

UNITY 2022 LTS EDITION \rightarrow E \forall \neg γ



Unity で仮想現実体験と 複合現実体験を作成

目次

はじめに
業界用語9
現実を選択: VR、AR、MR
モーションに関する考慮事項13
XR でのワールド構築14
ナラティブ開発14
環境ストーリーテリング14
目的に合わせたデザイン14
ビジュアルとオーディオによる手がかり14
対話的な要素15
ユーザーエンゲージメント戦略15
細部まで没入
探求を促す15
適応的難易度16
ソーシャルインタラクション
反復設計16
Unity は初めてですか?
XR 用アセット作成19
コンセプトとデザイン19
モデリング 19
ピボットポイント20
テクスチャリング21
トリムシート 21
リギングとアニメーション

	モデル最適化22
	エクスポート
Unity	での XR: 重要なシステムとツールセット
	レンダーパイプライン25
	AR Foundation
	XR Interaction ツールキット28
	XR 開発用の SDK
	OpenXR
	エディターのバージョンとモジュール30
	VR コアのサンプル
	バージョン 3.0 へのアップデート
	Tutorials ウィンドウ35
	サンプルシーン35
	サンプルシーン設定37
	プロジェクトテンプレート40
URP 경	を使用した新しい VR プロジェクトの作成41
	3D URP テンプレートで開始47
	XR Plug-in Management42
	インタラクションプロファイルの追加44
	XR Interaction Toolkit の実装46
	Starter Assets を使用した XRI の探求47
	デモシーン47
	Starter Assets デモシーンでの XR インタラクション設定
	Input Action Manager
	XR Interaction Manager49
	XR Origin49

Character Controller Driver50
XR Input Modality Manager51
TrackedPoseDriver (Input System) 51
Controller Input Action Manager52
XR Interaction Group コンポーネント52
XR Transform Stabilizer
インタラクションの設定53
XR Direct Interactor
Locomotion System53
回転54
移動54
Teleportation54
クライミング55
Tunneling Vignette
VR 世界との相互作用56
XRGrabInteractable
Poke インタラクタブル57
XR Simple Interactor
XR Poke Filter
XR Poke Follow Affordance58
Gaze インタラクタブル58
XR Gaze Interactor
Gaze Input Manager60
UI 要素とのインタラクション
XR UI Input Module コンポーネント 61
Tracked Device Graphic Raycaster 61
 サンプルプロジェクトを使用した VR 開発

全体像64
エンジンを始動させる66
Unity のバージョンと URP66
Build Settings67
Quality 設定
世界の構築68
プロトタイプとモデリング68
効率的なテクスチャリング69
光あれ70
ライティング方法の選び方
ライティングの最適化7
仮想世界のナビゲーション
テレポーテーションのタイプ72
XR Interaction Toolkit を使用してパズルを作成74
XR Grab Interactor
XR Socket Interactor
Socket Interactor の作成77
テストとイテレーション81
XR Device Simulator
XD Device Simulator のインストール83
Unity Profiler
ビルドとデプロイ
複合現実と空間コンピューティングのアプリケーション88
ユーザーインタラクションとインターフェースデザイン.88
空間認識と物理的性質89
クロスプラットフォーム開発戦略89
環境デザインと没入感89

- MR Interaction Setup プレハブ......91 AR Plane Manager92 MR テンプレートのテスト94 Apple Vision Pro による空間コンピューティング 95 Unity で visionOS を始める......96 インタラクション.....101 スケルトンハンドトラッキング.....102 完全没入型 VR103 MR アプリケーション.....106 Unity PolySpatial......108 グラフィックスとシミュレーション108
 - ボリュームカメラ109
 - Bounded カメラ.....110
 - Unbounded カメラ111

 - ウィンドウアプリケーション 112
 - visionOS プロジェクトテンプレート 113
 - その他のリソース.....115
 - プロフェッショナルトレーニングサービス 115
- Unity での AR の活用116

はじめに

没入型のバーチャル空間を旅立ち、次元間をテレポートし、デジタルの奇跡を現実世界とマージさせましょう。 仮想現実 (VR) と複合現実 (MR) の可能性がクリエイターの想像力をかき立てます。

Unity の包括的なガイドは、クリエイター志望者と熟練開発者の両方にとって、Unity を使用して VR/MR 体験 (または総称して "XR")を構築するための詳細を深く理解するのに役立ちます。

Unity は、2024 年の以下の調査結果が示すように、XR 体験を作り出すための主要プラットフォームの 1 つです。

- Vision Pro 向けの必見 Apple アーケードゲーム (第1週) の 50% 以上が Unity を使って制作された ものでした。¹
- 2024 年 2 月時点で、最も人気のある Quest 体験の 2/3 は Unity を使って制作されたものでした。²
- 2024 年 2 月時点で、Quest の人気ゲームの少なくとも 70% が Unity を使って制作されたものです。³
- 2023 年の売り上げ上位の Steam VR エクスペリエンスのうち、60% 以上が Unity を使って制作 されたものでした。⁴

1 出典:Apple Inc.

² 出典:SteamDB

³ 出典:SteamDB

⁴ 出典:SteamDB



没入感のあるインタラクティブな現実を制作するのに不可欠なツール、方法論、テクニックを学びます。環境の 構築から直感的なインタラクションの実装まで、VR/MR アプリケーションに命を吹き込むために必要なヒントと ガイダンスが得られます。

実体と幻想の境界が霞む、現実と想像が交錯する世界にステップを踏み出しましょう。Unity をガイドとして、 心をつかみ、インスピレーションを刺激する没入型現実の制作に取りかかりましょう。

主な著者

Daniel Stringer は経験豊富な 3D アーティスト、Unity 開発者、コンテンツクリエイターであり、20 年に及ぶ さまざまな業界にわたる専門知識を持っています。ゲーム、トレーニングとシミュレーション、建築ビジュアライ ゼーション、映像、VFX の制作に 20 年の経験があります。Daniel の実績には、受賞歴のある VR アプリケー ションの開発があり、特に英国の医療セクターと商業無線業界に貢献しています。

その他のコンテンツについては、YouTube と Udemy を参照してください。主に VR を中心とした Unity 関連の テーマを取り上げています。

Unity の貢献者

このガイドには、Unity の XR 製品、製品マーケティング、およびコンテンツマーケティングの各チームから以下の 同僚たちが寄稿しています。

Adam Axler, Dan Martinez, Eduardo Oriz, Isaac Seah, Kevin Semple, Leah Martin, Matt Fuad, Miruna Dumitrascu, Sam Zhang, Shanti Zachariah, Tricia Becker。



BigBox VR の『Population ONE』(左)、Owlchemy Labs の『Job Simulator』(右)



業界用語

Unity を使用した開発についてのコンテキストでは、XR は以下の種類のアプリケーションを含む総称です。

- 一 複合現実 (MR):MR は通常、ビデオシースルーディスプレイを使用するヘッドセットシステムに適用 され、現実世界をカメラで捉え、バーチャルコンテンツとデジタルに組み合わせてユーザーの目前に 提示します。
- 仮想現実 (VR):VR は完全没入型のデジタル体験を提供し、ヘッドセットや特殊なデバイスを通じて、 ユーザーを完全にシミュレートされた環境に移します。
- ー **拡張現実 (AR)**:AR とは、ユーザーが現実世界を直接見ることができ、その上に仮想コンテンツが 重ねられる、透明なグラスやバイザーなどの光学透過型ディスプレイを使用するシステムのことです。
- 空間コンピューティング:デジタル情報が現実世界にシームレスに統合され、仮想環境と現実世界が 融合した没入型の体験を創出するプロセスとテクノロジーのことを指します。

VR から AR の間は 1 つの連続体であり、MR はその中間に位置します。これは、特定のデバイスの能力と アプリケーションの意図によって異なります。

これらの用語は幅広い没入型テクノロジーを表し、各テクノロジーはゲーム開発やそれを超える創造的 探究のための独自の体験や機会を提供します。





上から下へ: VR で実行される URP 3D サンプル、Meta Quest 3 ヘッドセットで実行される Resolution Games による『Demeo Battles』、Unity の visionOS Spatial サポートのためのインタラクティブなサンプル、およびスマートフォンのカメラフィードに要素を追加する AR アプリケーション

現実を選択: VR、AR、MR

このセクションでは、制作する没入型ゲームの種類について意思決定に役立つ要素を検討します。



XR ゲーム開発における多くの略語と選択肢

没入型体験の目標

求める没入感のレベルが選択を左右します。VR は完全な没入感を提供し、AR と MR は現実に対するコン テキストの強化を提供します。

ここで決定することで、VR ヘッドセット、AR と互換性のあるスマートフォンやグラス、MR デバイスなどの ターゲットハードウェアも決まります。



MR に関する注記: MR を実現する XR ヘッドセットの重要な特徴は、パススルーです。ヘッドセットによって ユーザーに現実世界のライブフィードが表示される機能のことを指します。通常は正面に搭載されたカメラを 通じて、ヘッドセットを装着している間にユーザーは周囲の環境を見ることができます。

パススルーは、現実世界にデジタル要素を重ねる MR アプリケーションに必要です。また、ヘッドセットを 外さずに環境を操作できるため、ユーザーの安全性と利便性が向上します。この特徴は、完全なバーチャル 没入感とユーザーの実際の物理的空間の間のギャップを埋めるものです。

ユーザーインタラクションとエンゲージメント

アプリケーションでユーザーにどのような操作を行ってもらいたいと考えていますか?これは重要な事です。 AR と MR は物理環境と仮想環境を組み合わせたものですが、VR はユーザーを専用のデジタル領域内に 隔離します。

ナラティブおよびゲームプレイ要素

ナラティブとゲームプレイのメカニクスは重要な役割を果たします。VR は完全没入型のストーリー中心の 体験に適している一方で、AR/MR はロケーションベースのゲームプレイを強化できます。

ユースケース

実用アプリケーションがその決定に影響します。AR/MR は、現実世界とのインテグレーションが有益な、 トレーニングシミュレーションや教育体験に最適です。

開発リソースと制約

開発者の専門知識、リソース、技術的な制約も選択に影響します。それぞれの現実タイプでは、それぞれ 異なる開発アプローチが必要になるからです。



『I Expect You To Die 2: The Spy and the Liar』 Schell Games の受賞作である VR ゲームの続編



モーションに関する考慮事項

もう1つの重要な考慮事項はモーションです。その理由は、次のとおりです。

- **ユーザー快適性:**過剰なモーションや非現実的なモーションは、不快感や乗り物酔いのような症状に つながるおそれがあります。方向感覚の喪失や吐き気を避けるために、アプリケーションがスムーズに 動くようにします。
- インタラクション設計:ユーザーはバーチャル空間内で直感的に移動して操作できる必要があります。 体験のタイプ (ゲーミング、トレーニング、探検など) に基づいて動きのメカニクスを設計します。
- パフォーマンスの制約:スピードが速いモーションや複雑なモーションでは、処理能力がさらに必要に なり、アプリケーションのパフォーマンスに影響を及ぼす場合があります。開発サイクル全体を通して パフォーマンスをテストおよび最適化することで、意図したデバイスでアプリケーションがスムーズに 実行されるようにします (このガイドで後述するテストとイテレーションのセクションを参照)。
- 物理的空間の制限:ユーザーが持つ物理的空間を考慮しましょう。トレーニングのために自宅や職場で アプリケーションを使用する予定はありますか?異なる場所で安全に使用できるようにアプリケー ションを設計しましょう。
- **ユーザーアクセシビリティ:**すべてのユーザーが自由に動けるわけではありません。モビリティに制限が あるユーザー向けに代替インタラクション方法を提供し、より多くのユーザーがアプリケーションに アクセスできるようにしましょう。

これらの側面を理解して対処し、アクセスしやすく快適で魅力的な AR/VR 体験を開発しましょう。



コントロール、モーション、物理的空間のコンテキストはすべて、ゲームまたは産業用 XR アプリケーションに固有のものとなります。

XR でのワールド構築

ワールド構築とは、単に視覚的に美しい環境を作成することではありません。ストーリーを伝え、感情を伝え、 ユーザーを魅了する没入型の世界を構築することです。このセクションでは、XR でのワールド構築のビル ディングブロックについて説明します。

ナラティブ開発

すべての仮想世界は、その開発を導く強力なコンセプトから始まります。これは、プレイヤーが解明できる ような独自の設定、説得力のあるストーリー、または興味深い謎の場合があります。このコンセプトは、 ビジュアルデザインから対話的な要素まで、世界のすべての要素のバックボーンとして機能する必要が あります。

環境ストーリーテリング

目的に合わせたデザイン

仮想環境のすべての要素は、ストーリーを進める、雰囲気を高める、手がかりや課題を提供するなど、目的を 果たす必要があります。雑然とした表現や体験全体に寄与しない要素は避けましょう。

ビジュアルとオーディオによる手がかり

ビジュアルおよびオーディオによる手がかりを使用して、ストーリーと設定をガイドします。ライティング、 サウンドスケープ、建築デザインは、注意を引き、雰囲気を設定し、背後にあるストーリーのヒントを与える ことができます。



対話的な要素

世界とそのストーリーの詳細を明らかにする対話的な要素を取り入れましょう。例えば、歴史の一部を 明らかにする古代の遺物、一人称視点の物語を提供する日誌、秘密のエリアや伝承を解き明かす環境 パズルなどが挙げられます。



Resolution Games による『Demeo』。すべての主要プラットフォームで利用可能な MR ゲーム。

ユーザーエンゲージメント戦略

細部まで没入

環境内の細部に注意を払いましょう。テクスチャ、サウンドエフェクト、対話的オブジェクトは、アーティス ティックな意図と一致する必要があります。プレイヤーを没入させ、熱中させるために、まとまりのある 一貫した体験を作成しましょう。

探求を促す

あなたの世界を探検したユーザーに報酬を与えましょう。隠しエリア、秘密の伝承、復活祭の卵は、ユーザーが ゲームの本編を超えて視野を広げる動機になり、体験と世界との関わりを豊かにします。



適応的難易度

ユーザーのパフォーマンスに基づいて課題を調整する適応型難易度メカニズムを実装しましょう。これにより、 すべてのスキルレベルのユーザーを惹きつける体験が実現し、難しくてもフラストレーションを感じさせません。

ソーシャルインタラクション

必要に応じて、ソーシャルインタラクションをあなたの世界に組み込みます。ユーザーが協力してパズルを 探ったり解いたりできるマルチプレイヤー要素は、没頭する側面を追加し、体験をより迫力あるものにする ことができます。



『Job Simulator』という Owlchemy Labs の作品には、どんどん困難になりながらも楽しくなるタスクを提供する、こと細かな環境があります。

反復設計

フィードバックを活用して反復的に世界を洗練させましょう。ナラティブフローを調整するか、ナビゲーションを 改善するために特定のエリアを再設計するか、エンゲージメントを高めるためにより対話的な要素を追加 します。

これらの戦略をワールド構築に組み込むことで、VR/MR 体験の品質とインパクトを高めることができます。 ナラティブ開発、環境ストーリーテリング、ユーザーエンゲージメントに重点を置くことで、単なるゲームや アプリケーションではなく、ユーザーが何度でも繰り返し戻りたくなるような魅力的な世界を作り出します。



Unity でのゲームデザインとレベルデザインの詳細については、以下の e ブックをダウンロードしてください。



Unity は初めてですか?

Unity Learn では、エディターの操作法や使い方を学ぶための豊富な無料リソースを用意しています。新規 ユーザーには、次の3つの パスウェイ コース (Unity Essentials・Junior Programmer・Creative Core) がおすすめです。基本的なパスウェイコースを完了したら、VR Development パスウェイを試してください。

最後に、Unite と GDC の詳細なセッションと基調講演を参照し、XR スペースと Unity の最新動向を把握 できます。以下のリンクから開始:

- Unity at GDC 2024
- Unite 2023
- XRI Session | XR Developer Night 2024
- XR Roadmap 2024 XR Developer Night 2024
- Hand Input & OpenXR Development | XR Developer Night 2024
- XR Composition Layers | XR Developer Night 2024
- Mixed Reality Authoring & Simulation | XR Developer Night 2024
- AR Foundation | XR Developer Night 2024

XR 用アセット作成

XR ゲーム用のアートアセットを作成または調達する際には、重要な考慮事項があります。小さな小道具から キャラクターまで、これらのヒントに従って開発サイクルの後半で時間を節約しましょう。

コンセプトとデザイン

作成したいもののはっきりとした概念をもって開始します。アイデアをスケッチし、3D 360 度の空間での アセットの表示方法を検討しましょう。これらの概念は、環境の見た目からパズルのしくみまで、アイデアを 形にするのに有益です。スケッチは、早い段階で他の人に自分のアイデアを伝えることで、あなたと協力者 たちがより多くの情報に基づいた意思決定を下し、効果的な計画を立てるのに役立ちます。

コンセプトをシンプルな手描きの絵として作成することで、特に初期段階でインタラクションやビジュアルを すばやく計画できます。大規模なスタジオでは、アイデアに命を吹き込むコンセプトアーティストがいますが、 個人や小規模なチームでは、アイデアを表現するのに AI を活用することもできます。

モデリング

3D モデリングソフトウェア (Blender、Autodesk Maya、3ds Max など) を使用してモデルを構築しましょう。 特にモバイル AR では、パフォーマンスのためにポリカウント (ポリゴン数) を考慮する必要があります。 ポリゴン数の多いモデルを最初に作成し、後で最適化できます。

ピボットポイント

ピボットポイントは、モデルの回転とスケーリングの中心として機能します。適切に配置することでアニメー ションプロセスを合理化し、動きや変換をより自然かつ直感的に行うことができます。たとえば、人物像の 下部にピボットポイントを設定することで、人間が足を軸にして回っているような、よりリアルで安定した 回転が可能になります。この配置は、アニメーションとゲーム開発の両方で環境との正確なインタラク ションを行うために不可欠です。

Unity の 3D シーンで作業する場合、Y 軸は垂直方向の動きまたは高さを表します。この向きは、多くの 3D グラフィックス環境の標準規則です。X 軸は水平方向の動き (左右)、Y 軸は垂直方向の動き (上下)、Z 軸は 深度 (前後) を表します。この軸システムを理解することは、Unity 内でオブジェクトを正しく配置、回転、 スケーリングするために不可欠です。

3D モデリング ソフトウェアで作業するときに、モデルのピボットポイントを設定し、向きを Unity に合わせる ことができます。以下は、3ds Max でピボットポイントを設定する方法の例を示します。この例では、Z 軸を アップ軸として使用します。



