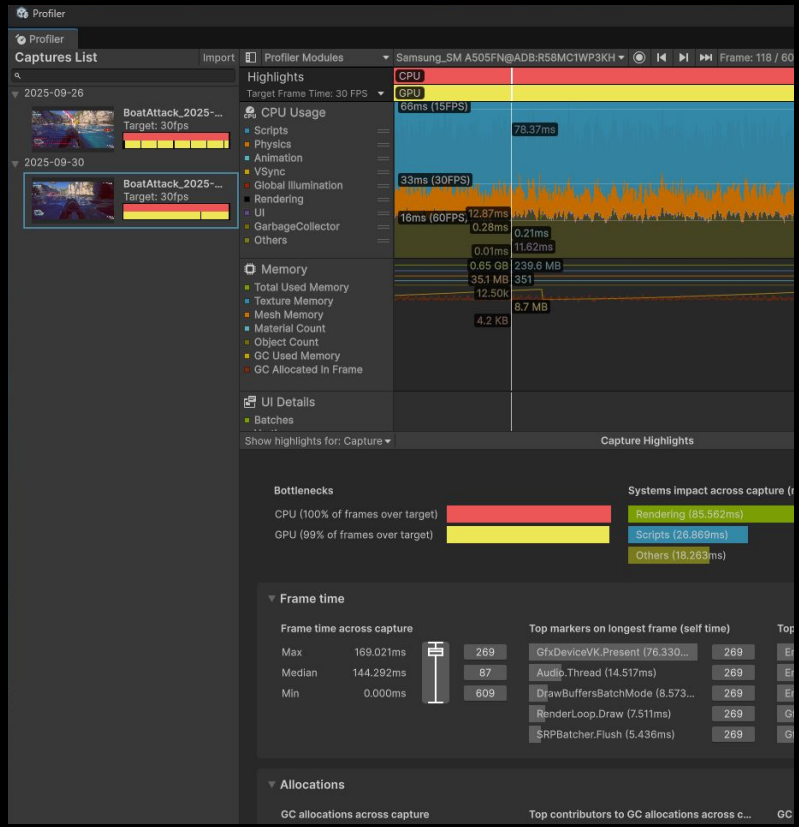
Artifact Retention

→ Optimize storage and security with build artefacts data retention policies

6.4

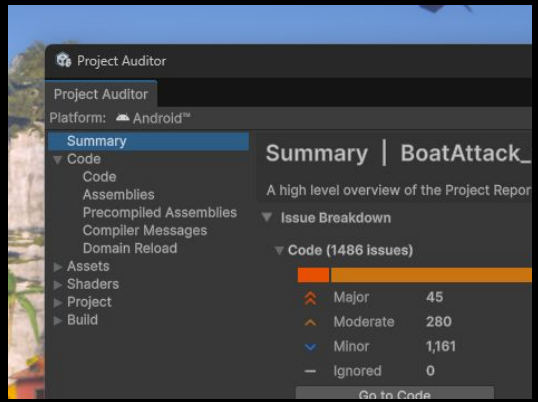
Profiler Overview

- → Track your performance over time
- → Get a simple summary of your data
- → Drill down into what is expensive
- → Optimize where it makes a difference



