UNREAL ENGINE

クラウド ソリューション

UNREAL ENGINE コンテナの機能の詳細

ホワイト ペーパー

画像提供：Michael Wallace 氏

目次

はじめに **3**

Unreal Engine コンテナ **4**
- コミュニティによって維持されてきたコンテナのサポートの歴史 4
- Unreal Engine におけるコンテナの公式サポート 5
- コンテナ イメージのタイプ 5

ユース ケース **6**
- 継続的インテグレーションおよび継続的デプロイメント (CI/CD) 6
- Pixel Streaming 7
- AI および機械学習 8
- リニア メディアをレンダリングする 9
- 没入型のビジュアライゼーション 10
- リモート バーチャル デスクトップ 11

次のステップ **12**

まとめ **12**

はじめに

クラウド コンピューティングとクラウドネイティブの開発ワークフローは、最新のソフトウェアの作成方法やデプロイ方法に変革をもたらしました。クラウド プラットフォームにより、デベロッパーはコンポーネントを購入したりアップグレードしたりするコストをかけずに最新のハードウェアにアクセスできるようになり、デプロイメントは需要に応じて柔軟なスケーリングが可能です。また、GPU などのハードウェアが一元的に利用可能になったことで、3D グラフィックや機械学習を利用したエクスペリエンスを、フォーム ファクターや計算能力にかかわらずあらゆるデバイスに提供できるようになりました。

クラウドネイティブのテクノロジーによって、デベロッパーはクラウドの可能性を最大限に活用し、ツールや開発ワークフローの利用により、極めて堅牢かつかつてない規模で稼働するソリューションの構築を実現しています。このようなツール エコシステムの中核となる、基礎的なクラウドネイティブ テクノロジーの 1 つがコンテナです。そして、最新の開発ワークフローの力を活用するカギとなるのがコンテナ テクノロジーとの緊密な統合です。

コンテナは、ソフトウェアとそのサポート環境を軽量かつ移植可能な形式でカプセル化し、オン プレミスでもクラウドでもインフラストラクチャをまたいでデプロイすることができます。また、仮想マシンなどの従来のテクノロジーよりもはるかに効率的かつコスト効率の高い方法によって、内部でソフトウェアを実行する一貫性ある再現可能な環境を提供します。この効率化は、仮想マシンで行われるようにハードウェア スタックを完全に仮想化し、デプロイメントの密度を最大化しながら運用のオーバーヘッドを低減するのとは異なり、共通のファイル システムのデータおよびオペレーティング システムのカーネルをコンテナ間で共有することで実現されています。