アップ軸の調整

次の2つのステップで、Unity でのモデルの向きを修正することもできます。

- 1. Hierarchy に空のゲームオブジェクトを作成します。
- 2. モデルをこのゲームオブジェクト内にネストし、その位置をすべての軸で 0,0,0 に設定して、 完全に中央揃えにします。

この方法では、すべての軸で中立的な回転 0 を維持する親ゲームオブジェクトにモデルが固定されます。 また、前を向いている Z 軸に合わせてモデルが正しい方向を向き、すべての軸で均一なスケール 1 が維持 されます。また、インポーターで Bake Axis Conversion を試すこともできます。

テクスチャリング

テクスチャを適用して、モデルにリアルな外観または様式化された外観を与えます。詳細なテクスチャに ついては、UV マッピングの使用を検討してください。ゲームの美観によっては、PBR Textures が必要に なる場合があります。Unity のデフォルトの Universal Render Pipeline (URP) シェーダーは物理 ベースであるため、マテリアルの物理的特性を現実的な方法でシミュレートします。テクスチャ、反射率、 粗さなどの要素を考慮し、ライトとサーフェスの相互作用を正確に描写するアルゴリズムを使用しています。 Unity の物理ベース (PBR) シェーダーには通常、2 つの主要コンポーネントが含まれます。メタリック/粗さ ワークフローと反射/光沢ワークフロー。それぞれマテリアルの特性の異なる側面を処理します。その結果、 よりリアルでダイナミックなビジュアルを実現し、Unity ベースのゲームやシミュレーションの 3D シーンや オブジェクトの現実感を高めます。

トリムシート

トリムシートは、3D モデリングやゲーム開発で使用されるテクスチャリングツールです。成形、エッジ、 境界線など、さまざまなトリムパターンやディテールを含む1つのテクスチャで構成されます。アーティストは、 これらのトリムを3D モデルのさまざまなパーツに適用し、固有のテクスチャを複数用意することなく、複雑な ディテールを追加できます。この方法は、モデルの視覚的に複雑で多様なデザインを可能にする一方で、必要な 個々のテクスチャの数が減るため、ゲームのパフォーマンスの最適化が効率的に行えます。



環境全体のさまざまなメッシュに使用されるトリムシート

リギングとアニメーション

キャラクターのようにモデルに動きが必要でメッシュをデフォームする必要がある場合は、スケルトン構造体を使用してリギングし、アニメーション化できます。

ー人称視点の VR アプリケーションでは、アバターハンドを設定する必要があります。Animation Rigging パッケージの Runtime IK 機能を使用して腕のボーンを調整し、TwoBonelK コンストレイントで手を追跡 対象のコントローラーの位置に合わせることができます。

肘の位置は一般的に追跡されないため、興味深い課題です。肘の位置を連続的に推測できる場合、 TwoBonelK コンストレイントは Hint Transform を入力として使用して肘のボーンを揃えることができます。



Endeavor One による『Arashi: Castles of Sin- Final Cut』。TwoBonelK システムを使用した VR の手と腕が表示されています。

モデル最適化

ポリゴン数とファイルサイズを削減し、テクスチャのサイズを効率的に設定することで、Unity でのパフォーマンスに合わせてモデルを最適化する必要があります。法線マップを作成することで、モデル内のメッシュの詳細をキャプチャできます。

法線マップ 3D は、表面の各ピクセルの法線 (垂直ベクトル) の方向をエンコードするテクスチャです。これに より、モデルのポリゴン数を増やすことなく、凹凸などの詳細な表面テクスチャを作成できます。ライトの相互 作用の仕方を変えることで平面に深度とディテールがあるように錯覚させ、現実感を高めます。



3D モデリング用の法線マップを生成する最善の方法は次のとおりです。

- ポリゴン数の多いモデルから:ポリゴン数の多い詳細なモデルとポリゴン数の少ないモデルを作成します。次に、Blender や Substance Painter などのソフトウェアを使用して、ポリゴン数の多い モデルの表面のディテールを法線マップとしてポリゴン数の少ないモデルにベイクします。
- **2D 画像エディターの使用:**Adobe Photoshop や GIMP などのプラグイン付きツールでは、グレースケールの高さマップを法線マップに変換できます。この方法は、布地や石などの表面テクスチャで便利です。
- 専用の法線マップジェネレーター:CrazyBump や NormalMap Online などのソフトウェアは、
 2D 画像から法線マップを生成できます。
- フォトグラメトリ:フォトグラメトリ手法を使用して現実世界のオブジェクトをデジタルモデルに変換する
 ことで、法線マップを作成します。



SUPERHOT Team の SUPERHOT VR は、テーマに合った低ポリゴンの美麗さを追求し、ゲームプレイに必要なパフォーマンスを実現しています。

エクスポート

Unity と互換性のある形式でモデルをエクスポートします (FBXを推奨)。

3D モデルのエクスポートスケールは、Unity で AR や VR を扱う際に重要です。スケールは、モデルが仮想 世界内にどのように適合し、現実世界の次元と相互作用するかを決定します。特に、現実世界との相関が 不可欠な AR では重要です。スケールの不一致は、非現実的または非実用的なインタラクションにつながる 可能性があります。モデルのスケールが Unity の単位および意図した実世界または仮想環境のスケールと 一致することが必須です。

Unity では、1 単位が 1 メートルに正確に対応する、直感的な測定スケールが設計されています。インポートした アセットのスケールを測定するには、シーンにキューブゲームオブジェクトを導入するだけです。このキューブ (寸法 1 m x 1 m x 1 m) は、あらゆるアセットのスケールを即時かつ正確に評価するための基準として機能 します。

環境や小道具を作成するときに重要なのは、美観とパフォーマンスのバランスを取り、スムーズで魅力的な AR/VR 体験を実現することです。



ProBuilder でさまざまな高さの面のプロトタイプを作成し、FBX Exporter で 3D モデリングソフトウェアにエクスポートして微調整できます。

Unity での XR:重要な システムとツールセット

レンダーパイプライン

Unity のレンダーパイプラインは、さまざまなプラットフォームやパフォーマンスのニーズに対応する豊富な レンダリング機能を提供します。



Unity URP 3D サンプルからの庭園環境 (左)、HDRP ボリュメトリック環境 (右)

Unity URP 3D サンプルからの庭園環境 (左)、HDRP ボリュメトリック環境 (右)

XR アプリケーションを開発する場合、ユニバーサル レンダー パイプライン (URP) が推奨されるパイプ ラインです。URP は軽量で、モバイルからハイエンドまでの幅広いデバイス向けに最適化されています。 URP は、最新のレンダリング手法、最適化されたシェーダー、カスタマイズ可能なポストプロセスエフェクトを 提供します。



HD レンダー パイプライン (HDRP) は、ハイエンドの PC およびコンソールプラットフォーム向けで、物理 ベースのレンダリング、高度なライティング、マテリアル、複雑なポストプロセスにより、最高レベルのビジュ アル品質を実現します。HDRP は、グラフィックの限界を押し広げる際に、ターゲットハードウェアで非常に 高いリフレッシュレートを実現できることが保証されている環境で使用します。

スクリプタブルレンダーパイプライン (SRP) は、URP、HDRP、および特定のプロジェクトのニーズやプラット フォーム仕様に合わせてカスタマイズされた独自のカスタムレンダーパイプラインの基盤フレームワークです。

ビルトインレンダーパイプラインは Unity の古い機能のパイプラインです。Unity 2022 LTS 以降の 一般的なマルチプラットフォームゲーム開発では、URP を使用することを推奨します。

適切なパイプラインの選択は、プロジェクトの要件、プラットフォームターゲット、パフォーマンスとビジュアル 品質の望ましいバランスによって決まります。各パイプラインはさまざまなニーズに対応し、さまざまなレベルの パフォーマンスの最適化と現実感のあるビジュアルのオプションを提供します。



AR Foundation

AR Foundation は、没入型の AR 体験を作成するための強力で統一されたフレームワークを提供する Unity のパッケージです。これにより、平面検出、ワールドトラッキング、顔検出、環境理解、オクルージョン などの基本的な AR 機能を実装できます。

AR Foundation には AR 機能用のインターフェースは含まれていますが、機能自体は実装されていません。 ターゲットプラットフォームで AR Foundation を使用するには、そのプラットフォーム用の別個のプロバイ ダープラグインパッケージも必要です。Unity は次のプロバイダープラグインを正式にサポートしています。

- Android O Google ARCore XR Plug-in
- iOS Ø Apple ARKit XR Plug-in
- visionOS Ø Apple visionOS XR Plug-in
- HoloLens 2の OpenXR Plug-in
- Meta Quest の Unity OpenXR: Meta

AR Foundation でサポートされている機能とプラットフォームのリストについては、こちらの ページを参照 してください。

AR Foundation は、複数のプラットフォームにわたる開発プロセスを効率化する API を提供します。プラット フォーム固有の実装の複雑さを取り除くことで、デバイスを超えて幅広いプレイヤーに対応できる AR 体験の クリエイティブな作業に集中できます。



この AR サンプルは、AR Foundation ドキュメントから入手できます。



XR Interaction ツールキット

XR Interaction Toolkit (XRI) パッケージは、掴む、触る、移動する、オブジェクト操作するなどのインタラ クションに対する標準化されたアプローチを提供することで、XR インタラクションを効率的に開発できる ようにします。

このツールキットは 3D インタラクションや UI インタラクションを Unity 入力イベントから利用できるように するフレームワークを提供します。このシステムの中心は、ベースとなる Interactor (インタラクター) コン ポーネントと Interactable (インタラクタブル) コンポーネントのセットと、これら 2 種類のコンポーネントを 結びつけるインタラクションマネージャーです。これに加えて、ロコモーションとビジュアルの描画に使用できる コンポーネントも含まれています。

Unity のクロスプラットフォーム機能を活用する XR Interaction Toolkit により、開発者はさまざまなプラット フォームやデバイスにまたがる異なるインタラクションモデルの管理という複雑な作業にわずらわされること なく、XR 体験の設計に集中できます。



XR 開発用の SDK

Unity は、プラグインフレームワークと一連の機能ツールおよびパッケージを通じて XR 開発をサポートして います。Unity の Project Settings ウィンドウから入手できる XR Plug-in Management パッケージを 使用すると、開発者はターゲット XR プラットフォームで使用できるプラグインを選択できます。

以下の図は、現在の XR プラグインフレームワークの構造体と、それがプラットフォームプロバイダーの実装と どのように統合されるかを示しています。



各 XR プラットフォームには通常、ハードウェアのデバイス固有の機能をすべて使用する独自の API があり ます。たとえば、Meta デバイスは Unity Asset Store で入手できる Meta XR All-in-One SDK を利用 します。OpenXR と互換性のあるプラットフォームでは、OpenXR API を介して非独占的デバイス機能に アクセスすることもできます。例えば、Meta デバイスです。

ハードウェアに必要な SDK にアクセスする方法については、デバイスの開発者ガイドを確認してください。



OpenXR

OpenXR は Khronos によって開発されたロイヤリティフリーのフレームワークで、クロスプラットフォーム XR 体験を作成するために設計されています。Unity の OpenXR プラグインは、準拠する OpenXR ラン タイムをサポートするあらゆるデバイスで機能する必要があります。

OpenXR は XR ハードウェア用の標準化されたインターフェースを提供するため、開発者はデバイス固有の コードを書くことなく、幅広い XR デバイスと互換性のあるアプリケーションを作成できます。これにより、さま ざまなプラットフォームでの XR アプリケーションの開発と展開が容易になり、互換性が向上し、開発の時間と 労力が軽減されます。

エディターのバージョンとモジュール

このガイドでは Unity 2022 LTS エディターを使用しており、同じバージョンまたはより新しいバージョンを 使用することを推奨します。

デバイス	VR	MR/ 空間	AR	Unity モジュール	XR プラグインプロバイダー
Meta Quest 3	V	V		Android Build Support	ARCore、Oculus、OpenXR
Meta Quest 2	V	V		Android Build Support	ARCore、Oculus、OpenXR
Apple Vision Pro	V			visionOS Support、iOS Build Support	ARKit
HTC Vive Focus 3	V			Android Build Support	Vive wave XR、
					OpenXR
HTC Vive Pro 2	V			PC、Mac、Linux スタンドアロン	Vive wave XR、
					OpenXR
PlayStation®VR2	\checkmark			Sony 開発者登録が必要	
Android モバイル デバイス			V	Android Build Support	ARCore
Apple iPhone および iPad (A9 プロセッサー以降)			V	iOS Build Support	ARKit
Microsoft HoloLens			V	ユニバーサル Windows プラットフォーム (UWP)	Windows Mixed Reality XR プラ グイン
Magic Leap 2			V	Android Build Support	Magic Leap XR プラグイン
Meta Quest 3	\checkmark	V		Android Build Support	ARCore、Oculus、OpenXR

Unity がサポートするターゲットプラットフォームごとの、ダウンロードする必要がある対応プラットフォームビルドモジュールと XR プラグイン

エディターをインストールすると、インストールにモジュールを追加するように求められます。Meta Quest 2 などのモバイル VR プラットフォームをサポートするには (ターゲットモバイルデバイスに応じて)、Android Build Support、OpenJDK、Android SDK および NDK、iOS Build Support および visionOS Build Support の各モジュールが必要です。

Add modules for Unity 2022.3.23f1 Silicon LTS			
Add modules	Required: 0 by	rtes Available: 1.57 TB	
✓ PLATFORMS	DOWNLOAD SIZE	SIZE ON DISK	
Android Build Support	Installed	2.02 GB	
└─ OpenJDK	Installed	219.56 MB	
└── Android SDK & NDK Tools	Installed	4.59 GB	
iOS Build Support	Installed	1.55 GB	
visionOS Build Support	Installed	816.11 MB	
tvOS Build Support	525.44 MB	1.54 GB	
Linux Build Support (IL2CPP)	76.38 MB	214.9 MB	

VR コアのサンプル

新しい VR プロジェクトを開始するには、Unity Hub で入手できる VR テンプレートを使用します。テンプレートにはサンプルシーン、アセット、および推奨されるパッケージと設定が用意されています。テンプレートを取得するには:

- 1. Unity Hub を開き、青い New Project ボタンをクリックします。
- 2. エディターのバージョンを、利用可能な最新の Unity 2022 LTS バージョンに変更します。
- 3. テンプレート リストで、**VR Core** テンプレートを探します。**Download template** ボタンを クリックします。
- プロジェクトの名前を選択し、Project Settings でその場所を選択します。Unity が開き、 開始画面が表示されます。



Unity Hub の VR Core テンプレート

バージョン 3.0 へのアップデート

デフォルトでは、XRI の 2.5.x バージョンは、Unity Hub からのこのテンプレートに基づいて新規プロジェクト を開始するときに使用されている可能性があることに注意してください (このブックのリリース時点)。すでに バージョン 3.0 の場合は、このセクションをスキップできます。Package Manager からバージョン 3.0 に アップデートすることをお勧めします。このアップデートには、visionOS との互換性など、一連の重要な 改善点と機能が盛り込まれています。XRI の新しい Near-Far および Poke (ポーク、つつく) インタラクターと 統合された visionOS での Shared Space (共有空間) 入力の処理に関するベストプラクティスを示して います。これらのアップデートの詳細については、ドキュメントのこのページを参照してください。



XRI 3.0 のサンプルに含まれる visionOS ジェスチャの例

バージョン 3.0 をインストールする前に、このドキュメントのページで説明されているように、バージョン 2.5.x に属する Starter Asset フォルダーを削除して競合を回避してください。



XRI のバージョン 3.0 にアップデートする前に、Starter Assets フォルダーを削除してください

アップグレードオプションを使用できない場合は、手動で XRI 3.0 をインストールできるようになりました。 ドキュメントで説明されているように、名前でパッケージを追加し、そのバージョンが機能することを示します。



XRI パッケージとサンプルの手動アップデート

バージョン 3.0 の XRI をインストールすると、Package Manager から新しいバージョンの Starter Assets サンプルを利用できるようになります。

 # Scene ∞ Game ■ Package Manager + ▼ Packages: In Project ▼ Sort: Name (asc) ▼ All Services 	Filters 🔻	Clear Filters	: ×
 ✓ Packages - Unity △ Oculus XR Plugin ○ OpenXR Plugin ⑦ XR Core Utilities XR Interaction Toolkit ⑦ XR Legacy Input Helpers 	4.2.0 ⊗ 1.10.0 ⊗ 2.3.0 ⊗ 3.0.3 ⊘ 2.1.10 ⊗	XR Interaction Toolkit Remove Image: Constraint of the second	Î
XR Plugin Management	4.4.0 🕜	Starter Assets 8.82 MB ⊘ Import Assets to streamline setup of behaviors, including a default set of input actions and presets for use with XR Interaction Toolkit behaviors that use the Input System. Dependencies: • Shader Graph Hands Interaction Demo 2.96 MB Import Sample scene and other assets for hand-tracking integration with Import	

バージョン 3.0 の Starter Assets のインポート

変更がある API を使用してスクリプトを更新するよう求めるプロンプトが表示された場合は、Yes をクリックします。

プロジェクトを表示する準備が整い、Welcome to the VR Template Project ウィンドウが表示されます。 表示されない場合は、Toolbar > Tutorials > Welcome ダイアログから開きます。

2 Project Settings XR Plug- in Management Project Settings XR Plug- in Management Project Settings Plug- in Settings Project Settings Plug- in Providers Plug- in Providers Plug- in Providers Plug- in Providers Plug- in Providers Plug- in Providers Plug- in Providers Plug- in Management Plug- in Management Plug- in Providers Plug- in Management Plug- in Management Plug- in Management and select the platform(s) Plug- in One deploy to. If you don't see your platform listed, make sure you have the right build targets installed in Unity Hub. Please refer to the Quick Start Guide for more detailed information on the content and settings used in this target but
2 Welcome to the VR Template Project! 2 Defore you begin, go to Edit > Project Settings > XR Plug-in Management and select the platform(s) you plan to deploy to. If you don't see your platform listed, make sure you have the right build targets installed in Unity Hub.
this template.