6.3 LTS

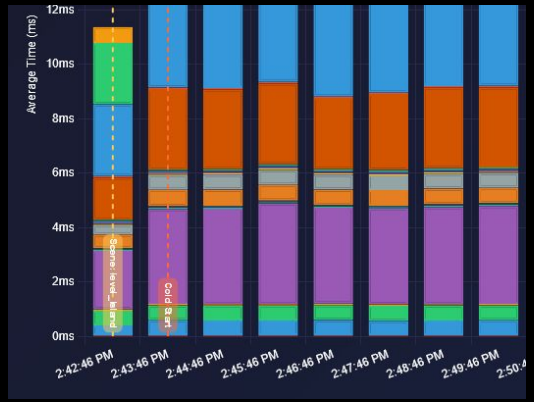
Profiling



Project Auditor in the Editor

→ Enables easier analysis of your Unity project for common issues

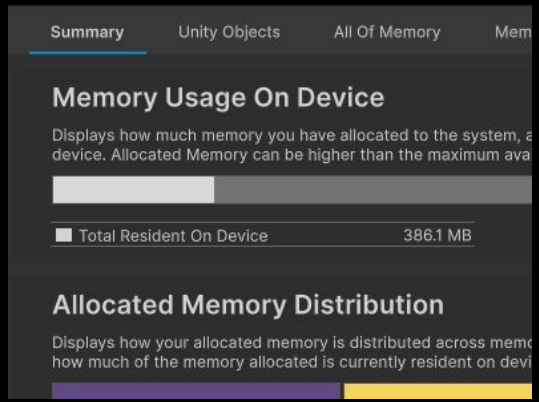
6.4



Real world performance data

→ Optimize your game based on live games performance measurements

6.5



Memory Profiler better insights

→ Objects that remain in memory, or unknown/untracked memory

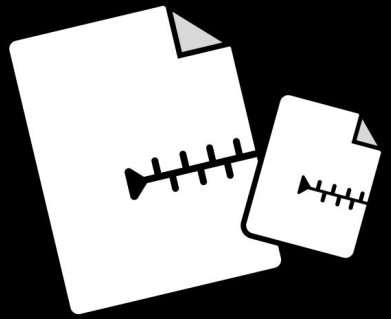
6.5

Optimized live content with Addressables

Documentation

→ Hundred of pages, warnings, and error messages reviewed

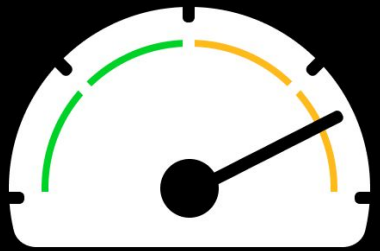
6.3 LTS



Smaller Asset Bundles

→ Reworking how TypeTrees are handled leading to reduced memory footprint!

6.5



Addressables Backend

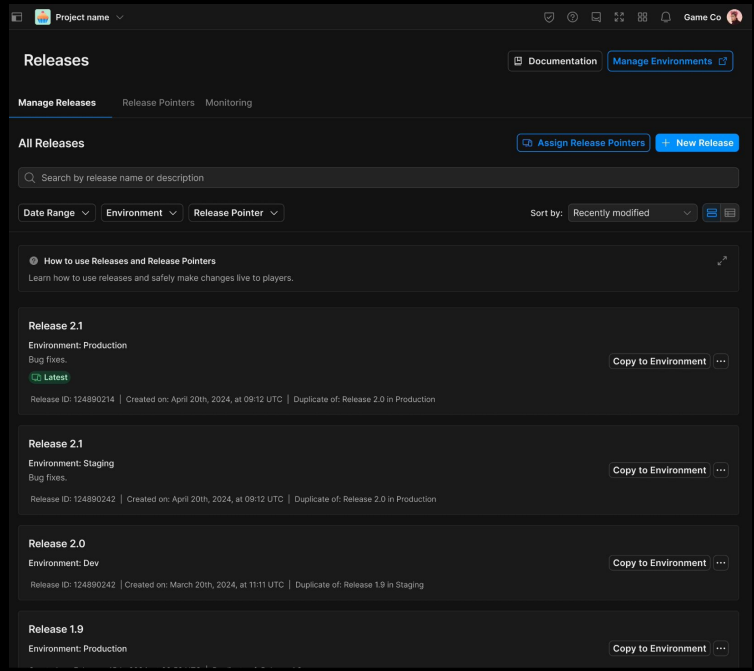
→ Faster, simpler, with better dependency management

6.7 LTS

Grow your Game

Build and Evolve Your Game Economy

- → New economy approach
- → Modular and outcome-focused
- → Production-ready workflows
- → Ship and iterate faster



A New Era for In-Game Purchases

- → Walled gardens opening up
- → Evolving regulations
- → Negotiate lower fees
- → Engage players directly
- → Flexible payment and rewards
- → Greater transaction control