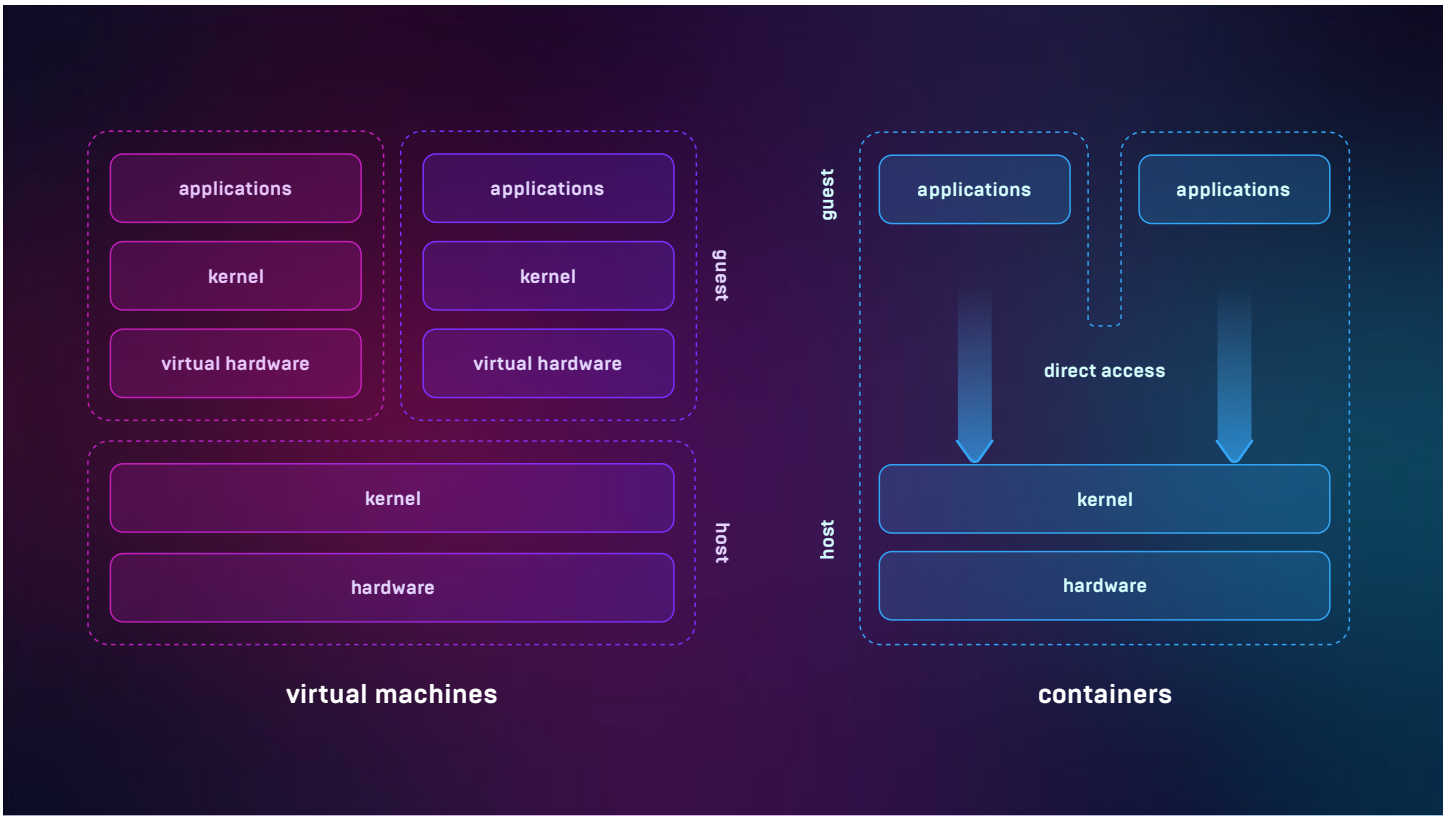


図 1：仮想マシンとコンテナの分離アプローチの違いを示した比較。

Unreal Engine のような最新の 3D レンダリングおよびシミュレーション エンジンとコンテナを組み合わせることで、革新的な新しいユース ケースやクラウドベースのワークフローが可能になり、幅広い業界で有効な次世代のソリューションが生まれます。ゲーム開発や映画/テレビなどのクリエイティブ業界から、トレーニングやシミュレーションなどの研究や企業アプリケーションに至るまで、多くの Unreal Engine プロジェクトでコンテナを使用するメリットを得ることができます。

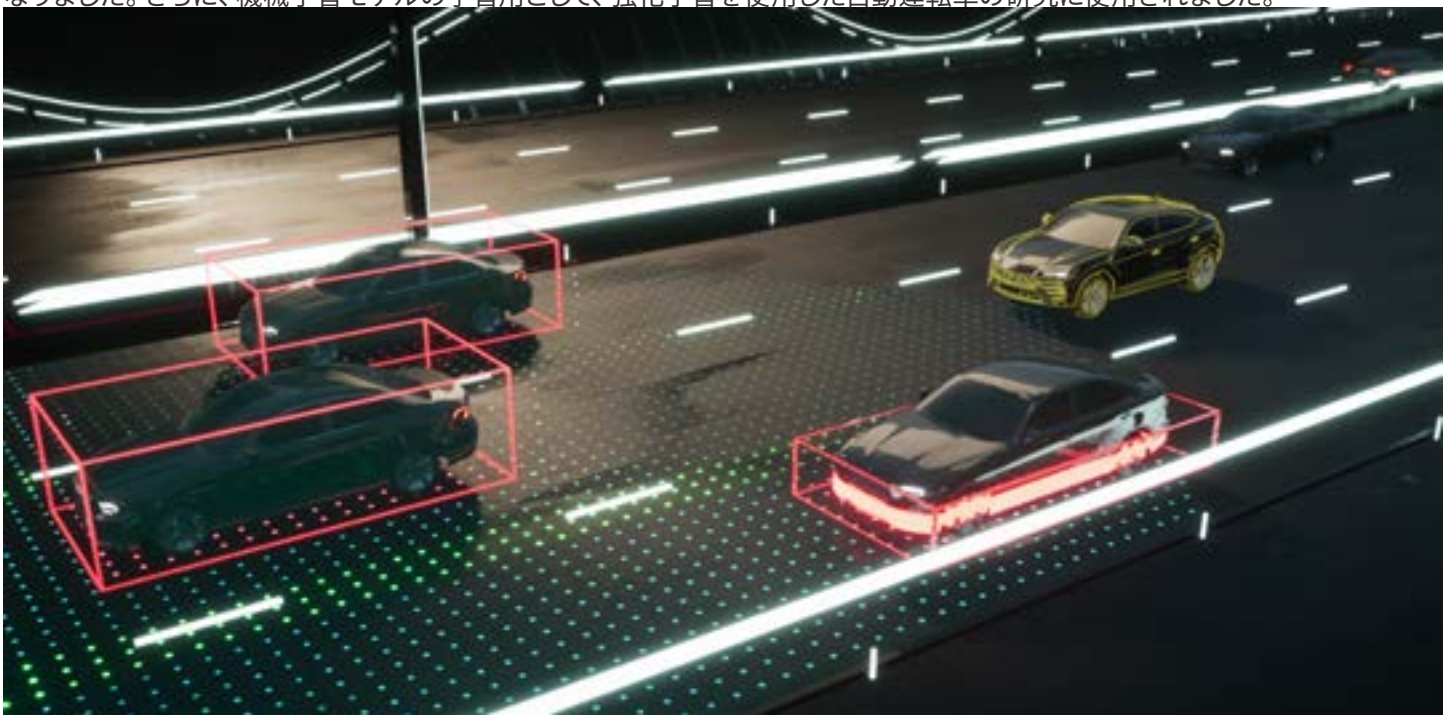
このホワイト ペーパーでは、Unreal Engine 4.27 でのコンテナの公式サポートの導入について説明した後、数多くあるユースケースの中から Unreal Engine コンテナを使用することで変革を実現した例をいくつかご紹介します。まとめでは、デベロッパーが Unreal Engine のコンテナを自社のアプリケーションで活用する方法について詳細情報を記載しています。