開始ウィンドウでは、XR Plug-in Management 設定にアクセスして展開先のプラットフォームを選択する ようプロンプトが表示されます。XR Plug-in Management の設定は、Project Settings 内にあります。 使用しているプラットフォームが表示されない場合は、ハブに戻り、追加モジュールをインストールして ください。XR Plug-in Management の設定の詳細については、次のセクションで説明します。

VR Template Project の開始画面



Tutorials ウィンドウ

Tutorials ウィンドウも表示され、サンプルプロジェクトの詳細が示されます。



Tutorial ウィンドウ

サンプルシーン

VR テンプレートに含まれるサンプルシーンは、インタラクションとナビゲーションに必須の要素を使用して 設定されています。



VR サンプルシーンのテスト

エディターで再生モードに入り、シーンの機能をテストします。コントローラーには、シーン内のアイテムの 移動や操作に役立つ UI ツールチップが含まれていることが分かります。ヘッドセットがない場合については、 この e ブックの後半の XR Device Simulator のセクションで、シミュレーターを使用してヘッドセットなしで テストする方法を示しています。



追跡されたコントローラーとツールチップ



VR コントローラーを使用してさまざまな UI 要素を操作します。

VR での UI とのインタラクション


シーン内のさまざまなオブジェクトを取り上げて扱います。

遠隔グラブを使用してオブジェクトを掴みます

シーンには、ゲーム開発作業の開始に役立つ数多くの機能が含まれています。

サンプルシーン設定

サンプルシーンの主なコンポーネントをいくつか見てみましょう。



XRI 2.5.x のサンプルシーンの Hierarchy にあるプレハブバリアント Complete XR Origin Setup Variant

Hierarchy には、バージョン 2.5.x の Complete XR Origin Set up Variant と呼ばれるプレハブ バリ アントがあります。このバリアントには、VR 世界での移動や操作の開始に必要なすべてのコンポーネントが 含まれています。XRI 3.0.3 にアップデートしたら、そのプレハブを削除し、3.0.3 の Starter Assets から XR Origin (XR Rig) プレハブを追加する必要があります。 これには、オブジェクトを掴む操作、UI コンポーネントの操作、ロコモーションテクニック (テレポート、 スナップターン、コントローラーのサムスティックによる移動など) を使用したナビゲーションを含む、完全に 機能するユーザーインタラクションに必要なすべての設定が含まれています。



XRI 3.0.3 のサンプルシーンの Hierarchy にプレハブバリアント XR Origin (XR Rig) を手動で追加する必要があります

さらに、プレハブには次のようなコア要素がすべて含まれています。

- Input Action Manager: このクラスは、XR 環境内での入力アクションの有効化と無効化を管理 します。
- XR Interaction Manager: XR コントローラ、インタラクタブルオブジェクト、環境のそれぞれの間の インタラクションを管理します。
- XR Origin: XR セッションの空間原点を管理します。これは、物理的な動きと向きを仮想世界に変換 するうえで重要な役割を果たします。これには、カメラ、左右のコントローラートラッキング、移動設定が 含まれます。

これらのコンポーネントの詳細については、次のセクションで説明します。

Hierarchy には、Interactables と呼ばれる親ゲームオブジェクトも含まれます。このオブジェクトには、 さまざまな方法で操作できるすべてのゲームオブジェクトが含まれます。

例えば、スフィアなど、通常の掴むことが可能なオブジェクトはいくつかあります。これは、XR Grab Interactable という XR Interaction Toolkit のコンポーネントによって可能となります。



XR Grab Interactable コンポーネントで掴むことが可能なオブジェクト

インタラクタブル要素は、VR にて両手で掴んだときに拡大/縮小することもできます。これを可能にするのが、 XR General Grab Transformer スクリプトです。

	Tag Untagged Layer Default	*
	Transform	0 ‡ :
z • 7	P Sphere 1 (Mesh Filter)	0 ‡ i
	▶ 🗒 🗸 Mesh Renderer	0 ‡ ;
< Persp	▶ 😔 🗸 Sphere Collider	0 ⊉ :
	🕨 😌 Rigidbody	07:
	🕨 🗰 🖌 XR Grab Interactable	0 ‡ :
	🚽 🔻 🔽 XR General Grab Transformer	07:
	Script	C
	Translation Constraints Regmitted Displacement Avec Eventhing	
	Constrained Avis Displacement Mode Object Polative With Looked World Lin	
	Constrained Axis Displacement Mode Object Relative With Locked World Op	
	Rotation Constraints	
	Two Handed Rotation Mode First Hand Directed Towards Second Hand	
	Scaling Constraints	
	Allow One Handed Scaling 🗸	
	Allow Two Handed Scaling 🗸	
	One Handed Scale Speed	0.5
	Threshold Move Ratio For Scale	0.01
	Clamp Scaling 🗸	
	Minimum Scale Ratio 🛛 🔍 🔍	0.25
	Maximum Scale Ratio —	2
	A I Scale Multiplier	0.25
a 🛛 🖉 🏕 🖉 🕁 🖈	29 29	0.25
	Torus (Material)	0 ‡
	Shader Shader Graphs/InteractablePrimitive	Edit 📜 🛨
• # #	Add Component	
cy Anderge of an Anderse of An		

XR General Grab Transformer はオブジェクト操作を可能にします。

プロジェクトには BasicScene という別のシーンがあり、ここには環境とインタラクションを追加するための設定プレハブが含まれています。



コントロールがすでに設定されている環境を追加する準備ができている BasicScene。XRI 3.0.3 を使用している場合は、Complete XR Origin Set Up を削除 し、XR Origin (XR Rig) を追加してください。

次のチュートリアルに従ってサンプルを設定できます。



プロジェクトテンプレート

さまざまな機能の使用方法に関するその他の例は、GitHub からダウンロードして見ることができます。

- XR Interaction Toolkit Examples
- Mixed Reality (MR) Example for Meta-OpenXR
- XR Simulation Environments

URP を使用した新しい VR プロジェクトの作成

VR Core テンプレートの設定手順に従えば、空の新しい VR プロジェクトを開始して、主なコンポーネントの 詳細を確認する準備は完了です。

3D URP テンプレートで開始

新しいプロジェクトを開始するには、Unity Hub を開き、New Project を選択して、最新バージョンの Unity 2022 LTS を選択します。テンプレートリストから下へスクロールして 3D (URP) をダウンロードします。

• • •	New project Editor Version: 2022.3.19f	11 LTS 0
i All templates	Core	٥
 Core Sample 	Core	• SRP
Learning	3D (URP) Core	
	SEP 3D (HDRP) Core	3D (URP) The URP (Universal Render Pipeline) blank template includes the settings and assets you need to start creating with URP
	AR Mobile Core	Read more PROJECT SETTINGS
	Small Scale Competitive Multiplayer Core	Project name My project
		Cancel Create project
_		

Unity Hub の 3D URP テンプレートを選択します

プロジェクト設定を入力し、Create Project をクリックします。

エディターで Inspector ウィンドウが開き、URP ドキュメント、フォーラム、およびバグ報告へのリンクが 表示されます。

XR Plug-in Management

XR Plug-in Management パッケージは、Project Settings ウィンドウから入手可能で、XR プラグインの 統合と設定を管理します。これには VR、AR、MR の各デバイスの設定が含まれます。これにより、プロジェクト が各種ハードウェアのサポートに使用する適切な XR SDK とプラグインを選択して設定できます。基本的には、 Unity プロジェクトがさまざまな XR テクノロジーと相互作用する方法を設定してカスタマイズするための 一元的なハブとして機能します。

Project Settings ウィンドウに移動し、リストから **XR Plug-in Management** パッケージを選択して、バーを クリックしてインストールします。



XR Plug-in Management パッケージのインストール

これで、ビルドプラットフォームとプラグインプロバイダーを選択できます。必要なビルドモジュールがインストールされていることを確認します。使用しているプラットフォームが表示されない場合は、Unity Hub にアクセスしてインストールしたものを探します。ここで歯車アイコンをクリックし、Add Modules をクリックしてモジュールを追加できます。

	Add modules for Unity 2022.3.22f1 us			
	Add modules	Required: 0	bytes Available: 1.4 TB	
	Microsoft Visual Studio Community 2022	Installed	1.59 GB	
	Android Build Support	Installed	2.07 GB	
	L OpenJDK	Installed	222.86 MB	
	└─ Android SDK & NDK Tools	Installed	3.03 GB	
	OS Build Support	351.9 MB	1.59 GB	
	tvOS Build Support	350.01 MB	1.58 GB	
	visionOS Build Support	413 MB	1.97 GB	
	Linux Build Support (IL2CPP)	53.01 MB	219.92 MB	
	Linux Build Support (Mono)	52.54 MB	216.65 MB	
	Linux Dedicated Server Build Support	100.29 MB	403.92 MB	
	Mac Build Support (Mono)	346.53 MB	1.9 GB	
	Mac Dedicated Server Build Support	344.24 MB	1.88 GB	

Hub からのビルドモジュールの追加

エディターに戻ると、使用しているプラットフォームがリストに表示されています。各プラットフォームのプラ グインプロバイダーを選択します。この例で選択したのは、OpenXRです。

		Project	Settings				
🗘 Project Settings							
				٩			
Editor ▼ Graphics URP Global Settings	XR Plug-in M	lanagem	ent				
In-Editor Tutorials	모				5		
Input System Package	Initialize XR on Start	p 🗸					
Memory Settings Package Manager	Plug-in Providers €	•					
Physics							
Physics 2D Player	🗸 OpenXR 🚱 🛕						
Preset Manager	Unity Mock HM						
Quality Scene Template							
Script Execution Order							
Services ShaderGraph	Information about configuration, tracking and migration can be found below.						
Tags and Layers							
▼ TextMesh Pro							
Time							
Timeline							
UI Builder							
Unity Version Control Settings							
Visual Scripting							
XR Plug-in Management							
OpenXR							
Project Validation							
XR Interaction Toolkit 🛛 👻							

ビルドターゲットとして OpenXR を選択

注:この入力システムを有効にするには、エディターの再起動を求めるプロンプトが表示されることがあります。エディターの再起動時に、OpenXR オプションが選択されていない場合は、ここで選択できます。

インタラクションプロファイルの追加

XR Plug-in Management パッケージのインタラクションプロファイルを使用してプリセットを設定し、 ゲームがターゲットデバイスとのインタラクション方法を理解できるようにします。例えば、ゲームが Meta Quest をサポートしている場合は、Oculus Touch コントローラープロファイルを選択します。複数のデバ イスをターゲットにしている場合は、複数のプロファイルを割り当てることができます。

ひとたび XR Plug-in Management パッケージがインストールされ OpenXR に設定されると、インタラク ションプロファイルを OpenXR に割り当てることができてアプリケーションの実行時にコントローラーが 追跡されるようになります。これはターゲットハードウェアによって異なります。

Enabled Interaction Profiles セクションで、+ アイコンをクリックし、リストからコントローラーの種類を 選択します。



インタラクションプロファイルを選択する

Project Validation タブで警告がないことを確認します。これは、左側の列で Project Validation を選択する ことで素早く行えます。すべての問題がここに表示されます。解決するには Fix All をクリックしてください。

Project Validation

次のステップは、XR Interaction Toolkit (XRI) パッケージのインストールです。

XR Interaction Toolkit の実装

XRI Toolkit の概要についてはこのガイドの前半を参照してください。それでは、その実装方法を見ていきましょう。

Window > Package Manager で、Unity Registry のドロップダウンから XR Interaction Toolkit を 選択し、Install をクリックします。Input System バックエンドを有効にするかどうかを確認するメッセージが 表示されたら、**Yes** をクリックします。エディターが再起動します。



XRI Toolkit パッケージのインストール

Starter Assets を使用した XRI の探求

次のセクションの手順は、パッケージに含まれる Starter Assets の使用に基づいています。これは、動作、 入力アクション、XR Interaction Toolkit の機能を紹介するデモシーンの設定を合理化する効率的な方法です。

このアプローチに興味をお持ちの方は、**Samples** をクリックし、Starter Assets サンプルをプロジェクトに インポートしてください。

					3
	Package Manager				
					٩
	All Services	11.22 ×			
	TextMathDro	208	XR Interaction Toolkit	Update to 3.0.1 Remove	
e nt	Timeline	176 ¥	0 2.5.3 - February 29, 2024 Release		
			From Unity Registry by Unity Technologies Inc.		
	Toolchain MacOS Linux x64				
	Tutorial Authorian Tarla		Resultation Version History, Rener January Complex		
	Tutorial Authoring Tools		Description Version History Dependencies Samples		
	Lutorial Framework		Starter Assets 2.5 MB 🗸	Reimport	
	Unity Denoising		Assets to streamline setup of behaviors, including a default set of		
	Unity Distribution Portai		nput actions and presets for use with XR Interaction Toolkit		
	Unity Logging				æ
	Unity OpenXR Meta		Hands Interaction Demo 3.04 M8	Import	Mac, Linux
	Unity Physics	1.0.16			
	Unity Profiling Core API		the XR Interaction Toolkit. Dependencies:		esentation
	Unity Transport				
			• Snader Graph		
			AR Starter Assets 3 MB	Import	
			Assets to streamline setup when building for Mobile AR or Mixed		
			Reality with the XR Interaction Toolkit. This includes prefabs and demo scenes to get you started. Note: This depends on the Starter		
			Assets sample package.		for Windows
					for Linux
			XR Device Simulator 1.03 MB 🗸	Reimport	*
Assets Sam		1.9.2 🕤	Assets related to the simulation of XR HMD and controllers.		
Scer			Meta Osta Adoptor 1/ 07 VB	Impart	
Setti			This sample provides an example to get eve tracking working with	import	
NR XR			the Meta Quest Pro and the XRI Gaze Interactor.		reshold
Rear			Oculus Integration Assets		
G Univ		2.5.3 😯			3
					oding
					ig itv

XR Interaction Toolkit Starter Assets

デモシーン

このセクションでは、XRI Toolkit のデモシーンに含まれる各プレハブとコンポーネントの役割について 説明します。見つからない場合は、前のセクションで説明したようにサンプルがインストールされている ことを確認してください。

Starter Assets のサンプルには、サンプルフォルダー > XR Interaction Toolkit > (ツールキット バージョン) > Starter Assets > DemoScene でデモシーンが含まれています。

🖿 Project	👭 Audio Mixer	E Console	🕒 Animation	🕲 Unity Version Control
+-				
🕨 ★ Favorit	es	<u>^</u>	Assets > Samples	> XR Interaction Toolkit > 3.0.0-pre.1 > Starter Assets
▼ 🕞 Assets ▼ 🍃 Samp ▼ 🏳 XR ▼ 🏳 3	les Interaction Toolkit 3.0.0-pre.1 Starter Assets Animations DemoSceneAss	ets	 Animations DemoScene Editor Filters Materials Models Prefabs Presets 	Assets
•	 Editor Filters Materials Models 		 Scripts Shaders Tunneling Vi 	gnette
Scene	 De Prefabs Presets Scripts Shaders Tunneling Vigness 	ette	 Definitiscene Unity.XR.Inte W XRI Default I 	eraction.Toolkit.Samples.StarterAssets nput Actions
Settin	as		🕄 Assets/Samples	XR Interaction Toolkit/3.0.0-pre.1/Starter Assets/DemoScene.unity

このデモシーンには、独自の VR ゲームの制作を開始するために必要なすべてのものが揃っています。

- ー ナビゲーション
- Grab インタラクションとさまざまな動作タイプ
- ー Poke インタラクション
- Gaze インタラクション
- Climb インタラクション
- UI インタラクション
- ー クライミング



XR Interaction Toolkit のデモシーン



Starter Assets デモシーンでの XR インタラクション設定

XRI 3.0 のデモシーンには、Input Action Manager コンポーネントと XR Interaction Manager ゲーム オブジェクトを含む XR Origin (XR Rig) プレハブが入っています。



XR Origin (XR Rig) プレハブ

Input Action Manager

Input Action Manager は、XR 環境内での入力アクションの有効化と無効化を管理します。 InputActionAsset オブジェクトのリストで動作し、さまざまなアセットの入力アクションを一元管理できます。 アクションが必要なときに反応を示し、そうでないときは休止状態になります。これは、動的な入力処理が ユーザーインタラクションとユーザー体験にとって不可欠な VR/AR シナリオにおいて特に便利です。

XR Interaction Manager

このマネージャーは、インタラクター (VR ハンドコントローラー、ポインターなど) とインタラクタブル (VR/ AR 空間内の相互作用可能なオブジェクト。プレイヤーが手に取ることのできるボタンや小道具など) の間の インタラクションを管理するための中央ハブとして機能します。これにより、インタラクターとインタラクタ ブルが有効になっているときに、それぞれが Interaction Manager に登録され、そのシーンにおいて正しい コミュニケーションと機能が可能になります。詳細については、XR Interaction Toolkit の Glossary を 参照してください。

XR Origin

XR Origin プレハブは、XR セッションの空間原点を管理します。XR デバイスに関連するカメラの位置決めを 処理し、"フロア"レベルや"デバイス"レベルなどのさまざまなトラッキングモードをサポートします。これは、 身体の動きや向きを仮想世界に変換する重要な役割を果たします。Unity で XR 環境を設定し、仮想空間内 での正確かつ応答性に優れたインタラクションを実現するうえでこれは不可欠です。 XR Origin プレハブには、Character Controller と XR Input Modality Manager が含まれています。



XR Origin コンポーネント

Character Controller Driver

Character Controller Driver は、プレイヤーの VR ヘッドセットの現実世界での動きに合わせて仮想キャ ラクターの高さと位置を調整するよう設計されています。プレイヤーが VR 空間内で移動を開始したり停止 したりするタイミングを監視し、移動と衝突を処理する Character Controller コンポーネントを更新します。



XR Input Modality Manager

XR Input Modality Manager は、トラッキングステータスに基づいてランタイム中のハンドトラッキングと モーションコントローラー間の動的な切り替えを管理します。手とコントローラーを表すさまざまなゲーム オブジェクトのセットを切り替えて、適切な入力方法がアクティブになるようにします。ハンドトラッキングが 開始されると、インタラクターのハンドグループに切り替わります。モーションコントローラーが使用されて 追跡されると、モーションコントローラーグループがアクティブになります。このコンポーネントは、ハンド トラッキングを使用していないプロジェクトでも便利です。ハンドトラッキングを使用すると、追跡されて いないときにコントローラーをデフォルト位置に表示できなくなるからです。

XR Origin プレハブには、次の 2 つの子ゲームオブジェクトがあります。メインカメラを有し、コントローラーを 追跡する機能を持つ Camera Offset、および Locomotion System です。

TrackedPoseDriver (Input System)