The Next Level of Unity IAP

Maximize revenue through choice

Multiple checkout flows
Providers per market to
Players beyond app stores

One product catalog

Make changes once
Unity syncs across stores
Maintain full control

Data-driven actionable insights

Unified purchase data
One dashboard
Optimize globally

In Beta

Maximize IAP Revenue

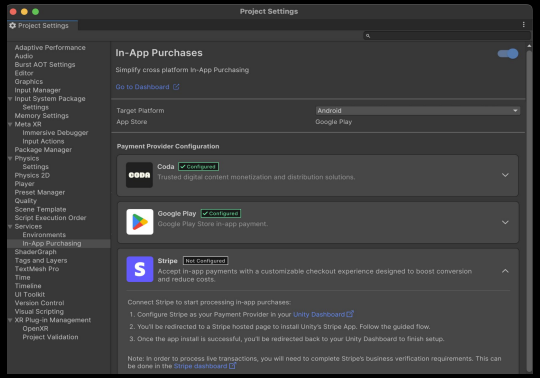
```
public PurchaseProcessingResult ProcessPurcha
{
    bool validPurchase = true;

    var validator = new CrossPlatformValidato
        AppleTangle.Data(), Application.ident

    var result = validator.Validate(e.purchas

    if (validPurchase) {
        // Unlock the appropriate content her
    }

    return PurchaseProcessingResult.Complete;
}
```