Unreal Engine コンテナ

コミュニティによって維持されてきたコンテナのサポートの歴史

Unreal Engine のコンテナのサポートは、オープン ソースの
ue4-docker プロジェクトにおいて、2018 年にコミュニティによって初めて行われました。このプロジェクトでは、クラウド環境で最も広く使用されているプラットフォームである Windows と Linux 向けに、Unreal Engine 用の開発コンテナ イメージをビルドするためのコードが提供されました。ue4-docker プロジェクトでは、より古いバージョンの Unreal Engine のソースコードに対するパッチもさまざまな方法で提供され、コンテナベースの環境との互換性改善も行われましたが、Unreal Engine 4.26 ではこれらのパッチは不要となりました。

ue4-docker の最初のリリースから数か月も経たないうちに、初期の利用者はプロジェクトやプラグインをビルドするための再現可能な環境として Unreal Engine コンテナを使用しはじめました。また、デベロッパーは、クラウド環境でリアルタイムのレンダリングを実行する用途に向けて Unreal Engine コンテナの実験を開始しました。2019 年までには、Unreal Editor を Linux デベロッパーのワークステーションにデプロイするために、また Unreal Engine の Pixel Streaming とよく似たサードパーティ プラグインを使用してリアルタイムのビデオを Web ブラウザにストリーミングするためにコンテナが利用されるようになりました。さらに、機械学習モデルの学習用として、強化学習を使用した自動運転車の研究に使用されました。



画像提供：Lamborghini Urus – アダプティブ インテリジェンス提供：WE ARE HAPPY

増え続ける Unreal Engine コンテナのユース ケースをドキュメントにまとめるため、ue4-docker のリード デベロッパーである Adam Rehn 博士は Unreal Containers コミュニティ ハブ を作成しました。コンテナ内で Unreal Engine を使用するデベロッパーに向けたドキュメントおよびトレーニング リソースの一元的なソースとして機能するこのハブは、2019 年 7 月に Epic Games MegaGrant を獲得しました。獲得以降もコミュニティ ハブが一般向けに公開するリソースは増加し続け、プロダクション環境でコンテナをデプロイするデベロッパーの増加に伴って Unreal Engine コンテナの採用も安定的に増加しています。

Unreal Engineにおけるコンテナの公式サポート

Epic Games では、 TensorWorks とのコラボレーションによって Unreal Engine 4.27 でコンテナの公式サポートを実現しました。TensorWorks は、ue4-docker プロジェクトや Unreal Containers コミュニティ ハブなどのコミュニティ イニシアチブに参加するデベロッパーを採用しています。継続的なコラボレーションに取り組む中で、Epic Games と TensorWorks は、Unreal Engine 4.27 と次期の Unreal Engine 5 においてコンテナのサポートを最優先事項として作業を進めています。

Unreal Engine の公式コンテナ イメージは、実証済みのオープンソース コードである ue4-docker プロジェクトおよび ue4-runtime プロジェクトに基づいて構築されています。Unreal Engine 4.27 における公式コンテナ イメージのリリース プロセスは新しく、コンテナのサポートはベータ機能とみなされるものの、中核となるテクノロジーは成熟した堅牢なものであり、すでに世界中のデベロッパーがプロダクションで利用しています。