メインカメラには、TrackedPoseDriver (Input System) コンポーネントがあります。これは、リアルタイム でのユーザーの頭のトラッキングなど、現実世界の物理的な動きをバーチャルな動きにマッピングして、その 動きを VR に変換できるようにする重要な役割を果たします。

	volume ringger	whole (mansionil)		`	2
	▶ Output			9	
a :	🎧 🗹 Audio Listener		9 -	ż	
🕫 🕛 ★ 👁 29	🔻 🤟 🖌 Tracked Pose Driver (Input	System)	9 7	ż	
	Script	Second PoseDriver			
	Tracking Type	Rotation And Position			•
	Update Type	Update And Before Render			•
VDL Llocat/	Ignore Tracking State				
XRI Head/	Position Input	: 🚯 XRI Head/Position (Input Action Refere	ence)		0
	Rotation Input	: 🚱 XRI Head/Rotation (Input Action Refer	ence))	0
- (4) (Tracking State Input	SARI Head/Tracking State (Input Action	Refe	ere	0
	🔻 🐞 🛛 Universal Additional Camer	a Data (Script)	0.	ż	
XRI Left In					
		Add Component			

メインカメラの Tracked Pose Driver (Input System) コンポーネント

入力は、XRI Toolkit によって提供される入力アクションにマップされ、すぐに使用できます。

Left Controller および Right Controller の設定も Camera Offset に含まれています。これらのオブジェ クトは両方とも、Controller Input Action Manager と XR Interaction Group コンポーネント 同じコンポーネントを持っています。



コントローラー用のコンポーネント

Controller Input Action Manager

Controller Input Action Manager は、左右のコントローラーにあるコンポーネントです。VR コントロー ラーがオブジェクトと相互作用し、VR アプリケーションでのユーザー入力に応答する方法を管理するのに 役立ちます。レイで遠くにあるオブジェクトを指差したり、テレポートしたりなど、さまざまな移動方法とイン タラクション方法を切り替えることができます。

プレイヤーの行動に応じて、常に適切なアクション (移動やテレポートなど)のみが可能です。

XR Interaction Group コンポーネント

XR Interaction Group Component は、対話的要素のグループを管理するためのシステムとして機能し、 グループ内で一度にアクティブにできる対話的要素またはインタラクターは1つだけです。

XR Transform Stabilizer

XR Origin プレハブ内の最後のオブジェクトセットは、Left/Right Controller Stabilized Origin ゲーム オブジェクトです。どちらにも XRTransformStabilizer スクリプトが添付されており、XR 環境をより 精密に感じさせ、意図しない小さな動きの影響を受けにくくするのに役立ちます。Left Controller および Right Controller の変換をターゲットオブジェクトとして割り当てると動きが滑らかになり、トラッキング データのブレや不安定な動きを抑えます。

インタラクションの設定

テレポートする、掴む、つつく、相互作用することを制御するシステムはすべて Left Controller および Right Controller に含まれています。



Left Controller および Right Controller に含まれるインタラクション設定

これらのインタラクターにはそれぞれ対応するコンポーネントが設定されています。例えば、プレイヤーが オブジェクトを選択できる Poke Interactor ゲームオブジェクトには、XR Poke Interactor というコン ポーネントがあります。

XR Direct Interactor

XR Direct Interactor コンポーネントは、ボタン、レバー、その他多くのオブジェクトや小道具など、プレイヤーが 触れたり、動かしたり、掴んだりできるインタラクタブル要素との直接的なインタラクションを可能にします。

Poke (つつく)、Ray (レイ)、Teleport (テレポート) の各インタラクターは、すべて VR 世界内での特定の インタラクションを可能にします。この e ブックの後半ではさまざまなインタラクタブル要素を見ていきます。

Locomotion System

XR Origin プレハブへのアクセスを管理 および 制御します。XR Origin プレハブは、仮想空間の参照位置 (通常はプレイヤーやユーザーがいる場所)を表します。これにより、常に XR Origin を移動できるのは 1 つの Locomotion プロバイダーのみとなり、競合を防止し、XR アプリケーションでのスムーズで予測可能な移動 体験を実現します。

プレイヤーの位置の操作方法には、回転、移動、テレポーテーション、およびクライミングがあります。

回転

XRI Toolkit には、プレイヤーが回転するための 2 つのコンポーネントが用意されています (Snap Turn と Continuous Turn)。Snap Turn (スナップターン) では、プレイヤーが一定角度の単位で回転でき、 Snap Turn Provider コンポーネントでカスタマイズできます。Continuous Turn (連続ターン) は、 Continuous Turn Provider コンポーネントで設定された可変スピードで滑らかに回転します。どちらの コンポーネントも Unity の Input System と統合されており、使いやすくなっています。ツールキットには 入力アクションがあらかじめ設定されているため、開発が簡単になり、最小限の設定でこれらの回転の コンポーネントをすぐに使用できます。

移動

DynamicMoveProvider は、継続的な移動制御のバリアントであり、ユーザー設定に基づいて移動方向の 基準フレームを調整するよう設計されています。フォワード方向を制御でき、ユーザーは左右の手に応じて 頭追従または手追従を選択できます。これにより、XR 環境でのカスタマイズ可能で直感的な移動体験が 提供されます。

Teleportation

XRI Toolkit の Teleportation により、プレイヤーは仮想環境内の新しい場所にすぐに移動できます。これは、 Teleportation Anchor と呼ばれるターゲット位置を目指し、テレポートアクションを開始することで実現 できます。テレポーテーションシステムには、テレポートのパスと移動先を示す円弧や線などの視覚的な インジケーターが含まれています。これは、より大きな仮想空間を快適に移動するための重要な特徴で、 特に連続的な移動を不快に感じる可能性があるシナリオの場合や、スペースの制約により連続的な移動が 実行不可能なシナリオで効果を発揮します。

Teleportation Provider コンポーネントは XR 環境でのテレポーテーションを処理します。ユーザーの 要求に応じて、XR Origin を指定された場所に移動させます。重要な機能として、テレポーテーションの遅延 時間設定があります。この設定は、トンネルビネット効果などの快適設定に使用できます。また、world space up、target up、target up and forward などによって、移動先のさまざまなタイプの向きに合うようにサポート します。プロバイダーが、テレポート要求をキューに入れ、これらの要求の適用 (向きや位置の変換など) を 管理することによって、目的のテレポーテーション効果が得られます。



アンカーポイントを使用したテレポート

クライミング

この ClimbProvider コンポーネントは、XR 環境でのクライミングメカニクスを容易にします。ユーザーは ClimbInteractable オブジェクトを操作できるようになり、ユーザーのアクションに応じて XR Origin を 移動できるようになります。クライミングモーションはユーザーのインタラクターの動きによって相殺され、 インタラクタブル要素の各軸に沿って制限できます。

Tunneling Vignette

サンプルシーンには、Main Camera の子である TunnellingVignette プレハブも含まれます。シーン内を 移動すると、移動中に黒い境界線がフェードインおよびフェードアウトします。この効果は VR ビネッティング または動的視野 (field-of-view、FOV) 削減と呼ばれ、乗り物酔いのような症状を減らすことができます。 これは周辺視野を減らすことによって、ユーザーの注意を画面の中心に集中させます。周辺の動きを減らす ことで、視覚への入力と前庭器官への入力の差異を減らし、乗り物酔いのような症状がでる可能性を減ら します。



サンプルシーンで機能する TunnellingVignette 効果

VR 世界との相互作用

VR ゲームを制作するうえで不可欠なことは、プレイヤーがゲーム内のアイテムを操作できるようにする ことです。

このデモシーンでは、オブジェクトの選び方が異なる 3 種類の Grab (掴む) インタラクタブル要素を紹介して います。インタラクタブルオブジェクトには XRGrabInteractable コンポーネントが必要です。



サンプルシーン内の 3 つの Grab (掴む) インタラクタブル要素。

XRGrabInteractable

XRGrabInteractable コンポーネントは、VR/AR 環境での基本的な Grab (掴む) 機能を有効にします。 このコンポーネントを持つオブジェクトをインタラクター (VR ハンドコントローラーなど) が掴んだり選択 したりして、その動きを追従させることができます。

XRGrabInteractable コンポーネントの機能の1つに、追跡、つまり Movement Types があります。

- Kinematic:オブジェクトは物理的な力と衝突を回避して直接インタラクターとともに移動し、その 位置と回転は正確に制御されます。これは、予測可能でスクリプト化されたインタラクションに最適 です。
- Instantaneous:オブジェクトの位置と回転は、物理演算を行わずにインタラクターに合わせて即座に 更新され、応答性に優れた直接的な制御を実現します。
- Velocity Tracking:インタラクターの動きに基づいてオブジェクトの速度が動的に更新されることで 動作が生じ、リアルな物理的現象のインタラクションと、慣性や惰力などの自然な動作を可能にします。

Poke インタラクタブル

デモシーンには、Poke (つつく)機能を使用する以下の 3 つの異なるインタラクションをテストするための 指定エリアも含まれます。パーティクルシステムの有効化、サウンドの再生、UI テキストのインクリメント。



Poke インタラクタブルタイプのサンプルシーン

XR Simple Interactor

最終的には、ボタンを押すかつつくことで、何でもできるようになります。Poke インタラクタブル要素の ルートゲームオブジェクトには、XR Simple Interactable コンポーネントが含まれています。これは XRBaseInteractable クラスを拡張し、そのプロパティとメソッドを継承します。XRSimpleInteractor クラスは、さまざまなインタラクション状態に関するイベントを提供することに焦点を当てています。これらの イベントにより、開発者は Poke が検出されたときのオブジェクトの応答をカスタマイズできます。

XR Poke Filter

XR Poke Filter は、VR/AR 環境のオブジェクトに Poke インタラクションを追加するコンポーネントです。それ 自体を XRBaseInteractable に関連付けることで機能します。デモシーンでは、これが XRSimpleInteractor です。さらに、これらのカスタム設定に基づいて、オブジェクトがつつかれたことに対する応答の必要があるか どうか、およびその方法を判別する機能を提供します。このクラスには、初期化、設定、およびクリーンアップの ためのさまざまなライフサイクルメソッドのほか、エディター内でのインテグレーションとデバッグを改善する ための機能が含まれます。XR Poke Filter は、微妙な Poke インタラクションを促進することで、オブジェクトの インタラクティビティと現実感を強化する重要な役割を果たします。 **XR Poke Follow Affordance**

XRI Toolkit の XRPokeFollowAffordance クラスは、VR や AR の設定でボタンなどをつついたり操作したり するときに UI 要素をアニメーション化するために使用します。重要な機能として、滑らかな動き、インタラク ション後に元の位置に戻る機能、リアルなインタラクション体験を実現するための移動距離の制約などが 挙げられます。

Gaze インタラクタブル

XRI Toolkit のコンテキストでの Gaze (視線) インタラクタブルとは、ユーザーが仮想環境内のオブジェクトを 見るだけで操作できる一種のインタラクション手法のことです。これは、ハンドコントローラーやタッチなどの 従来の入力方法が効果的でなかったり利用できなかったりする VR/AR アプリケーションで特に便利です。 また、身体の不自由なユーザーでも VR/AR コンテンツを操作できるため、アクセシビリティの観点からも 優れています。





デモシーンでの Gaze (視線) インタラクタブル設定

シーンでこれらのインタラクタブル要素の1つをクリックすると、その機能がXRSimpleInteractorコン ポーネントによって操作されることが示されます。このコンポーネントのイベントは、ユーザーの視線に 合わせて機能を作成するために使用されています。

視線機能は、XR Origin 内の Gaze Interactor オブジェクトに設定された以下の 3 つのコンポーネントに よって主に働きます。XR Gaze Interactor、Gaze Input Manager、Tracked Pose Driver (Input System)

XR Gaze Interactor

XR Gaze Interactor は、XRRayInteractor スクリプトを拡張し、VR で視線を通してオブジェクトと相互作用 できるようにします。スクリプトにはカスタマイズ可能な視線補助機能が搭載されており、インタラクション領域 を固定サイズまたはコライダーベースの動的なサイズに設定できます。これには、Gaze (視線) インタラク ションの感度と精度を調整するプロパティも含まれます。視線補助のための "スナップボリューム"の自動 作成とスケーリングがサポートされ、インタラクション精度が向上します。さらに、このスクリプトはインタラク タブルターゲットの処理を管理し、さまざまなオブジェクトとのインタラクション機能をチェックするメソッドを 含んでいます。また、Gaze インタラクションのデフォルト動作 (Hover to Select や Auto Deselect などの時) のオーバーライドもサポートし、視線ベースの VR インタラクションのための柔軟で包括的なソリューションを 提供します。



VR Core サンプルの Gaze Interactor コンポーネント

Gaze Input Manager

GazeInputManager スクリプトは、アイトラッキングを使用できないときの XRGazeInteractor の入力 フォールバックを管理します。起動時にアイトラッキングサポートをチェックし、アイトラッキングデバイスが 検出された場合はログを記録します。そのようなデバイスが検出されない場合はデバイス接続イベントに サブスクライブし、デバイスが後で接続された場合のアイトラッキングを有効にします。スクリプトには、アイ トラッキングがない場合にヘッドトラッキングへのフォールバックを有効にするトグルも設定されています。 これにより、アイトラッキングが利用できない場合はヘッドトラッキングに切り替えることで機能を継続的に 確保し、インタラクション機能を維持できます。GazeInputManager は、視線インタラクションに使用できる 入力メソッドに適応することで、ユーザー体験を向上させます。

XRGazeInteractor と XRSimpleInteractor は連携して、遠くへの視線による選択から近くの手動操作まで、 幅広いシナリオに対応する柔軟で堅牢なインタラクションシステムを VR で提供します。

UI 要素とのインタラクション

XR 世界にアクセスしたプレイヤーは、ある段階でユーザーインターフェースを操作する必要があります。XR Interaction Toolkit のサンプルシーンは、プレイヤーの VR コントローラーからの入力に UI コンポーネントが どのように反応するかを示しています。

このデモシーンでは、UI インタラクションの例を3つ紹介します。UI 機能の作成の中核要素は、Event System ゲームオブジェクトにある XR UI Input Module コンポーネントです。



サンプルシーンの UI 要素

XR UI Input Module コンポーネント

XR UI Input Module は、Event System と正しくインターフェースするために XRI にとっては必要なコン ポーネントです。XR 環境ユーザーと UI 要素間のインタラクションを管理します。XR Ray Interactor と 連携して動作し、XRI と UI の相互作用が正しく処理されるようにします。3D 追跡デバイスや、マウス、タッチ、 ゲームパッドコントロールなどの非 XR ソース (設定されている場合) など、幅広い入力方法をサポートして います。このスクリプトは、XR インタラクターが UI 要素と相互作用し、ホバーやクリックなどのイベントを 処理するためのインターフェースを定義し、Unity の古い機能の Input Manager と新しい Input System の 両方と互換性があります。架け橋として機能し、XR 環境でのユーザーアクションを UI インタラクションに変換 します。

Tracked Device Graphic Raycaster

各キャンバスオブジェクトには、**Tracked Device Graphic Raycaster** コンポーネントがあります。Unity XRI Toolkit 用の GraphicRaycaster のカスタム実装であり、XR 環境でのレイキャスティングの処理専用に 設計されています。標準の GraphicRaycaster の機能を XR で追跡対象デバイスと連携するように拡張し、 キャンバス上の UI 要素とのインタラクションを検出できるようにします。



Unity の UI の詳細については、こちらの無料の e ブックと付属のサンブルプロジェクトを参照してください。 ダウンロード XRI Toolkit のサンプルシーンは、XR ゲーム開発の領域に挑もうとしている開発者にとって、貴重な出発点となります。このデモは本質的には試行錯誤の場であり、XR 体験に不可欠な対話的な要素、制御、メカニクスを幅広く揃えて披露しています。

このシーンを探検して、オブジェクト操作、テレポーテーション、3D 空間での UI インタラクション、コント ローラー入力管理など、さまざまな XR 機能の実装について深く理解しましょう。

このシーンの利点は、そのモジュール性と、各要素の変更、結合、拡張のしやすさにあります。制作を目指して いるのが没入型の VR ゲームでも、教育用の AR 体験でも、その中間であっても、このサンプルシーンは あなたのビジョンを実現するための踏み台となります。



このセクションでは、Unity で VR プロジェクトを開発する主な手順の一部を紹介します。各開発フェーズを 推進する意思決定と、Unity での実用的なステップとワークフローについて説明します。

また、新しいプロジェクトを開始するにあたり順を追って進めたい場合は、以下のビデオを参照してください。





全体像

ゲームのアイデアは、閃光のようにすぐに思い浮かんだり、時間をかけて、より慎重なプロセスを経て思い 浮かんだりします。いずれにしても、自分が心から熱中しているジャンルやテーマにコンセプトを固定する ことをお勧めします。



ヘッドセットを使用した Unity の初期プロトタイピング

初期のアイデア出しの段階では、アプローチを幅広く自由にしておくようにしましょう。この時点では、細かい ことにまで没入する必要はありません。アイデアの最初のひらめきが自然に形を形成できるように落ち着いて 取り組みます。これは、壮大な絵画の輪郭をスケッチするようなものと考えてください。ディテールはまだ重視 しません。代わりに、全体的なテーマ、喚起したい感情、プレイヤーに提供したいコア体験に集中しましょう。 実用性や技術的な制約に縛られることなく、自由に想像力を膨らませましょう。画期的なゲームはすべて、 頭の中で一瞬の火のまたたきとして始まります。そして今、その火で炎を燃え立たせるときです。

このセクションで開発するアイデアの例として挙げたのはホラールーム脱出ゲームで、プレイヤーは逃げる ために謎を解かなければなりません。それが概要です。それでは、もう少し掘り下げてみましょう。

あらかじめ計画を立てる

最初のアイデアの準備ができたら、次のステージでは、この大まかなコンセプトを構造化された計画に変換 します。この重要な段階では、開発プロセスの基盤を築きます。主要な手順は次のとおりです。