Reliable Purchase APIs

→ Built-in server-side checks for predictable, secure transactions

6.3

All Your Stores, One Workflow

→ Unified store management across app stores and payment providers

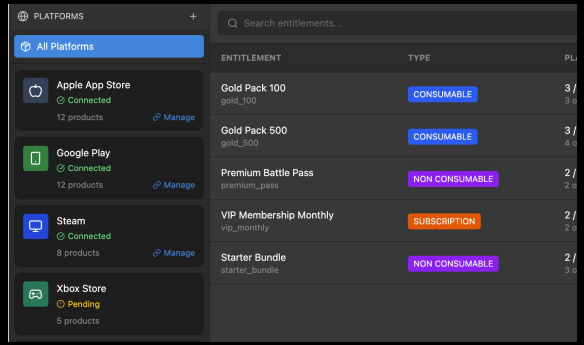
6.4

Optimize Revenue with Data

→ Actionable insights from unified transaction data

6.4

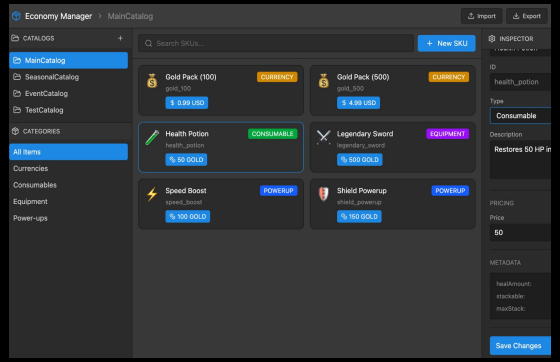
More effective in-game economy tools



Cross-platform entitlement

→ Store entitlements across platforms

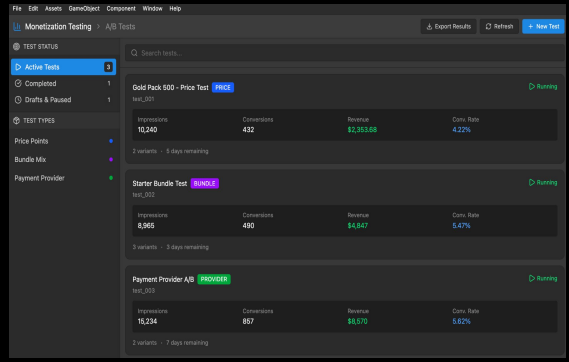
6.6



Cross-platform Catalog

→ Manage your SKUs and catalogs from one place

6.4

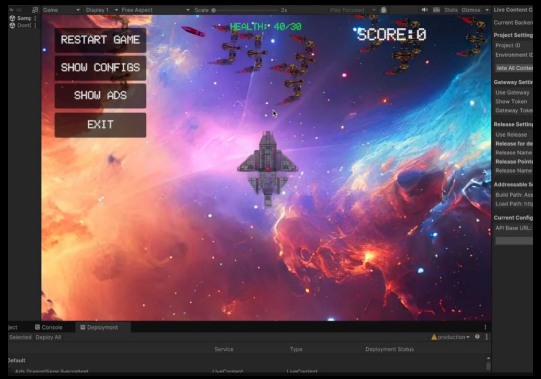


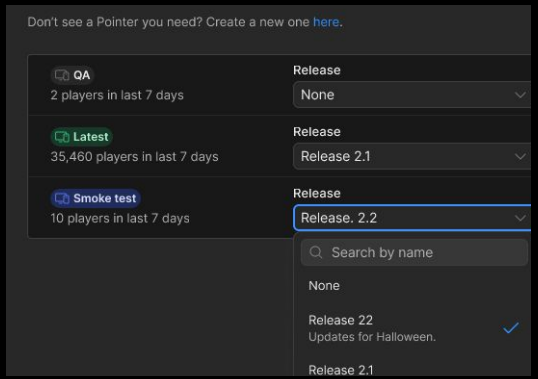
Experimentation & testing tools

→ Test different content, price points, and bundles

6.5

LiveOps - Remote live content





From editor to player

→ Ship from the Editor: atomic releases, QA, and rollback built in

6.5

Streamline release management

→ Automated, flexible workflows to ship across providers faster with lower risk

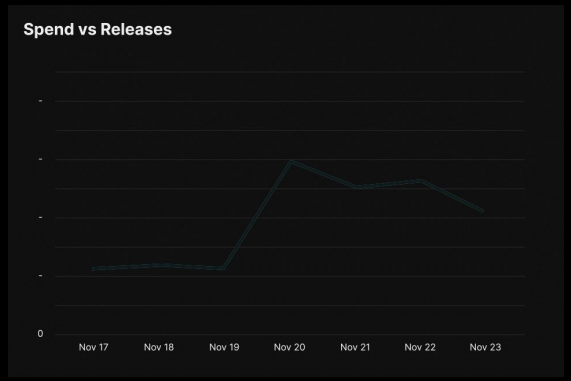
6.5

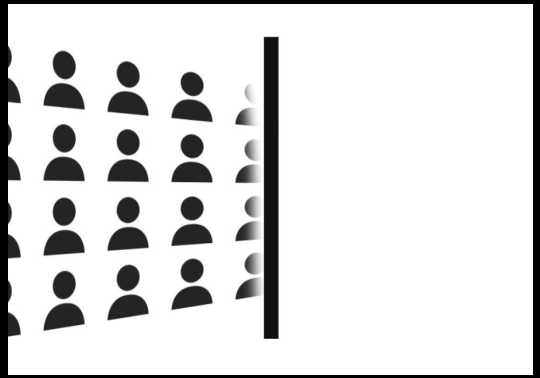
Targeted player experiences

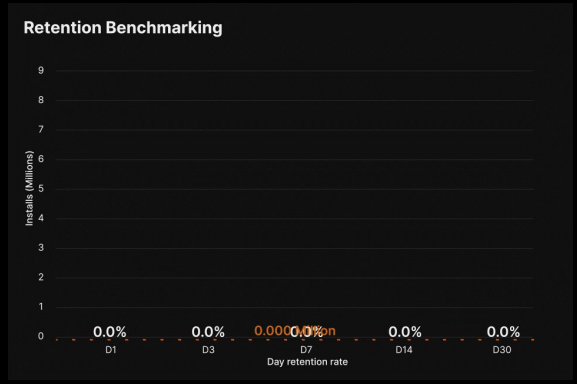
→ Segmentable variants: the right content, to the right player, at the right time

6.5

Optimize your Game with Data







Richer insights where you work

→ Integrated and unified performance, player, and business data

6.4

Personalization

→ Bring your audience to unified stateful and stateless segmentation

6.4

Insights at Scale

→ Market insights & ML-driven LiveOps tailored to your gameplay

6.4

3D as 2D

2D Physics

Cross-platform Commerce

Core Standards

Platform Toolkit

Diagnostics

Agentic AI workflows

Performance optimization

The Unity Roadmap