Unreal Engine におけるコンテナの公式サポートは、次のメリットを提供します。

- **コンテナの互換性：**Unreal Engine の以前のリリースでは、コンテナベースの環境との互換性を確保するために ue4-docker プロジェクトのパッチが必要でした。Unreal Engine 4.27 以降では、すべての新規リリースでコンテナの互換性がテスト済みであり、パッチが必要ないことが確認されています。
- **事前ビルド済みコンテナ イメージ：**デベロッパーが Unreal Engine 用のコンテナ イメージをビルドするために使用できるソース コードの提供に加え、Epic Games では、Unreal Engine 4.27 以降向けの 事前ビルド済み公式コンテナ イメージ を提供しています。これらの事前ビルド済みのイメージには、 エンジンの Installed Build を含む開発イメージが含まれており、これは Windows および macOS の Epic Games Launcher から利用できるものと同じタイプのバイナリです。公式コンテナ イメージにアクセスするには、 Epic Games アカウントが GitHub とリンク済みである ことを確認する必要があります。
- **Unreal Developer Network (UDN) によるプレミアム サポート：**Epic Games のお客様で カスタムの Unreal Engine ライセンスをお持ちの場合、または Unreal Enterprise Program に登録されている場合は、 Unreal Developer Network (UDN) による Unreal Engine コンテナのプレミアム サポートにアクセスしていただけます。このサポートは、オープン ソース コミュニティによって提供される既存のサポート チャンネルを補完するものです。

コンテナ イメージのタイプ

Unreal Engine 4.27 では、2 つの異なるタイプのコンテナ イメージに対するサポートが含まれており、それぞれのサポートが以下のように異なるユース ケースに対応します。

開発イメージ には、Unreal Editor とビルド ツールが含まれています。これらは、Unreal Engine プロジェクトとプラグインのビルドとパッケージ化、シネマティックスのレンダリング、コマンドレットの実行など、Unreal Editor を必要とするタスクに使用します。開発イメージの配布は、Unreal Engine EULA の条件に準拠しています。

ランタイム イメージ には、パッケージ化された Unreal Engine プロジェクトの実行に必要な依存関係のみが含まれています。デベロッパーは、パッケージ化された Unreal Engine プロジェクトのファイルを追加し、クラウド環境にデプロイできる新しいコンテナ イメージを作成することで、ランタイム イメージを拡張できます。ランタイム イメージには Unreal Editor またはビルドツールが含まれていないため、配布の際は開発イメージよりも制限が少なくなります。

開発イメージとランタイム イメージの違いの詳細については、Unreal Containers コミュニティハブ ドキュメントの「Development images vs. runtime images」のページを参照してください。

ユース ケース

継続的インテグレーションおよび継続的デプロイメント (CI/CD)

図 2：Unreal Engine 向けの CI/CD システムである Admiral は、Epic Games MegaGrant を獲得し、Unreal Engine コンテナを最大限に活用しています。画像提供：TensorWorks。

継続的インテグレーションおよび継続的デプロイメント (CI/CD の総称で知られる) は、最新の開発プラクティスのセットであり、堅牢なソフトウェアをタイムリーに提供するためのカギとして実績があります。CI/CD ワークフローは、繰り返し可能かつ再現可能な環境を利用することでメリットを得るものであり、その環境内でビルドを実行し、環境の構成による問題に関連した整合性の改善およびエラーの回避を行います。このような再現可能な環境として、コンテナは一般的な選択肢です。また、最新の CI/CD ツールの多くはコンテナの使用に特化して設計されています。

Unreal Engine コンテナは、既存のツールやソフトウェア業界のベスト プラクティスに対応することで、Unreal Engine ベースのプロジェクトやプラグインを作成するデベロッパーが CI/CD のプラクティスの採用を簡素化するのに役立ちます。開発コンテナ イメージにより、マシンをビルドする再現可能な環境のデプロイ プロセスが簡素化されます。また、新しいバージョンのエンジンがリリースされた際にこれらの環境に対する以降のアップデートのデプロイも簡素化されます。個々の Unreal Engine コンテナ インスタンスによって提供される一時的な環境では、ビルド プロセスの完了時に自動的にあらゆるリソースおよびファイルシステムの状態が確実にクリーンアップされ、信頼性の高いカプセル化および再現性を実現します。