- 1. コアとなるゲームプレイのメカニクスを定義します。ゲームの基本的なメカニクスの詳細から開始 します。プレイヤーは VR 環境とどのように相互作用しますか?プレイヤーが実行する主なアクションは 何ですか?ホラーゲームの例では、不気味な環境を探索し、パズルを解き、脅威に対して回避したり 立ち向かったりします。また、これらのメカニクスを Unity でどのように実装するかについても考え 始める必要があります。例えば、XRI Toolkit は、XR Socket Interactor などといったパズル形成に 役立つ機能を提供します。
- 基本的なストーリーラインまたはテーマを作成します。ナラティブ要素の強いゲームではなくても、 基本的なストーリーラインやテーマを用意することで開発を導くことができます。ホラーゲームの場合、 この過程で設定 (幽霊屋敷、廃墟と化した収容施設、不気味な地下室など)、背景のストーリー、主要な キャラクターや敵を決定することがあります。
- 3. レベルデザインや環境の概要を描写します。ゲームの環境を視覚化することから始めましょう。レベルや 主要エリアの大まかなレイアウトを描きます。プレイヤーがこれらの空間をどのように移動するか、 レイアウトがゲームの雰囲気にどのように貢献できるかを検討します。
- 4. 独自のセールスポイント (USP) を確認します。制作したゲームが同ジャンルの他のゲームと異なる 点は何でしょう?独自のゲームプレイメカニクス、説得力のあるストーリー、VR テクノロジーの革新的な 使用、または特定のスタイルのアートとサウンドデザインですか?プレイヤーを引き付けて留めておく ために USP は重要です。
- 5. プロトタイプを開発します。簡単なプロトタイプから始めて、コアとなるゲームプレイメカニクスをテスト しましょう。これは複雑なものである必要はありません。ゲームのプレイ感覚を把握し、早い段階で調整 を加えることが重要です。VR でのプロトタイピングは不可欠です。通常の画面では測ることが困難な スケール、距離、インタラクションを把握できるからです。
- 6. 早期にフィードバックを収集します。友人、開発者仲間、潜在的なプレイヤーにプロトタイプを見せましょう。早期のフィードバックは、ゲームのコンセプトやメカニクスの長所と短所を特定するうえで非常に貴重です。
- 7. 開発プロセスの計画を立てるここまでのステップを完了すれば、プロジェクトの主なマイルストーンの 概要を作成する準備が整います。これには、詳細なゲームデザイン、アートとサウンドの制作、プログラ ミング、テスト、仕上げが含まれます。個人開発者または小規模なチームの場合は、現実的なタイム ラインと優先順位を設定してみてください。
- 8. 技術要件を検討します。VR/AR 開発には、乗り物酔いのような症状を防ぐための安定したフレーム レートの確保、さまざまな VR ハードウェアのための最適化、直感的な VR コントロールの実装など、 具体的な技術的考慮事項があります。
- 設計をドキュメント化します。詳細なゲームデザインドキュメント (GDD) を作成します。これには、 ゲームメカニクス、ストーリー、レベルデザイン、アートとサウンドディレクション、ユーザーインター フェースデザイン、技術要件など、あらゆるものが含まれている必要があります。
- 10. バジェットとリソースの計画を立てます。ソフトウェア、ハードウェア、フリーランサーによる追加作業など、 手元のリソースと必要なリソースを評価します。

これらのステップに従って作業することで、最初のアイデアを実行可能な計画として形にし、最終的なゲームを 開発できます。Unity の次の 2 つの e ブックでは、ゲームの計画と開発に関するより実用的なヒントや アイデアを得ることができます。



ホラーゲームの例に戻りましょう。上記のステップを経て、プレイヤーを地下室に入れる 5 分から 10 分間の VR 体験の作成を目指すことにしました。プレイヤーは 3 つのパズルを解いて、部屋の中央にある棺を開ける 鍵を手に入れる必要があります。彼らは棺を介して地下室を脱出できます。恐怖心とストレスを加えるために、 天井をゆっくりと下げ、スパイクを突出させます。そのため、プレイヤーには迅速な行動が求められます。

エンジンを始動させる

Unity のバージョンと URP

今こそ Unity でゲームの制作を開始しましょう。この例では、XR アプリケーションで最もパフォーマンスの 高いレンダーパイプラインとして Unity 2022 LTS と URP を使用しています。ターゲットハードウェアは Meta Quest 2 ヘッドセットです。

XR Plug-in Management パッケージをインストールし、すべてのプラットフォームで OpenXR を選択します。

次に、インタラクションプロファイルを割り当てます。この例では、Meta Quest 2 ヘッドセットを使用する Meta Quest Touch Pro Controller Profile を追加します。

Build Settings

Build Settings でターゲットを Android に設定し、テクスチャ圧縮を VR アプリケーションに適した効率的な 方法である 適応型スケーラブルテクスチャ圧縮 (ASTC) に変更します。これは、VR の高品質のテクスチャ 要求とデバイスのパフォーマンスの制約のバランスを取るのに役立ちます。

	Build Settings		: 🗆 ×	
	Scenes In Build			
	✓ Scenes/SampleScene		Ŭ	
			Add Open Scenes	
	Platform	🗰 Android		
	🖵 Windows, Mac, Linux 🛭 🕸 🕯		ASTC -	
	Dedicated Server	ETC2 fallback Export Project	32-bit 👻	
	📫 Android	Symlink Sources Build App Bundle (Google		
	iOS ios	Create symbols.zip Run Device	Disabled Default device Refresh	
	PS4 PS4		Patch Patch And Run	
	PS5 PS5	Patching is disabled for Relea	se builds	
				a :
oolkit > Runtime > Locomotion	Universal Windows Platform	Script Debugging Compression Method	LZ4 -	≉ % 23
De De De	▼ Asset Import Overrides Max Texture Size No Override ▼ Texture Compression No Override ▼		Learn about Unity Build Automation	
Grab SnapTurn Teleportati	Player Settings	s	witch Platform Build And Run	

ホラー VR プロジェクトの例のための Build Settings

Quality 設定

設定の最後のステップは、Project Settings ウィンドウで Quality の設定を調整することです。デフォルト として **Performant** を選択します。

1	Inspector Project Settings					
🔾 🔻 2D 🌻 🞼 🛫 💋 📰 🖛 🔂 🔻						
i v 20 v 14. v v v v v v v v v v v v v v v v v v v	Adaptive Performance Audio Burst AOT Settings Burst AOT Settings Craphics UPP Global Settings Input Manager Input System Package Memory Settings Physics 20 Physics 20 Phys	ector P Project Settings AOT Settings Cuality AOT Settings Performance AOT Settings Performant M mager P collable Settings Performant M M Manager Default Py Settings Add Goulity Level Add Goulity Level Current Active Quality Level Sarager Ascriptable Render Pipeline is in use, some settings will not be used and are hidden Mane Eresplate Render Pipeline Asset Execution Order Render Pipeline Asset Render Pipeline Asset Current Active Quality Level Sorighta Render Pipeline Asset Render Depline Asset Current CPU Usage Is crighting Curture U's None Don't Syne Is crighting Curture				
	UI Builder Version Control	Realtime GI CPU Usage	Low			
	Visual Scripting VISUAL Scripting XR Plug-in Management	Global Mipmap Limit	0: Full Resolution			
	OpenXR	Mipmap Limit Groups				
	Project Validation XR Interaction Toolkit					
F-4-4-22-23						
The the			Disabled 👻			
3 :						
22 ₽ ₽ ♥ ♥ ★ 9023		Particles				
		Terrain				
		Billboards Face Camera Position				
er Character Constraine Locomotio Locomotio		Use Legacy Details Distribution	<u> </u>			

Quality 設定のデフォルトとして Performant を選択



Performant URP Asset は、レンダリングパフォーマンスを最適化するように設定されています。ドロー コールを減らし、シェーダー実行を最適化し、ライティングとシャドウレンダリングをより適切に制御する ことで、最適化を実現します。ビジュアルに優れたアプリケーションを作成しながら、ターゲットプラット フォームで高いパフォーマンスを維持できます。これは、VR の没入感を維持し、乗り物酔いのような症状を 防ぐうえで重要な要素です。URP は スクリプタブルレンダーパイプラインであるため、プロジェクトの特定の ニーズに合わせてレンダリングプロセスを調整することもできます。

全体として、目標とする重要なゴールは、ターゲットとする低品質のデバイスでも十分に動作する見栄えの 良いゲームを作成することです。これは強力な基盤となり後の構築を支えます。

世界の構築

プロトタイプとモデリング

このフェーズでは、アートパイプラインが 3D 環境のプロトタイピングを開始します。このホラーゲームの例は、 地下室という単一の環境で行われるため、非常にシンプルです。

3D ブロックアウトを作成して、空間のスケールとパズルの配置場所を確認することから開始します。これらは 大きさを形づくるシンプルな形状であり、これをもとに下降する天井の仕組みについて考え始めることが できます。



環境のためのブロックアウト

環境に合わせて五角形の形状が選択され、地下室から続く小さなトンネルがプレイヤーの脱出ルートになります。

ProBuilder パッケージを使用することで、Unity では効率的なプロトタイピングが可能になります。プレイ する環境内で制作できるため、創造の柔軟性を最大限に高め、イテレーションサイクルを効率化できます。 シーン内のスケールを制御し続けることが重要なので、これは VR アプリケーションにおいては便利です。 ProBuilder を使ってアイデアのモックアップを作成し、ヘッドセットを装着してテストできれば、開発を スピードアップできます。

以前リンクを示したレベルデザイン e ブックには、ProBuilder ツールの使用に関する包括的なセクションが あります。また、以下のビデオチュートリアルでも役立つヒントを提供しています。



チュートリアルを見る

チュートリアルを見る

効率的なテクスチャリング

ブロックアウトをもとに、モデリングとテクスチャの追加を開始できます。ホラーゲームの例では、繰り返しが 多いため、柱、壁、出入り口で構成されるモデルの小さなキットを作成できます。これらを複製して環境を 形成することができます。また、これらを接合して1つの全体的なモデルを作成すると、下の画像に示す ように、トリムシートを使用して構造体全体にテクスチャを適用できます。



環境のトリムシート

トリムシートの領域はモデル内のエリアに適用され、テクスチャ付き地下室は次の画像のように表示されます。



テクスチャ付き 3D 環境

よく見ると、構造体のエリアがトリムシートにどのように割り当てられているかが分かります。

サンプルプロジェクトでは、引き続き、小道具、パズル、出口をすべて作成し、それらを Unity に取り込んで、 シェーダー (ここではデフォルトの URP Lit シェーダー)を割り当てます。次に、この環境のライト方法を見て いきましょう。

光あれ

ライティング方法の選び方

地下室では、光源は天窓からの光とドアの両側にある 2 つの松明です。トーチライトは静的なので、ベイク したライティング (ライティング情報が事前に計算されてテクスチャマップにベイクされたもの) に適した シーンです。

ただし、ドアや棺の蓋など、動くアイテムはベイクしたライティングに影を投影しません。静的オブジェクトに 対してベイクしたライティングが計算され、動的オブジェクトがリアルタイムのライティングを受ける混合 ライティングを試すことができます。しかし、最終的にはベイクしたライティングを採用します。棺の蓋を 開けたとき影がないことにプレイヤーがおそらく気付かないからです。このようなトレードオフの関係に ついては、経験豊富な開発者であれば実世界のプロジェクトから理解していることでしょう。 以下に、ベイクしたライティングを使用したシーンの画像を示します。



ライティングをベイクした結果

シーンは 512 × 512 の 1 つのライトマップで照らされています。シーンに適した設定を取得することは、 パフォーマンスと品質のバランスを取ることを意味します。

ライティングの最適化

地下室のような小さなシーンにおける目的は 1 つのライトマップです。ライトマップの元のサイズは 1024 で、 見栄えがしました。テストのため、それを 512 に減らし、3 つのライトマップを作成してドローコールに追加 します。ドローコールは、GPU に対するコマンドで、マテリアルやテクスチャなどの特定の設定を含むジオメ トリのバッチをレンダリングします。ドローコールごとに一定の処理オーバーヘッドが発生するため、ドロー コールの数を減らすと、特にローエンドのハードウェアでパフォーマンスが向上する可能性があります。

テクセル密度を下げることで、すべてを1つの512 ライトマップにまとめることができます。これは、ライトマップ内のピクセルと3D世界内の実際の物理的空間のピクセルの比率を示します。表面の単位面積を表すために使用されるライトマップピクセル数(テクセル数)の指標です。テクセル密度が高いほど、単位面積あたりのテクセル使用量が多くなり、その表面のライティングと影のエフェクトのディテールと解像度が向上します。逆にテクセル密度が低いと、単位面積あたりのテクセル数が減り、ライティングや影のディテールが薄れます。

テクセル密度の値を 40 から 20 に変更して、再度ベイクしましょう。今回は単一の 512 × 512 ライトマップを 使用します。



地下室環境のベイクしたライトマップデータ

ライトマップの品質は良好です。ヘッドセットでテストしても、全体的なビジュアル品質への影響は最小限です。 ここではライティングは変更せずに、後でゲームのパフォーマンスが良ければ品質を上げましょう。

Unity プロジェクトにおけるライティングの課題のトラブルシューティングについては、ブログ を参照して ください。

次に、プレイヤーがこの空間をどのように移動するかを見てみましょう。

仮想世界のナビゲーション

シーンの灯りがついたら、今度はプレイヤーがシーン内を移動できるように機能を追加します。XR Rig の デモシーンで使用したのと同じプレハブを追加します。次に、空のゲームオブジェクトと XR Interaction Manager コンポーネントを追加します。

テレポーテーションのタイプ

プレイヤーを自由にテレポートさせますか、それともテレポートアンカーを使用しますか?地下室環境では、 テレポートアンカーによってシーンの重要なエリアを特定でき、プレイヤーに対する一種のヒントとして機能 します。プレイヤーがどこかへテレポートできれば、それは重要な意味を持ちます。これは、時間が限られた ゲームの性質上、優れた解決策です。
テレポートアンカーをパズルの場所、棺、ドアに配置しましょう。



テレポートアンカーの場所

テレポートアンカーを使用すると、テレポートするときのプレイヤーの向きをカスタマイズできます。このサン プルゲームでは、プレイヤーはテレポート時にパズルなどの特定の位置において重要なオブジェクトの方を 向いている必要があります。これを実現するには、プレイヤーがテレポートアンカーの向きに合うように、 テレポーテーションの設定を Target Up And Forward に設定します。

コンポーネントのスタイルは簡単にカスタマイズできます。例えば、テレポーターのレイスタイルを変更する には、XR Origin の Right Controller で **Teleport Interactor** を選択します。XR Ray Interactor の **Raycast Configuration** では線の種類を変更でき、**XR Interactor Line Visual** では色を制御できます。

XR Interaction Toolkit を使用してパズルを作成

以下のホラーゲームの例におけるシーンでは、2 つの魅力的なパズルがプレイヤーに提示され、それぞれ クリア時の報酬として鍵が提供されます。これらの鍵は、部屋の中央にある棺を開けるのに不可欠です。 どちらのパズルも対話的なデザインが似ており、プレイヤーは部屋を探り、パズルを解くのに使用できる 特定のアイテムを見つけなければなりません。

最初のパズルでは、部屋に隠されたアイテムを探します。このアイテムを見つけると、プレイヤーは引き出しの ロックを解除でき、鍵の1つが現れます。これらの対話的な要素はすべて、XR Interaction Toolkit を使用して シームレスに統合および管理され、プレイヤーにとってスムーズで没入感のある体験を実現します。



ホラーゲームの例にあるパズルの1つで、Socket Interactor を利用するもの

このパズルに使用できる機能は、XR Grab Interactor と XR Socket Interactor の組み合わせです。例えば、 パズルには 3 つの Socket Interactor があり、その中にオブジェクトを配置する必要があります。 以下で詳しく説明しますが、XR Socket Interactor には Socket Interactor が相互作用するレイヤーを 指定できる Interaction Layer Mask があります。有効なオブジェクトがソケットに配置されると、カスタム スクリプトは XR Socket Interactor から来るイベントを処理します。

```
using UnityEngine;
     using UnityEngine.XR.Interaction.Toolkit;
    5
          [SerializeField] private XRSocketInteractor socketInteractor;
public Puzzle linkedPuzzle;
              socketInteractor.selectEntered.AddListener(SocketActivated);
              socketInteractor.hoverEntered.AddListener(SocketHovered);
              socketInteractor.hoverExited.AddListener(SocketHoverExit);
          }

③ Unity Message | 0 references
private void OnDisable()

              socketInteractor.selectEntered.RemoveListener(SocketActivated);
              socketInteractor.hoverEntered.RemoveListener(SocketHovered);
              socketInteractor.hoverExited.RemoveListener(SocketHoverExit);
          private void SocketActivated(SelectEnterEventArgs arg0)
              linkedPuzzle.PuzzlePieceFound();
              arg0.interactableObject.transform.GetComponent<Collider>().enabled = false;
          2 references
          private void SocketHovered(HoverEnterEventArgs arg0) => arg0.interactableObject.transform.GetComponent<ResetGrabbedItem>().enabled = false;
          private void SocketHoverExit(HoverExitEventArgs arg0) => arg0.interactableObject.transform.GetComponent<ResetGrabbedItem>().enabled = true;
```

Socket Interactor を処理するカスタムスクリプト

XR Interaction Toolkit のコアコンポーネントと、ゲームプレイを制御するための簡単なカスタムクラスを 組み合わせることで、複雑な没入型のインタラクションを作成できます。

ホラー ゲームの例で使用されている XR Interaction Toolkit 機能のいくつかを詳しく見てみましょう。

XR Grab Interactor

Grab Interactor は、この e ブックの Interacting With The VR World セクションで紹介されています。 ユーザーが選択するこのパズルの各ゲームオブジェクトには、XR Grab Interactor コンポーネントが割り 当てられます。



XR Grab Interactable で掴むことができるオブジェクト

XR Socket Interactor

XRSocketInteractor クラスは、XR 環境でソケットベースのインタラクションを処理するために設計された 高度なツールです。さまざまな機能が提供され、No Scaling、Fixed Scaling、Dynamic Scaling などのモード があり、ソケットに接続されたときのインタラクタブルオブジェクトのスケール方法を定義できます。

このクラスは、潜在的な相互作用を示すホバーメッシュを表示することでユーザー体験を向上させ、オブ ジェクトをソケットにスナップするオプションを提供します。これには、使用可能なインタラクションと使用 できないインタラクションの両方に対するホバーメッシュマテリアルの設定が含まれ、ソケットインタラク ションをアクティブ化および非アクティブ化できます。スクリプトは衝突とターゲット処理を管理し、有効な インタラクションターゲットと衝突イベントを適切に処理します。高度なカスタマイズオプションは、ソケット 変換やインタラクタブル処理に使用できます (ソケットスナップ半径やインタラクタブル変換の調整など)。 さらに、XRSocketInteractor は Unity ライフサイクルのさまざまなイベントを処理し、XR 環境内のイン タラクタブル要素を堅牢かつ効率的に管理します。

このコンポーネントを使用して、掴んだオブジェクトをパズルに配置します。

Socket Interactor の作成

最初のパズルでは、2 つのオブジェクトを特定のエリアに配置する必要があります。空のゲームオブジェクトを 作成し、XR Socket Interactor コンポーネントを追加します。Sphere Collider が追加され、トリガーとして マークされます。この Socket Interactor は特定のオブジェクトを 1 つだけ受け取る必要があります。この 機能を実現するには、Interaction Layer Mask ドロップダウンを使用してレイヤーを指定する必要があり ます。ここで新しいレイヤーを作成して割り当てると、Socket Interactor がこの特定のレイヤーに限定され ます。ドロップダウンを使用して PuzzlePeice1 というレイヤーを作成し、割り当てます。