小規模な独立系ゲームから、大規模なエンタープライズ アプリケーションや AAA タイトルまで、あらゆる規模の Unreal Engine のプロジェクトで CI/CD のプラクティスのメリットを得ることができます。Unreal Engine のデベロッパーによる CI/CD のプラクティスの活用方法については、GDC 2018 のプレゼンテーション、「Adopting Continuous Delivery in Sea of Thieves」を参照してください。このプレゼンテーションでは、デベロッパーである Rare が、自社の開発ワークフローにおいて CI/CD のプラクティスを採用し、高レベルの品質と堅牢性を維持しながらゲーム Sea of Thieves に迅速なアップデートを配信可能にした方法が説明されています。

Unreal Engine コンテナを使用して CI/CD を実践するための詳細については、Unreal Containers コミュニティ ハブのドキュメントの「Continuous Integration and Deployment (CI/CD)」のページを参照してください。

Pixel Streaming

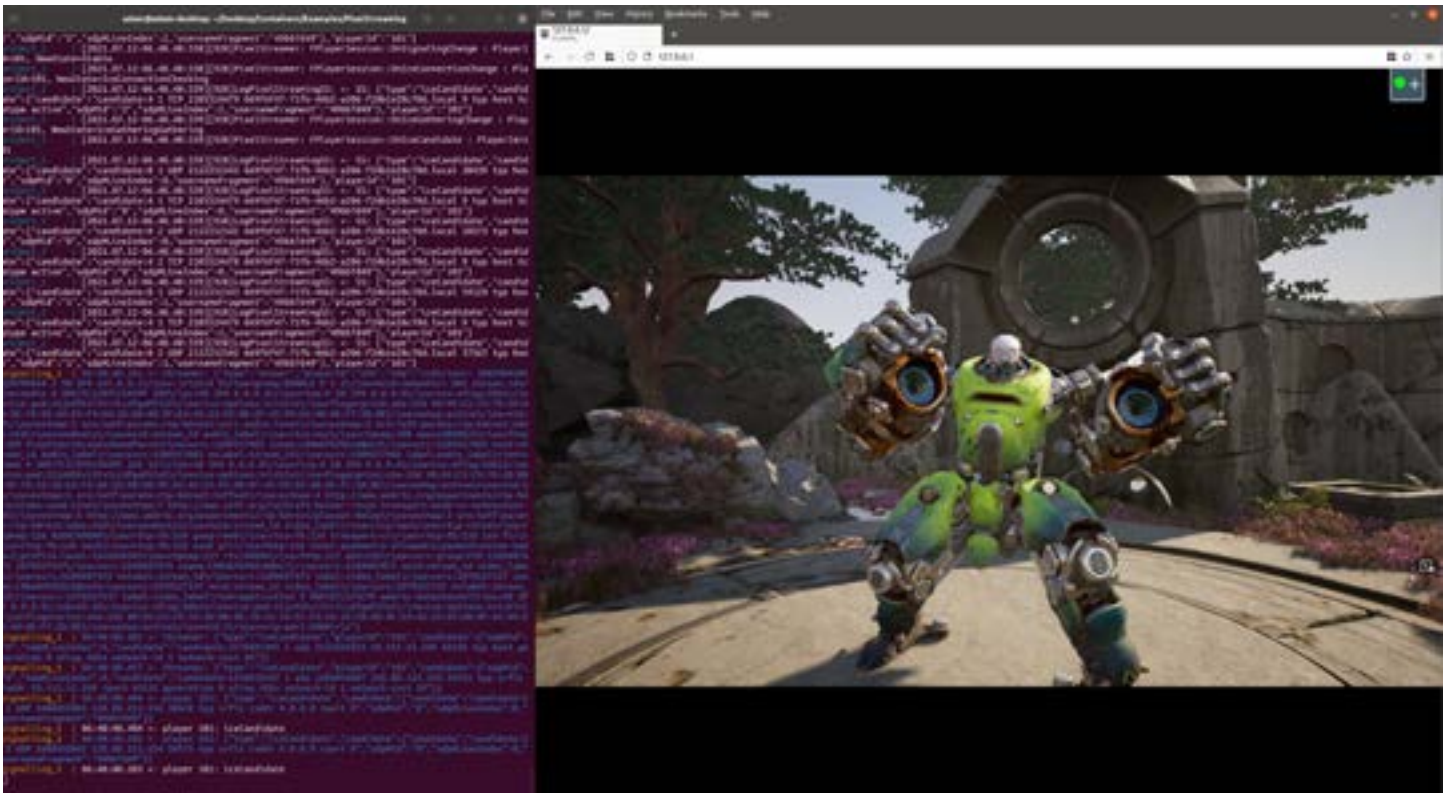


図 3：Unreal Engine コンテナ内で実行中の Pixel Streaming デモ プロジェクト。画像提供：TensorWorks。サンプル プロジェクト提供：Epic Games。

Unreal Engine の Pixel Streaming システムを利用すると、クラウド サーバー上でレンダリングを実行し、その結果を Web 経由でエンド ユーザーにリアルタイムでストリーミングすることにより、高い忠実度のインタラクティブなエクスペリエンスをあらゆるデバイスに提供できます。Unreal Engine 4.27 以前は、仮想マシンなどの従来のテクノロジーを使用して Pixel Streaming アプリケーションをクラウドにデプロイすることが必要でしたが、仮想マシンではオーバーヘッドが大きく柔軟性も限られるため、大規模なデプロイメントではコストと複雑さが増大します。

Unreal Engine 4.27 以降では、Unreal Engine コンテナを使用したデプロイメントが Pixel Streaming システムによってサポートされています。公式ランタイム コンテナ イメージを使用することで、パッケージ化された Pixel Streaming アプリケーションを必要なコンポーネントとともに簡単にカプセル化し、クラウドからストリーミングできます。また、既存のコンテナとの連携テクノロジーを備えた相互運用性によって、オンデマンドでのデプロイメントの動的スケーリングが簡素化されます。Unreal Engine コンテナを使用すると、Pixel Streaming アプリケーションのデプロイおよび運用上のコストや複雑さが低減され、リアルタイムのストリーミングがすべてのデベロッパーにとって利用しやすいものになります。

Pixel Streaming を利用することで実現できるエクスペリエンスの例については、 Project Anywhere のデモまたは MetaHuman Creator クラウド アプリケーションを参照してください。

Unreal Engine コンテナを使用して Pixel Streaming アプリケーションをデプロイするための詳細については、Unreal Containers コミュニティ ハブのドキュメントの Pixel Streaming のページを参照してください。