正しいオブジェクト強調表示メッシュを示す Socket Interactor。

次のステップは、この Socket Interactor に適合するパズルのピースを探し、XR Grab Interactable の Interaction Layer Mask を Socket Interactor に合うように変更することです。これで、この特定の掴む ことのできるオブジェクトだけがその Socket Interactor で機能するようになります。

下図は、Socket Interactor を配置したパズルです。



配置された Socket Interactor

Socket Interactor には、受け入れるレイヤーが割り当てられます。

	🔻 # 🗹 XR Socket Interactor	r (9 ∓:	:
< Persp	Script	XRSocketInteractor		0
	Interaction Manager	AR Interaction Manager (AR Interaction Manage Duzzle1 Ster	:1)	Ű
	Attach Transform	Puzziel_Stal		
	Attach Transform	None (Transform)		0
	Starting Selected Interactabl			U
	Keep Selected Target Valid			
	Show Interactable Hover Me	eshes 🗸		
	Hover Mesh Material	None (Material)		\odot
	Can't Hover Mesh Materia	al None (Material)		\odot
	Hover Scale			
	Socket Active	✓		
	Recycle Delay Time			
	▶ Interactor Filters			
	Interactor Events			
		Add Component		
		Add Component		

Layer Mask の設定

パズルに含まれる XR Grab Interactor ゲームオブジェクトも、Socket Interactor に合わせて Interaction Mask が同じレイヤーに設定されています。

	<u>y</u> =	rag ontaggeo	• Layer De	iauit
		Model Open	Select	Overrides
	×	▶ 🙏 Transform		0 ‡
	•	▶ 🌐 Crypt_Puzzle 1_Star Eleme	ent (Mesh Filter)	@ ‡
	< Persp	🕨 🖽 🗹 Mesh Renderer		@ ≓
		🕨 😪 Rigidbody		@ ≓
		🕨 🧮 🗹 Mesh Collider		@ ≓
		🔻 🛃 🗹 XR Grab Interactable		@ ‡
			XRGrabInteractable	
		Interaction Manager	XR Interaction Mana	ger (XR Interaction Manager)
			Puzzle1_Star	
		▶ Colliders		
		Distance Calculation Mode	Collider Position	
		Custom Reticle	None (Game Object)	
		Select Mode	Single	
		Movement Type	Instantaneous	
			~	
			~	
		Smooth Position		
		Track Rotation	~	
		Smooth Rotation		
		Throw On Detach	◆ 0.25	
		Throw Smoothing Curve	0.20	
		Throw Velocity Scale	1.5	
		Throw Angular Velocity Scale		
			None (Transform)	
		Attach Ease In Time	0.15	
		Attach Point Compatibility Mode	Default (Recommende	a)
ne	a:	Grab Transformers Configuration		
Q	₽ ₽ ♥ ★ 9≥5	Interactable Filters		
6		Interactable Events		
		🕨 🛃 🖌 Reset Grabbed Item (Scrip	t)	0 ‡
		Crypt_Puzzle 1 (Material)		Ø ∓
		Shader Universal Render P	ipeline/Lit	✓ Edit
			Add Component	

掴むことができるオブジェクトのインタラクションレイヤーは、Socket Interactor と同じに設定されています。

XR Socket Interactor をさらにカスタマイズして、トリガーにカーソルを合わせたときに、有効または無効な オブジェクトを表示する色とともにメッシュを表示することもできます。

このように Socket Interactor を使用すると、ゲームのパズルシステムを素早く構築でき、Socket Interactor のイベントを利用して機能をカスタマイズできます。

ホラーゲームの例では、パズルのコンポーネントであるシンプルな C# スクリプトを作成できます。この スクリプトは、パズルを完成させるために必要なステップ数を int として認識します。プレイヤーが正しい オブジェクトをソケットに配置すると、Socket Interactor によって **SelectEntered** イベントが呼び出さ れます。このイベントを使用して、ピースが見つかったことをパズルスクリプトに伝えることができます。

A REAL POINT AND A REAL PROPERTY AND A	×►	🔻 茾 🗹 XR Socket Interactor	Ø ‡
A REAL PROPERTY AND A REAL			XRSocketInteractor
	<persp< th=""><th>Interaction Manager Interaction Layer Mask</th><th>XR Interaction Manager (XR Interaction Manager) Puzzle1_Star</th></persp<>	Interaction Manager Interaction Layer Mask	XR Interaction Manager (XR Interaction Manager) Puzzle1_Star
		Attach Transform	None (Transform)
		Starting Selected Interactable	None (XR Base Interactable)
		Keep Selected Target Valid	~
		Show Interactable Hover Meshes	~
		Hover Mesh Material	None (Material)
		Can't Hover Mesh Material	None (Material)
		Hover Scale	
		Socket Active	~
		Recycle Delay Time	
		▶ Interactor Filters	
		▼ Interactor Events	
		Hover	
		Hover Entered (HoverEnterEventArgs	
		Runtime Only - Reset	:Grabbeditem.enabled
		Crypt_Puzzle1_Star (: 💿	
			+ -
		Linuar Fuite d (Linuar FuiteFuert Arres)	
		Hover Exited (HoverExitEventArgs)	
		Runtime Only - Reset	Grabbeditem.enabled
		🗈 Crypt_Puzzle1_Star (: 💿 🖌	
			+ -
		Select	
	0/1/	Select Entered (SelectEnterEventArg	s)
		= Runtime Only - Mesh	Collider.enabled
		Crypt Puzzle1 Star (O	
	: 6		
٩	ሥቆ ♥ ★ ም5	- Runtime Only - Puzz	le.PuzzlePieceFound
		■ Puzzle1Components ④	

インタラクションの促進に使用されている Socket Interactor 上のイベント

puzzlePieceCounter が増加され、必要な数に達しているかどうかがチェックされます。必要なピースが見つかるとイベントが呼び出され、パズルは終了します。次のコードスニペットはこの単純なインタラクションを制御するものです。

	⊡usi	ing UnityEngine;
	usi	ing UnityEngine.Events;
		Inity Script 1 reference
	🗆 pub	olic class Puzzle : MonoBehaviour
	}	
		//how many peices does this puzzle require
		[SerializeField] private int requiredAmountOfPuzzlePieces = 0;
		//keep track of found peices
		private int puzzlePieceCounter = 0:
11		
12		//Fvent for puzzle completion
		nublic UnityEvent OnPuzzleCompleted.
		parte on exercise on derectompreted,
		//Inack if the puzzle is completed
		mark in the puzzle is completed - false:
17		private bool isruzziecompieteu - raise,
		(/second by the endut interaction _)[s]art[strand] such
		//accessed by the socket interactors - Selectintered event
		nublic void PuzzlePieceFound()
	T	
22		
		ChackBuzzloCompletion().
		t checkPuzziecomplection(),
		(/shark to see if the user completed the number
		//check to see if the user completed the puzzle:
		neivate void (heckPuzzleCompletion()
29		
		if(isPuzzleCompleted)
		n (1) directomprecedy
33		
		if/nuzzlaPieceCounter requiredAmountOfPuzzlaPieces)
	11	(puzzierieceounter requireuxmountorruzzierieces)
		L ORBUTTLeContractor
		on bizziecompleteur. invoke();
		ISPOZZIECOMPIETEO = True;
37		Debug.Log(Puzzle'is completed");
	-!	
	-	}
	}	

XRI Toolkit の機能をいくつかのスクリプトと組み合わせて、完成したインタラクションを追跡することで、 ゲームのパズルを効率的に作成できます。XRI Toolkit、Grab と Socket Interactor、そしてそれらをつなぐ シンプルなロジックレイヤーが、プレイヤーに没入型のインタラクションをもたらします。プレイヤーは部屋の 中を移動してパズルに必要なピースを見つけて、ピースを所定の位置に置き、報酬としての鍵を入手する 必要があります。その間、天井のスパイクが下がってきます。

ゲームの残りの部分では、XRI Toolkit で利用可能なコンポーネントを使用して開発スピードを上げられます。 すべてのパズルが完成したら、次は Unity Profiler を見て、ゲームのパフォーマンスを確認します。

パズルを制御するコード

テストとイテレーション

テストは、多くの理由からゲーム開発プロセスの重要なステップです。

- ー バグの特定と修正
- ー ゲームプレイのバランス
- ー ユーザー体験と操作性
- ー パフォーマンスの最適化
- ー 互換性テスト
- ー フィードバック
- ー コンプライアンス
- ー 市場の準備

Unity は、Unity Profiler、Memory Profiler、Profiler Analyzer などのプロファイリングツールおよびテスト ツールの一式を提供しています。この Unity Discussions post では、専用の e ブック『Ultimate guide to profiling Unity games』など、これらのプロファイリングツールのための上級者向け学習リソースの概要 について説明します。 プロファイリングツールに加えて、ゲームで自動テストを作成、管理、実行するための Unity Test Framework があります。Unity best practices hub には、次の記事を含め、テスト、デバッグ、QA、プロファイリングの 上級者向け知識が多数用意されています。

- Tools for profiling and debugging
- Speed up and improve QA testing with Unity's debug class
- How to debug game code with Roslyn Analyzers
- Testing and quality assurance tips for Unity projects

プロファイリングのヒントについては、以下のビデオチュートリアルをご覧ください。



視聴する

視聴する

視聴する

Unity Asset Store は、ワークフローに統合できるサードパーティ製のテストツールも提供しています。これらの プラクティスのいくつかを取り入れ、Unity のツールを活用することで、ゲームの品質と成功の可能性を大幅に 高めることができます。

XR Device Simulator

XR Device Simulator は、XR Interaction Toolkit パッケージのサンプルアドオンの一部として含まれて いるランタイムユーティリティです。このユーティリティを使用すると、ユーザーのプレーンキー入力 (キー ボードとマウスのコンボやコントローラーからの入力など)をシミュレートして、シーン内の XR ヘッドセットと コントローラーデバイスを操作できます。

XR Device Simulator を使用すると、物理デバイスを常に装着することなく、Unity エディターで直接プロ ジェクトのテストとイテレーションを効率的に行えるため、時間を節約できます。さまざまな環境条件や ユーザーシナリオをシミュレートできるため、広範囲にわたる実世界の条件やユーザー入力を処理する アプリケーションを開発できます。これにより、潜在的なパフォーマンスの問題を早期に把握できるように なり、開発サイクルの後期に広範な最適化やハードウェアテストを行う必要性を減らすことができます。

先ほど使用したサンプルシーンにシミュレーターを導入する方法を見てみましょう。



XR Device Simulator のインストール

XR Device Simulator は、Starter Assets と同様にインポートが必要です。**Package Manager > XR** Interaction Toolkit > Samples から選択します。リストでシミュレーターを選択し、Import をクリック します。



Package Manager と インポートの準備ができた XR Device Simulator

XR Device Simulator はシーンで簡単にテストできます。デモシーンでは、次に示すように、Project ウィン ドウから XR Device Simulator プレハブを見つけてシーンにドロップします。



XR Device Simulator のプレハブを見つけてデモシーンにドロップします

エディターで再生モードにすると、シミュレーターが起動し、マウスとキーボードを使って動き回ることが できます。画面上の一連の指示に従って、どの機能がどのキーボードのキーまたはマウスボタンにリンク されているかを確認できます。



再生モードで動作する XR Device Simulator

ヘッドセットを装着している場合と同様に、シーン内を移動してゲーム内の要素を操作できるようになり ました。

Unity Profiler

Unity Profiler は、CPU 使用率、メモリ割り当て、レンダリング、ネットワーク操作など、ゲームのランタイム 動作のさまざまな側面に関する詳細情報を提供します。リアルタイムデータを読みやすい形式で提示する ことで、Profiler は開発者がパフォーマンスのボトルネックや非効率性を特定して分析できるようにします。

Profiler は、スムーズなゲームプレイを維持するためにリソース管理が不可欠な開発環境、とりわけ複雑な シーンやローエンドのハードウェアで特に威力を発揮します。ゲームのパフォーマンスメトリクスの包括的な ビューが提供され、最適化する対象について情報に基づいた意思決定を下すのに役立ちます。

Profiler を使用して XR ゲームやアプリケーションを最適化する際には、具体的にどのような側面に注力 すべきかを知ることが重要です。監視すべき主要な領域の内訳は次のとおりです。

CPU 使用率:

- メインスレッド:メインスレッドがオーバーロードされているかどうかを確認します。使用率が高く なると、フレームレートが低下する可能性があります。
- スクリプト: スクリプトの実行時間を確認します。スクリプトの最適化が不十分だと、コストが高くなる 可能性があります。
- ー 物理演算: XR では、物理演算が高負荷になることがあります。CPU への影響を監視します。



GPU 使用率:

- レンダリング: XR では、高い安定したフレームレートが求められます。レンダリング時間を監視し、
 GPU がボトルネックにならないようにします。
- ー **ドローコール:**ドローコールが多いと、GPU の動作が遅くなることがあります。パフォーマンス向上の ためにこれらを減らすことを目指します。

メモリ使用量:

- ー 合計割り当てメモリ: アプリケーションのメモリ使用量を監視します。
- ガベージコレクション: ガベージコレクションを頻繁に実行すると、パフォーマンスのスパイクが発生 する可能性があるため、割り当てを減らす方法を探します。

フレームレート:

- 安定性: XR アプリケーションは、一定のフレームレートを維持する必要があります。低下やむらを探します。
- ターゲットフレームレート: アプリケーションが目的の XR プラットフォームのターゲットフレームレート を満たしていることを確認します。
- ネットワークパフォーマンス (Multiplayer XR の場合):
- ー 待ち時間:待ち時間が長いと、マルチプレイヤー XR のユーザー体験が妨げられます。
- ー データ転送:転送されるデータ量を監視し、最適化してラグを減らします。

オーディオプロファイリング:

- オーディオソース管理: 複数のオーディオソースが負荷となる場合があります。効率的に使用できる ようにします。
- DSP ロード: 特に高度なオーディオエフェクトを使用している場合は、Digital Signal Processing の 負荷を確認してください。

レンダーパイプライン:

- バッチと SetPass の呼び出し: バッチ数とセットパスコール数をより少なくするために最適化を行い、
 GPU の負荷を軽減します。
- ー VR **固有の指標**: 可能であれば、待ち時間や瞳孔間距離調整などの VR 固有の指標を確認して ください。

UI プロファイリング:

 キャンバスレンダリング: 複雑な UI は負担になる可能性があります。キャンバス要素とその更新を 最適化します。



カスタムプロファイリングマーカー:

ー ユーザー定義マーカー: コードでカスタムマーカーを使用して、アプリケーションにとって重要な特定の メソッドやプロセスを監視できます。

サーマルスロットリング (特にモバイル XR の場合):

 デバイスの温度: ハイパフォーマンスが求められると、モバイルデバイスが過熱し、サーマルスロット リングが発生する可能性があります。

この領域を慎重に分析することで、パフォーマンスのボトルネックを特定し、それに応じて XR ゲームやアプ リケーションを最適化できます。目標は、現実感のあるビジュアル、応答性、スムーズなパフォーマンスの バランスを維持することです。

Profiler ウィンドウを開くには、Window > Analysis > Profiler を選択するか、Ctrl+7 を押します。 Play を押すと Profiler が起動し、プロジェクトからキャプチャしたデータがビューに表示されます。



Profiler でのプロジェクトのプロファイリング

いつでも一時停止して、フレームレートの低下を引き起こしているゲーム内の領域を特定できます。左側の メニューを使用して Rendering や CPU Usage などの特定のレイヤーを選択すると、詳細を取得できます。 各項目は色分けされており、ゲームのパフォーマンスの問題が発生している場所を素早く特定するのに 役立ちます。 前述のプロファイリングの e ブックに加えて、2 つの上級者向け最適化ガイドには最適化のヒントが多数 用意されています。



ビルドとデプロイ

ワールドの構築、インタラクションの構築、テスト、最適化に時間をかけてきました。さあ、制作したゲームを 世界に公開しましょう。すぐにゲームをデバイスにビルドできるようになりました。または、ゲームをストアに 提出することもできます。各ストアには、アプリケーションに関する独自のガイドラインと要件があります。 アプリケーションの提出に関するドキュメントを確認してください。



VR アプリケーションの構築





複合現実 (MR) アプリケーションは、現実世界と仮想世界を統合してハイブリッドな状況にします。これらの 要素の融合は、ユーザーにとってシームレスに感じられる必要があります。

MR 開発は、次のような VR と AR の世界構築の側面を組み合わせたものです。

ユーザーインタラクションとインターフェースデザイン

VR や AR からの UX と UI の設計に関する共通の原則は、MR アプリケーションに適用できます。さまざまな デバイスで自然に感じられる直感的なインターフェースとインタラクションモデルを設計する必要があります。 これにはハンドトラッキング、視線制御、音声コマンドの活用などがあり、ユーザーが直感的に仮想世界を 操作できるようにします。



空間認識と物理的性質

VR、AR、XR では、仮想世界の物理的性質を理解して組み込むことが不可欠です。これには、バーチャル オブジェクト間のリアルなインタラクションの作成や、AR の場合はそれらのインタラクションを現実世界と 融合することが含まれます。開発者は空間認識を考慮し、重力、衝突、その他の物理的特性に関するユーザーの 予想と一致するように仮想オブジェクトが動作する必要があります。

クロスプラットフォーム開発戦略

スタンドアロンのヘッドセットからモバイルベースのシステムまで、VR/AR に使用できるデバイスは多岐に わたるため、クロスプラットフォームの互換性を念頭に置いて開発することが重要です。Unity は、開発者が 最小限の調整で複数のプラットフォームに展開できるアプリケーションを作成できるツールとワークフローを 提供します。これには、さまざまなデバイスの機能と制限に合わせてユーザーインターフェースやインタラク ションモデル、パフォーマンスの最適化を適合させることが含まれます。

環境デザインと没入感

MR の環境設計は、VR と AR の両方からの没入型の手法を組み合わせたものです。これには、視覚的な 側面だけでなく、空間オーディオ、触覚フィードバック、その他の感覚入力を組み込むことで、説得力のある 魅力的な世界を作り出します。目標は、VR の完全な仮想世界であろうと、AR の現実世界のデジタルオー バーレイであろうと、仮想環境の中に自分が存在しているようにユーザーに感じさせることです。