AI および機械学習

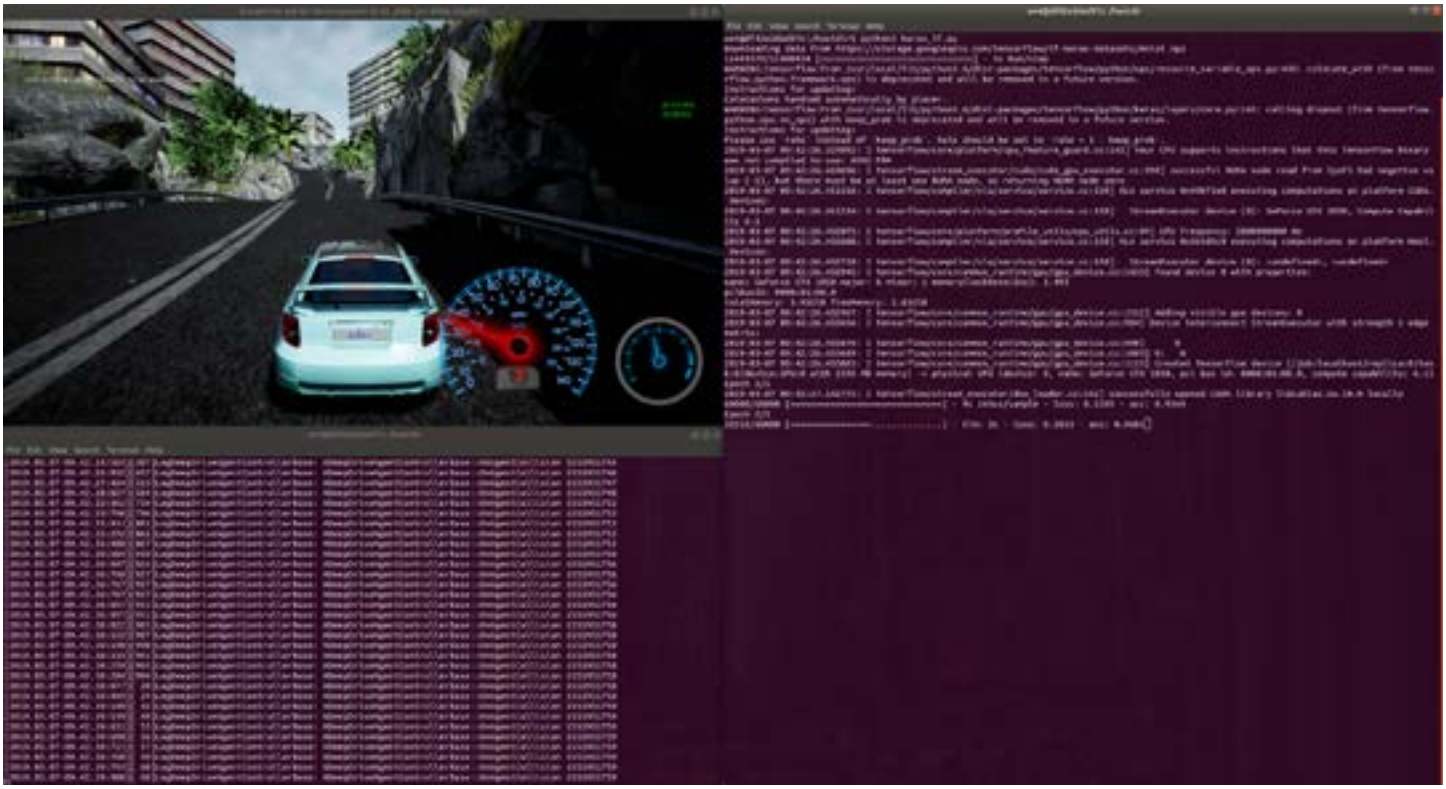


図 4：機械学習ワークロードと並行して Unreal Engine 内で実行中のオープン ソースの自動運転車シミュレーター、Deepdrive。画像提供：TensorWorks。

AI 業界では、機械学習モデルに対してシミュレーション ベースの学習を行わせることが急速に広まっており、特にこれは自動運転車のテストや研究の分野で顕著です。Unreal Engine のような最新のゲーム エンジンによって生成されたフォトリアルなイメージを利用すると、他の方法では不可能な規模で AI の学習を実現できます。実際、機械学習モデルの学習は、科学研究の分野における Unreal Engine の用途として最もよく利用される用途の 1 つであり、このことは Unreal Engine For Research イニシアチブ によって収集された 学術文献のリスト にもよく表れています。

機械学習モデルの大規模な学習は、機械学習アルゴリズムを加速させるために多数のグラフィック カード (GPU) が利用できるクラウドまたは大規模なオンプレミスのデータ センターでよく行われます。機械学習のワークロードは、通常 GPU アクセラレーションを有効にしたコンテナにデプロイすることで、クラウド環境全体での再現性と移植性が確保されます。Unreal Engine コンテナでは同じクラウドネイティブのツールを使用して機械学習ワークロードと並行してシミュレーション ワークロードやレンダリング ワークロードをデプロイできるため、モデルの学習用イメージの生成に Unreal Engine を使用するとデプロイメントが大幅に簡素化されます。

Unreal Engine コンテナを活用したオープン ソースの自動運転車シミュレーターの例については、GitHub の Deepdrive プロジェクト、 CARLA プロジェクト、および AirSim プロジェクトを参照してください。

機械学習に Unreal Engine コンテナを使用する詳細については、Unreal Containers コミュニティ ハブのドキュメントの「AI and Machine Learning」のページを参照してください。