MR テンプレート

Unity の MR テンプレートは、Unity で複合現実を開発するときの開始点として機能するように設計されています。

•••	New p Editor Version: 20	oroject D22.3.21f1 urs 🗘
≔ All templates	SRP Core	•
 Core Sample 	SRP 3D (HDRP) Core	•
Learning	Mixed Reality Core	
	WR Core	Mixed Reality Unity's Mixed Reality (MR) Template provides a starting point for mixed reality development in Unity. The template configures project setting
	Core	Read more Download template
	Small Scale Competitive Multiplay Core	yer a
		Cancel Create project

Unity Hub の Mixed Reality テンプレート

テンプレートをダウンロードして、新しいプロジェクトを作成します。Unity 2022 LTS 以降を使用することを 推奨します。

プロジェクトが開始画面と共に開きます。チュートリアル、専用フォーラム、バグ報告などへのリンクを提供 するウィンドウもあります。

					ray ontagged
	Tutorials	/ 			Prefab RMRInteractions
	Mixed Realit	y Template ses to get you started	T	<persp< th=""><th>Position Rotation Scale</th></persp<>	Position Rotation Scale
	Ø	Quick Start Guide MR Learn about the template features			
	Ø	Forums Connect with the XR community			
	Ø	Bug Reporting Report bugs to the XR team			
		Template Feedback Tell us about your experience		\times	
XR		ų		A: ₽ ₽ ₽ 9 ★ ¢30	
ders				A	

チュートリアルコンテンツのウィンドウ



Quick Start リストには、MR テンプレートの ドキュメントへのリンクもあります。

MR テンプレートは、OpenXR、AR Foundation、XR Interaction Toolkit のツールセットを使用します。

MR Interaction Setup プレハブ

MR Interaction Setup プレハブは、XR カメラと XR 体験の原点を設定するための主要なゲームオブジェ クトです。Controller や Hand などの異なる入力システムに対応し、Poke (つつく)、Direct (直接)、レイ などのさまざまなインタラクターを備えています。



MR テンプレートの Hierarchy 内の MR Interaction Setup プレハブ

MR Interaction Setup プレハブには、次のさまざまなコンポーネントが用意されています。このガイドの 前半の章を読んでいれば知っていることでしょう。Input Action Manager、XR Interaction Manager、XR Origin (XR Rig)。MR テンプレートに固有のコンポーネントをいくつか見てみましょう。

AR Session

AR Session コンポーネントは、AR セッションのライフサイクルと設定を管理します。アクティブなセッションは 常に1つだけです。



AR Plane Manager

AR Plane Manager は、プレイヤーの周囲で検出された各平面のゲームオブジェクトの生成を担当します。 デバイスのカメラと AR Foundation を使用して、検出された平面を管理します。このコンポーネントは XR Origin コンポーネントと同じゲームオブジェクト上にあります。



XR Origin 上の Plane Manager コンポーネント。

Plane (プレーン) とは、ポーズ、寸法、および境界点で表される平面のことです。平面として検出できる要素の 例として、平らなテーブル、床、壁、カウンターがあります。Detection Mode (検出モード) はは、水平、垂直、 またはその両方に設定できます。さらに、Manager には、カスタムプレハブを使用して検出された平面を 視覚化するオプションもあります。



垂直面上の AR 平面



床と壁から検出された平面

M

パススルー

パススルーテクノロジーにより、VR ヘッドセットで現実世界と仮想オブジェクトの両方を同時に表示し、 デバイスのカメラを使用して周囲の環境をキャプチャできます。MR テンプレートでは、パススルーはソフト ウェアとハードウェアを組み合わせて機能し、デジタルコンテンツを現実世界と瞬時に融合させます。 この機能は主に AR 体験で使用され、バーチャルシーンにいる間に現実世界の環境を見て操作できます。 Unity では、パススルーは次のいくつかの重要な部分に依存します。

- XR Plugin Framework: このシステムは、カメラフィードやその他のハードウェアリソースへのアクセス に必要なインターフェースとコントロールを提供するため、パススルーを実現するうえで不可欠です。
- ハードウェアサポート: ターゲットデバイスには、ユーザー環境からライブビデオをキャプチャできる カメラなど、パススルーをサポートするために必要なハードウェアが要ります。このハードウェアの 品質と機能は、パススルー体験の効果と現実感に大きく影響します。
- カメラフィードへのアクセス: パススルー機能の中核となるのは、デバイスのカメラからライブフィードに アクセスできることです。その後、このフィードがユーザーの動的視野内でリアルタイムにレンダリング され、仮想環境内の現実世界へのウィンドウが作成されます。
- **ソフトウェアコンポーネント:** ARKit および ARCore デバイス用の AR Foundation パッケージや Oculus デバイス用の Oculus Integration パッケージなど、Unity プロジェクト内の特定のコンポー ネントは、パススルーの実装に必要なツールと API を提供します。これらのコンポーネントは、カメラ フィードのレンダリング、ユーザー入力の処理、デジタルコンテンツと物理環境の統合に伴う複雑さを 管理します。



MR テンプレートのテスト

MR テンプレート内のプロジェクト設定は事前設定されており、XR Plug-in Management パッケージは Meta Quest 機能グループで OpenXR を使用するように設定されています。Meta Quest Touch Pro Controller Profile と Oculus Touch Controller Profile の 2 つのインタラクションプロファイルが追加されました。

Meta Quest でテンプレートをテストします。Meta Quest は Android ベースのプラットフォームで、スタンド アロン VR/MR アプリケーションの Android ビルドターゲットと、Meta Quest Link を介した PCVR アプリ ケーションの Windows ビルドターゲットをサポートします。このテンプレートでは、複合現実機能は現在 スタンドアロン/Android サポートに限定されているため、ビルドターゲットは Android に設定されています。 エラーメッセージが表示された場合は、プロジェクトで Android が選択されたプラットフォームであることを 確認してください。デフォルトの Windows、Mac、Linux プラットフォームではエラーが発生する可能性が あります。

平面検出を機能させるには、ユーザーがプレイする空間に合わせてヘッドセットを設定する必要があります。 Meta Quest では、Settings で物理的空間を設定する必要があります。これにより、壁、床、天井、その他の アイテムの位置がヘッドセットに通知されます。

プロジェクトを構築して実行すると、UI が表示されます。この UI では、サンプルプロジェクトの機能をひと とおり説明します。UI を手で操作し、平らな表面をタップして 3D オブジェクトを配置、移動、除去することが でき、その他の機能も使用できます。

Unity の複合現実プロジェクトテンプレートでは、パススルーテクノロジーの難しい設定作業の多くが簡略化 されています。つまり、開発者は MR プロジェクトを作成するための設定、スクリプト、およびリソースが調整 済みのパッケージを受け取ります。デジタルと現実世界を融合したアプリケーションをもっと簡単に始めら れるように設計されています。それでも、プロジェクトのニーズやターゲットとするデバイスによっては、すべてを 適切に実行するために設定を調整したり、パーツを追加したり、独自のコードを記述したりする必要が あります。

Apple Vision Pro による 空間コンピューティング



visionOS でウィンドウアプリケーションとして実行される Unity サンプル

Apple Vision Pro は Apple 初の空間コンピューターです。現実世界とデジタル世界を統合するので、 ユーザーはその両方を同時にアプリケーションで操作できます。そのデバイスは、高度なセンサー、ディス プレイテクノロジー、ソフトウェアアルゴリズムを活用して、非常にリアルな VR/AR 体験を創造します。

Apple Vision Pro 上の空間コンピューティングは、ユーザーの環境や動きを認識して反応することで、 ゲーミングやエンターテインメントから教育、専門的なコラボレーションまで、幅広く応用できます。



Triband による[『]WHAT THE GOLF?』。Apple Vision Pro での没入型の体験

Unity で visionOS を始める

このセクションでは、Unity で作成できるさまざまな種類の Apple Vision Pro 体験について説明します。

- 一 没入型 (複合現実): Unity が新たに開発した PolySpatial テクノロジーにより実現可能な、デジタル コンテンツと現実世界を融合し、共有空間で他のアプリケーションと一緒に動作させることができる 体験
- 一 完全没入型 (仮想現実): 既存のバーチャルリアリティゲームの移植や、プレイヤーの周囲を別の環境に 置き換える新しい完全没入型体験
- ー ウィンドウ: 2D ウィンドウで動作し、共有空間内でユーザーがサイズや位置を変更できるコンテンツ



visionOS アプリケーションを作成するには、以下のハードウェアが必要です。

- visionOS の開発には Apple シリコンを搭載した Mac および visionOS シミュレーターを搭載した Xcode 15.2 以降が必要
- visionOS シミュレーターには Apple シリコン (M1 以降) Mac が必要
- Xcode Settings/Platforms に visionOS シミュレーターをインストールすることが必要
- ー visionOS ハードウェアがない場合は、XCode に組み込まれたシミュレーターを使用

						Platforms					
	දි දුදි General	(C) Accounts	Behaviors	Ravigation	Themes	Text Editing	Key Bindings	Source Control	Platforms	Locations	
	macOS	14.4								Built-in	
	iOS 17.4									7,23 GB on disk	
0	watchO	S 10.4							3,93	3 GB Get	
	tvOS 17.	4							3,68	3 GB Get	
\frown	visionOS	5 1.1								6,8 GB on disk	

Xcode で利用可能なシミュレーター

visionOS アプリケーションの開発を始めるには、Unity Hub から最新バージョンの Unity 2022 LTS を ダウンロードしてインストールします。 モジュールリストから、visionOS Build Support と iOS Build Support の各モジュールを選択して インストールします。

Add modules for Unity 2022.3.23f1 📹 Silicon LTS		×
Add modules	Required: 0 by	tes Available: 1.48 TB
Visual Studio for Mac	Installed	3.57 GB
▼ PLATFORMS	DOWNLOAD SIZE	SIZE ON DISK
Android Build Support	Installed	2.02 GB
└── OpenJDK	Installed	219.56 MB
└── Android SDK & NDK Tools	Installed	4.59 GB
iOS Build Support	Installed	1.55 GB
visionOS Build Support	Installed	816.11 MB
		Install

Vision Pro でのアプリケーションの表示方法とアクセス方法の図。画像提供: Apple visionOS と Unity

visionOSとUnity について



Vision Pro でのアプリの表示方法とアクセス方法の図。画像提供: Apple



visionOS のアプリケーションは、iPhone や iPad のアイコングリッドと同様に、Home View のアイコンから 起動します。上記の画像は、Vision Pro でのアプリケーションの表示方法を示しています。

- 1. 起動してウィンドウやボリュームでコンテンツを実行するアプリケーションは共有空間 (1) 上に 横並びで表示され、Mac デスクトップ上の複数のアプリケーションと同様に、移動や再配置が できます。
- 2. アプリケーションがボリューム **(2)** を使用する場合、Volume Camera (ボリュームカメラ)内 のコンテンツのみが表示されます。カメラのモードは Bounded (境界あり) に設定されており、 共有空間内のアプリケーションのコンテンツが表示されます。
- 3. また、カメラのモードが Unbounded (境界なし) に設定されている場合は、没入空間 (3) で 利用可能なスペース全体をアプリケーションが占有することもできます。これにより、共有 空間のアプリケーションが非表示になり、開いているアプリケーションにのみ焦点をあてます。

Bounded モードでも Unbounded モードでも、カメラパススルーと物理的空間が目の前の現実世界と バーチャル 3D オブジェクトを融合します。



Bounded モードが適用されたボリュームカメラでアプリケーションを実行する共有空間

アプリケーションのコンテンツにカメラフィードと物理的空間を使用しない 2 つの追加的なモードとして、 プレイヤーやユーザーを完全に仮想世界に入れる完全 没入型 VR と、目の前の平らなフローティング ウィンドウにレンダリングされる 2D または 3D 体験に使用できて共有空間で共存可能な**ウィンドウ アプリ** ケーションがあります。iPhone と iPad のアプリケーションは、visionOS 上でウィンドウアプリケーションと して動作します。



デバイスで利用できるすべてのアプリケーションを起動可能な visionOS の Home View。

アプリケーションの種類	共有空間	AR	レンダリング テクノロジー	Play To Device	Vision Pro プラグインが必要
完全没入型 VR	いいえ	いいえ	Metal	いいえ	はい
Bounded モードの MR	はい	はい	RealityKit	はい	はい
Unbounded モードの MR	いいえ	はい	RealityKit	はい	はい
ウィンドウアプリケーション	はい	いいえ	Metal	いいえ	いいえ

Apple Vision Pro のさまざまな XR 体験の主な機能と要件の概要は次のとおりです。

Vision Pro プラグインは、必要に応じて visionOS パッケージを自動的にインストールします。

Apple Vision Pro 向けの Unity プロジェクトにおけるインタラクティビティの仕組みを見てみましょう。

インタラクション

visionOS アプリケーションでは、ユーザーは目と手を使ってコンテンツを見て操作します。指先同士をタップ することでアイテムを選択します。ユーザーの許可があれば、完全なハンドトラッキングとヘッドポーズの データもアプリケーションで利用できます。カスタムジェスチャはインタラクションをリアルに感じさせる もので、一般的な対話的操作はタップジェスチャです。



既存のアプリケーションのインタラクションを visionOS に変換。出典:Apple Inc.

入力コライダーをオブジェクトに付けて、これらのイベントをオブジェクトが受信できるようにする必要が あります。離れたところからオブジェクトを見てタップして選択するか、手を伸ばして指で直接オブジェクトに 触れることができます。

Unity では、タップを WorldTouch イベントとして利用できます。2D タップイベントに似ていますが、位置は 完全に 3D です。

iPhone などですでにタッチで作業している場合は、適切な入力コライダーを追加して、引き続きタップを 主な入力メカニズムとして使用できます。

VR コントローラを使用している場合は、その複雑さに応じて、タップベース入力またはハンドベース入力の 観点からインタラクションを再定義する必要があります。

3D TouchとTouchSpace

境界ありボリュームと境界なしボリュームの両方で、ユーザーが入力コライダー付きのオブジェクトを見て "ピンチ"ジェスチャを実行すると、3D Touch 入力が行われます。SpatialPointerDevice入力デバイスは、 その情報を開発者に提供します。ユーザーがピンチジェスチャのままだと、ドラッグが開始され、アプリケー ションには元の開始位置を基準にした"移動"の更新内容が表示されます。



この入力デバイスには、VisionOSSpatialPointerDevice という VR 対応デバイスがあります。 この 2 つの主な違いは、インタラクションにコライダーを必要としないことです。したがって、 VisionOSSpatialPointerDevice にはインタラクションに関連する入力コントロール (targetId、 interactionPosition など) がありません。

スケルトンハンドトラッキング

スケルトン ハンド トラッキング は、XR Hands パッケージの Hand Subsystem によって提供されます。 シーン内で Hand Visualizer コンポーネントを使用すると、スキンメッシュやジョイントごとのジオメトリを プレイヤーの手に表示したり、ハンドベースの物理的インタラクション用の物理的オブジェクトを表示したり できます。Package Manager で入手できる XR Hands パッケージを使用して Raw データにアクセスする こともできます。



XR Hands パッケージは、関節を追跡し、独自のジェスチャを定義したり、カスタムメッシュの手のビジュアルを 置き換えることができます。



- 0 .handWrist
- 1 .handThumbKnuckle
- 2 .handThumbIntermediateBase
- 3 .handThumbIntermediateTip
- 4 .handThumbTip
- 5 .handIndexFingerMetacarpal
- 6 .handIndexFingerKnuckle
- $\label{eq:rescaled} \textbf{7} \textbf{.handIndexFingerIntermediateBase}$
- 8 .handIndexFingerIntermediateTip
- 9 .handIndexFingerTip
- 10 .handMiddleFingerMetacarpal
- 11 .handMiddleFingerKnuckle
- 12 .handMiddleFingerIntermediateBase
- 13 .handMiddleFingerIntermediateTip

- 14 .handMiddleFingerTip
- 15 .handRingFingerMetacarpal
- 16 .handRingFingerKnuckle
- 17 .handRingFingerIntermediateBase
- 18 .handRingFingerIntermediateTip
- 19 .handRingFingerTip
- 20 .handLittleFingerMetacarpal
- 21 .handLittleFingerKnuckle
- 22 .handLittleFingerIntermediateBase
- 23 .handLittleFingerIntermediateTip
- 24 .handLittleFingerTip
- 25 .handForearmWrist
- 26 .handForearmArm

手のデータと表示のマッピング



次の表に、使用可能な入力メソッドの概要を示します。ハンドトラッキングと AR データはユーザーによる 許可が必要であることに注意してください。

アプリケーションの種類	ルックアンドタップ (Input System タッチスペース イベント)	ヘッドポーズ (Input System ヘッドポーズイベント)	ハンド (XR Interaction Toolkit または XR Hands パッケージ)	AR データ (AR Foundation パッケージ)	Bluetooth デバイス (Input System)
完全没入型 VR	はい	はい	はい	いいえ	はい
Bounded モードの MR	はい	いいえ	はい	いいえ	はい
Unbounded モード の MR	はい	はい	はい	はい	はい
ウィンドウ アプリケーション	はい	いいえ	いいえ	いいえ	はい

完全没入型 VR



Owlchemy Labs による『Job Simulator』は、visionOS 用の VR アプリケーションです。同社のゲームの詳細については、こちらの Unity プログ記事を参照して ください。

VR アプリケーションの作成や visionOS への移植は、この e ブックで前述したように、VR 開発などの使い 慣れた Unity ワークフローを使用して行うことができます。



visionOS のビルドプロセスは、iOS でのデプロイと似ています。Unity は Xcode プロジェクトを生成し、 Xcode からシミュレーターまたはデバイスでプロジェクトを実行して公開できます。

Apple Vision Pro 開発のコンテキストでは、完全没入型 VR アプリケーションは Unity のグラフィックステク ノロジーと Metal フレームワークを使用してプロジェクトをレンダリングします。

プロジェクトがすでに他の VR プラットフォームで実行されている場合は、visionOS でも実行できます。

URP がサポートするのは 中心窩 レンダリングであるため、URP の使用が推奨されます。この特徴により、 視線が最も集中しやすい視覚領域にピクセル解像度がより集中し、この領域以外の解像度が低下するため、 パフォーマンスとユーザー体験が向上します。中心窩レンダリングはデフォルトで有効になっています。これを 使用しないと、Apple Vision Pro をデバイスの完全なネイティブ解像度ではレンダリングできません。既存の アプリケーションを移植する開発者は、フォールバックとしてそれを無効にする必要があるかもしれませんが、 そのようなことは推奨されません。

Metal グラフィックス API をサポートするようになったシングルパス インスタンスレンダリングを使用して、 プロジェクトを最適化できます。シングルパスレンダリングでは、Unity は両目に対して 1 つのドローコールを 送信するので、それぞれの目に対して別々にレンダリングする場合よりも CPU オーバーヘッドを削減します。

visionOS では、深度バッファは再投影に使用されます。深度情報がない場合、システムはエラー色で問題を示します。すべての Unity シェーダーは、そのまますぐに使用できます。

アプリケーションを完全没入型モードで実行するには、Apple visionOS XR プラグインをインストール する必要があります。使用しているバージョンの Unity に表示されない場合は、Package Manager の ドロップダウンから**com.unity.xr.visionos パッケージ**を選択することで、XR Plug-in Management から追加できます。



また、XR Plug-in Management パッケージの Project Validation セクションで、visionOS で実行する ために変更が必要な潜在的な設定を確認することも推奨します。



visionOS Plugin をインストールすると、構築するアプリケーションのタイプを定義できます。VR、MR、またはウィンドウ型

XR Plug-in Management で visionOS Plugin を有効にしたら、左側メニューで **Apple visionOS** を クリックし、アプリケーションのタイプをそこから選択します。Build Settings ウィンドウにそのタイプが反映 されます。

Xcode シミュレーターでプロジェクトを実行する場合は、Build Settings から **Target SDK** > **Simulator SDK** を選択するか、**Device SDK** を選択して Vision Pro デバイスで実行してください。



visionOS シミュレーターで実行される VR アプリケーション

Vision Pro 向け開発者 VR アプリケーションの詳細については、ドキュメントを参照してください。 Unity は、Unity グラフィックスシステムやツールの使用方法に関する詳細な e ブックも提供しています。

MR アプリケーション



Light Brick Studio の 6 人のチームが『LEGO® Builder's Journey』をわずか 3 か月で Apple Vision Pro に移植した方法を学びましょう。

Unity の PolySpatial テクノロジーを使用して、Apple Vision Pro の共有空間で実行される MR アプリケー ションを作成できます。これらのアプリケーションは Unity でシミュレートされますが、visionOS のシステム レンダラーである RealityKit でレンダリングされます。Unity VR 開発パラダイムのほとんどはここに適用 され、AR コンポーネントも追加されます。

MR アプリケーションの開発を始めるには、Apple visionOS の Project Settings ウィンドウに移動し、 App Mode を **Mixed Reality - Volume or Immersive Space** に変更します。ポップアッププロンプトが 表示され、PolySpatial のインストールを求められます。もう 1 つのオプションは、com.unity.polyspatial. visionos パッケージを名前によって手動で追加することです。



MR アプリケーションの作成には visionOS Plugin と Polyspatial の両方のパッケージが必要です



Unity PolySpatial

Unity PolySpatial は、Unity の visionOS パッケージの一部であり、共有空間内で他のアプリケーション と共存できる visionOS プラットフォーム用の空間体験を作成するための基盤となるテクノロジーです。



Unity の PolySpatial テクノロジーによって Apple の RealityKit に変換される Unity のシミュレーションとグラフィックスの概要

グラフィックスとシミュレーション

PolySpatial は、RealityKit でグラフィックスをレンダリングするために必要な変換を行い、Unity の機能や ワークフローの多くが MR で動作するようにします。



Unity マテリアルから RealityKit マテリアルへの変換フローです。


プロセスをより詳しく見てみましょう。

- マテリアル: Shader Graph で作成された URP Lit、Simple Lit、および Complex Lit シェーダー とカスタムマテリアルは、RealityKit で機能します。MaterialX は、複雑なマテリアルやカスタム シェーダーの変換に使用される交換方式です。Shader Graph ノードのほとんどがサポートされて いて、サポートの詳細についてはドキュメントを参照してください。エフェクトマテリアル、Unlit、 オクルージョンマテリアルもサポートされます。
- メッシュレンダラー:メッシュレンダラーとスキンメッシュレンダラーは、URP またはビルトインレンダー パイプラインでサポートされています。
- パーティクルエフェクト: ビルトインパーティクルシステムで作成したシンプルなエフェクトは RealityKit に変換されます。複雑なエフェクトはメッシュにベイクされ、スプライトエフェクトもメッ シュに変換されます。VFX グラフは MR ではサポートされていませんが、エフェクトをレンダーテクス チャにレンダリングしてプロジェクトで使用できます。
- ー **スプライト**: サポートされています。
- シミュレーション: Monobehaviour、ScriptableObject、その他の標準ツールなど、ワールドシミュレーション機能は引き続き使用できます。
 - 一 物理演算
 - ー アニメーションと Timeline
 - 経路検索
 - MonoBehaviour
 - ー その他の非レンダリング機能

ボリュームカメラ

Volume Camera (ボリュームカメラ) は PolySpatial での新しい概念です。アプリケーションはいつでも 2 つのタイプを切り替えることができます。Volume Camera コンポーネントは、レイヤーマスクをカリングし、 寸法を定義できます。サイズが大きいほど、表示されるシーンビューが大きくなります。また、アプリケーション ウィンドウの変更に基づいて機能をトリガーする Unity イベントがあります。カメラモードは、**Volume Camera Configuration** アセットを使用して設定します。



Unity での Bounded カメラの設定



Bounded カメラ

このモードでは、Unity はボリュームカメラ内部のコンテンツを表示するために実世界の単位に基づいて デバイスで使用される空間領域を定義できます。



ドキュメントでバウンドカメラのサンプルを検索

Unbounded カメラ

このモードでは、MR 体験を空間全体に遷移させることができます。ここで、アプリケーションはユーザーの 視野全体を自由に利用できます。作成した境界なしのコンテンツが空間で見える唯一のコンテンツです。 ボリュームカメラの寸法や出力の寸法は設定できません。Unity ユニットは実世界の単位 (1 Unity ユニット = 1 メートル) にマッピングされます。Unbounded カメラは一度に1つだけです。



空間を最大限に活用するアンバウンド MR アプリケーション。他のテスト用のサンプルについては、ドキュメントを参照してください。

Play To Device

この機能により、開発者は Xcode ビルドを作成することなく、visionOS デバイスとシミュレーターにて コンテンツを再生モードでプレビューできます。RealityKit を使用してレンダリングされた複合現実アプリ ケーションで利用できます。

Play to Device を使用すると、要素の配置の確認、マテリアルテクスチャやシェーダーグラフの変更、インタ ラクションのテスト、値の調整などを効率的に行うことができます。デバイスまたはシミュレーターで実行する インタラクションは、すべてエディターに同期されます。このワークフローの詳細については、こちらのディス カッションの投稿またはドキュメントを参照してください。



Unity の再生モードと並行して実行されている Play To Device

この機能は Play to Device Host アプリケーションを通じて提供されます。このアプリケーションは、 visionOS シミュレーターにインストールするか、Apple Vision Pro に直接インストールできます (Testflight 経由)。



Play to Device をシミュレーターにドラッグアンドドロップしてインストールし開きます。Unityの機能によってホストアプリケーションが検出され、再生モードに入ったときにアプリケーションに接続されます (これは Connect on Play で無効にできます)。

ウィンドウアプリケーション

このモードでは、コンテンツが 2D ウィンドウ内に表示され、ユーザーは共有空間内でコンテンツのサイズ 変更および再配置ができます。完全没入型の VR アプリケーションと同様に、ウィンドウアプリケーションは Metal ベースの Unity レンダリングを利用するため、従来の 2D ディスプレイ向けに制作されたゲームを visionOS 上にてウィンドウモードで実行できます。



visionOS で動作するウィンドウ型 Unity アプリケーション



visionOS プロジェクトテンプレート

新しいプロジェクトを開始するには、visionOS プロジェクトテンプレートをダウンロードすることを推奨 します。テンプレートは、テンプレートへのリンクが表示された「Starting a new visionOS project from the Immersive App Template」 からダウンロードできます。

visionOS テンプレートには、境界ありの MR 体験と境界なしの MR 体験、応答性に優れたレイアウト、 XR 入力とインタラクションの例が含まれています。このサンプルでは、境界なしのボリュームシーンでの タップ、ハンドトラッキング、平面検出を使用しています。

このテンプレートでは、プロジェクト設定を構成し、適切な XR パッケージをプリインストールします。また、 visionOS にデプロイする準備ができているプロジェクトを設定する方法を示す、さまざまな事前設定済み アセットも含まれています。

Unity Hub からテンプレートを開くには、Add project from disk オプションを選択します。開いたら、 エディター内のチュートリアルを使用できます。表示されていない場合は、Tutorials > Show Tutorials を 選択してください。

🖆 Unity File Edit Assets GameObject Compone	nt Services Jobs Tutorials Window Help	🛠 💿 🖅 🗢 🤇 😂 /	Sun 11 Feb 09:25
Untitled - visionOSTemplate-1.0.3 - Windows, Mac, Linux - Unity 2022.3.19f1 <metal></metal>			
⊖ DS - ● 🔞	• • • Tutorials	Experimental Packages In Use < Q L	ayers 🔻 Layout 👻
≔ Hierarchy A : # Scene	😎 Game	Inspector Project Settings	
+ ▼ • All 🖉 Game	Display	Package 'PolySpatial visionOS' Manifest	
♥ Untitled E Nain Camera		Edit	View in Package Manager
Directional Light	visionOS Template	Information	
		Name com unity polyspatial visionos	
		Display name PolySpatial visionOS	
	Editor UI	Version 1.0.3	
	Learn the Unity interface for PolySpatial	Category	
		Description	
	Shared World Apps Learn PolySpatial core technology	Unity PolySpatial integration package to support shared and im on Apple visionOS. Unity PolySpatial and visionOS support is only available to Unit and Unity Industry users. Learn more about these plans at unity	nmersive mode rendering ty Pro, Unity Enterprise, y.com/pricing.
		Dependencies	
		Package name Ve	ersion
	Documentation	PolySpatial 1.0	0.3
	Detailed documentation from the Unity	PolySpatial XR 1.0	
Project 🗟 Console 💿 Unity Version Control			
+•			
Coolections Core RP Library Core RP Library Core NU/nit Editor Coroutines Finathuffer support Finathuffer support Mono Cecil Nono Cecil Nono Cecil Nono Cecil PolySpatial			
g endered million copy of settings. Alterinduction receivences at Assets/Alt	y one of the stand of the stand gay the source of the transmission of the stand of		

さまざまな PolySpatial リソースにアクセスできる Tutorials ウィンドウ

チュートリアルでは、テンプレートのさまざまな機能をガイド付きで説明します。



プロジェクトとそのコンポーネントのガイド付きツアー

プロジェクトには SampleScene と SampleSceneUnbounded という 2 つのサンプルシーンがあり、 それぞれ Bounded モードと Unbounded モードの設定と機能を示しています。



バウンド MR アプリケーションのアプローチを示す SampleScene。

これらのアプリケーションは、以前説明した visionOS シミュレーターと Play To Device アプリケーションを 使用して、エディター内で体験することもできます。



アンバウンド MR アプリケーションの設定を示す SampleSceneUnbounded

その他のリソース

Unity コミュニティとつながり、Unity の visionOS 開発のためのディスカッション スペースで XR 開発に 関するインサイトやヒントを得ることができます。今後の機能については、ロードマップを参照してください。

プロフェッショナルトレーニングサービス

visionOS でプロフェッショナルな体験を作成することを目指している方々のために、Unityは 1:1 のライブ およびハンズオントレーニングを提供しています。この パーソナライズされた トレーニングでは、次の内容を 扱う Unity エキスパートをあなたのチームに提供します。

- ー visionOS での目標と独自のプロジェクト。
- visionOS 向けの開発の主なコンセプトと、プロジェクトのための Apple Vision Pro とその機能の 理解。
- ー 開発環境、構成設定、および初めてのアプリケーションの構築。

Unity での AR の活用



人気の AR 開発者 Niantic Labs による最新のプロジェクト Peridot のスクリーンショット

拡張現実 (AR) アプリケーションの開発は、ARKit (iOS 用) や ARCore (Android 用) のような AR 開発 プラットフォームで構成される AR Foundation などの Unity の強力なフレームワークを活用する刺激的な プロセスです。各種プラットフォームに対する幅広いサポートと広範なコミュニティリソース。



まずは、AR モバイルコアテンプレートをお勧めします。詳細はこちらで確認できます。



A

AR Foundation: 開発者が単一の API を使用して複数のプラット フォームにわたって拡張現実 (AR) 体験を構築できる Unity フレーム ワーク。

ASTC (適応型スケーラブルテクスチャ圧縮):品質とパフォーマンスのバランスを管理するのに役立つテクスチャ圧縮手法。モバイルや VR アプリケーションで特に便利です。

С

CPU 使用率: アプリケーションが使用しているプロセッサーリ ソースの量を測定します。CPU 使用率が高くなると、アプリケー ションのパフォーマンスと応答性に影響を及ぼす可能性があります。

D

DSP ロード (デジタル信号処理ロード): オーディオ信号の処理に 使用される処理能力。DSP ロードが高くなると、パフォーマンスに 影響を及ぼす可能性があります。

G

GPU 使用率: グラフィックスプロセッサーのリソースの使用量を 測定します。これは、レンダリングパフォーマンスとビジュアル品質に とって重要な要素です。

Η

HDRP (HD レンダーパイプライン): ハイエンドのハードウェアでの 忠実度の高いグラフィックスのために設計された Unity レンダー パイプライン。

Input system: キーボード、マウス、ゲームコントローラーなど、 さまざまなデバイスからの入力を統一された方法で管理するための Unity のシステム。

Interaction Layer Mask: Unity の XR Interaction Toolkit で 使用され、レイヤーに基づいてインタラクターが操作できるオブ ジェクトを指定します。

L

LOD (詳細レベル): カメラからの距離に基づいて 3D モデルの 複雑さを調整することでパフォーマンスを改善する手法。

0

OpenXR: 複数のプラットフォームでの VR/AR アプリケーションの 開発の標準化を目指す、ロイヤリティフリーの標準。



Ρ

PolySpatial: 空間コンピューティングに関連するテクノロジー または概念。物理的空間内でのデジタルオブジェクトと環境の シームレスかつ直感的な方法でのインテグレーションとインタラ クションを強調します。Unity のコンテキストでは、PolySpatial とは、リアルタイム 3D コンテンツの作成とインタラクションの ためのプラットフォームの機能を活用して、そのような空間的に 統合された体験の作成を促進するように設計されたツール、フレーム ワーク、またはプラクティスのことを指します。

Q

Quality 設定:開発者がアプリケーションのグラフィック品質を 調整してパフォーマンスや現実感のあるビジュアルを最適化できる Unity 設定。

S

SDK (ソフトウェア開発キット): 開発者が特定のプラットフォームや デバイス用のアプリケーションを作成するのに役立つよう設計された 一連のソフトウェアツールとライブラリ。

SRP (スクリプタブルレンダーパイプライン): 開発者がプロジェクトの 特定のニーズに合わせてレンダリングプロセスをカスタマイズできる Unity の機能。

U

URP (ユニバーサルレンダーパイプライン):モバイルからハイエンドの PC まで、幅広いデバイスのパフォーマンスに合わせて最適化 された Unity レンダーパイプライン。

V

VR (仮想現実): コンピューターで生成された環境をユーザーが 操作できるようにするテクノロジー。多くの場合、没入型の体験には ヘッドセットが使用されます。

X

XR (エクステンデッドリアリティ): 拡張現実 (AR)、仮想現実 (VR)、複合現実 (MR) など、あらゆる形態のコンピューター改変 現実を包括する用語。

XR Interaction Toolkit: 対話的な XR アプリケーションを構築する ためのツールとコンポーネントのセットを提供する Unity パッケージ。

か

ガベージコレクション: アプリケーションで不要になったオブジェ クトに割り当てられたメモリを再利用するプロセス。ガベージコレ クションを頻繁に実行すると、パフォーマンスのスパイクが発生する 可能性があります。

<

クロスプラットフォーム開発: 複数のハードウェアプラットフォームで、 ほとんど、またはまったく変更を加えることなく実行できるアプリケー ションを開発すること。

さ

最適化:リソース消費を削減し、パフォーマンスを向上させることで、 アプリケーションをより効率的に実行するプロセス。

そ

速度トラッキングの動き:Unity の XR Interaction Toolkit の機能。 インタラクターの速度に基づいてオブジェクトの動きを計算し、 リアルな物理的現象のインタラクションを可能にします。

τ

テクセル密度: モデルに適用されるテクスチャピクセル (テクセル) の密度のことを指し、テクスチャのビジュアル品質とパフォーマンス に影響を与えます。



と

トリムシート: さまざまなディテールを含む 1 つのテクスチャを使用 するテクスチャリング手法。これを 3D モデルのさまざまなパーツ に適用することで、複数のテクスチャを使用せずに複雑さを加える ことができます。

ドローコール: グラフィックスをレンダリングするために GPU に 送信されるコマンド。ドローコールの回数を減らすと、グラフィック パフォーマンスが大幅に向上する可能性があります。

は

パススルー: ユーザーがデバイスのカメラを通じて周囲の現実 世界を見ることができる VR デバイスの機能。

パフォーマンスの最適化: アプリケーションのスピードと効率性を 高める手法。VR/AR で高いフレームレートとスムーズな動作を維持 するために不可欠です。

ふ

フレームレート:1秒間に表示されるフレーム数。安定した高い フレームレートは、スムーズなゲームプレイと VR 体験に不可欠 です。

へ

ベイクしたライティング: テクスチャマップに格納された事前に計算 されたライティング情報。リアルタイムのライティング計算の必要性 を排除することでグラフィックアプリケーションのパフォーマンスを 強化するために使用されます。

ほ

法線マップ:ポリゴン数を増やすことなくディテールが豊かな表面を シミュレートし、表面に深度とディテールを加えるテクスチャタイプ。

め

メモリ使用量:アプリケーションが使用している RAM の容量。メモリ 使用量の最適化は、特に VR などのリソースが限られた環境に おいて、パフォーマンスのために不可欠です。

り

リギング: 3D モデルのボーン構造体を作成して、リアルな動きを 実現するプロセス。

れ

レンダーパイプライン:</mark>グラフィックスが処理されて画面上に表示 されるプロセス。Unity は、URP や HDRP など、複数のレンダー パイプラインを提供しています。

ゆ

ユーザーインタラクション: コントローラー、ジェスチャ、音声コマ ンドなど、ユーザーがゲームやアプリケーションを操作する方法。



unity.com