リニア メディアをレンダリングする

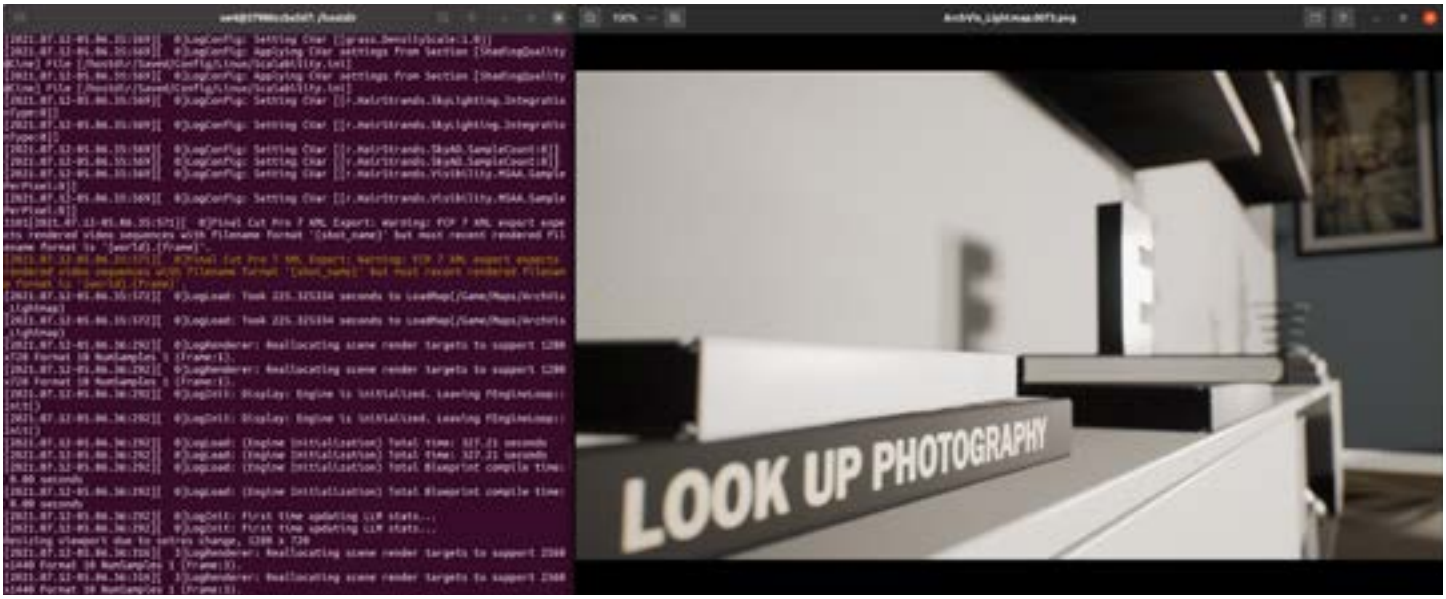


図 5：Unreal Engine コンテナ内でバッチ レンダリングを実行中の Unreal Editor。画像提供：TensorWorks。サンプル プロジェクト提供：Epic Games。

Unreal Engine によってフォトリアルなビジュアルをリアルタイムでレンダリングできるようになったことで、映画やテレビなどのリニア メディアの制作にさまざまな革新的ワークフローが新たにもたらされました。バーチャル プロダクション ワークフローでのインタラクティブな使用方法に加え、このビジュアライゼーション機能は高品質なバッチ レンダリングや分散レンダリング向けにオン デマンドで強化することができます。統合のためにクラウド インフラストラクチャを使用し、これらのパイプラインの処理をスケーリングすることで、顧客にとっても、地理的に分散する昨今のチームにとっても、パイプラインはアクセスしやすいものになります。

Unreal Engine コンテナは、クラウド環境に対するレンダリング ワークロードのデプロイメントを簡素化します。また、オン デマンドのスケーリングに対応する、分散レンダリング システムを構築する上で最新のマイクロサービスによるソフトウェア アーキテクチャとの相互運用性を高めます。このような柔軟性により、既存のメディア プロダクション ワークフローを強化するソリューションを構築したり、高速かつ高品質なレンダリングをサービスとしてエンド ユーザーに提供したりすることが可能です。

リニア メディアのレンダリングに Unreal Engine コンテナを使用する詳細については、Unreal Containers コミュニティ ハブのドキュメントの「Rendering Linear Media」のページを参照してください。

没入型のビジュアライゼーション

図 6：カリフォルニア大学サン・ディエゴ校 Qualcomm Institute の SunCAVE。画像提供：UCSD。

LED ウォールや CAVE などの複数のパネルを集めて構成される没入型のディスプレイ システムは、建築ビジュアライゼーション、医療用画像、科学研究、および映画やテレビのバーチャル プロダクションなど、さまざまな業界で人気が高まっています。Unreal Engine の nDisplay システムは、マルチディスプレイのレンダリングを管理する高度な機能を提供します。また、 マンダロリアンのためにディズニーが開発したバーチャル プロダクション手法 など、革新的な新たなワークフローを実現しました。

Unreal Engine 4.27 からは nDisplay システムで Linux がサポートされるようになり、 Unreal Engine インスタンスのクラスタを実行して CAVE などの大規模なマルチディスプレイ システムを運用する際に Linux コンテナ内で使用することでデプロイメントを簡素化することができます。カリフォルニア大学サン・ディエゴ校の研究者は、Unreal Engine コンテナを使用して SunCAVE システムに表示される没入型のビジュアライゼーションをレンダリングしましたが、このシステムは 70 のディスプレイで構成され、 30 以上のコンピューターで運用されています。SunCAVE のマシンでは Kubernetes コンテナ オーケストレーション フレームワークを使用してレンダリング ワークロードおよび処理ワークロードが管理されます。また、動作はクラウド内の従来のコンピューティング クラスタと同様です。Unreal Engine コンテナと、Kubernetes などの標準的なコンテナ ツールをシームレスに統合することで、必要に応じてスケーリングに対応し、大規模なマルチディスプレイ システムの需要を満たすビジュアライゼーション ワークロードの迅速なデプロイメントが可能になります。

複数の物理ディスプレイに対する出力のレンダリングに Unreal Engine コンテナを使用する詳細については、Unreal Containers コミュニティ ハブのドキュメントの「Multi-Display Output」のページを参照してください。

リモート バーチャル デスクトップ

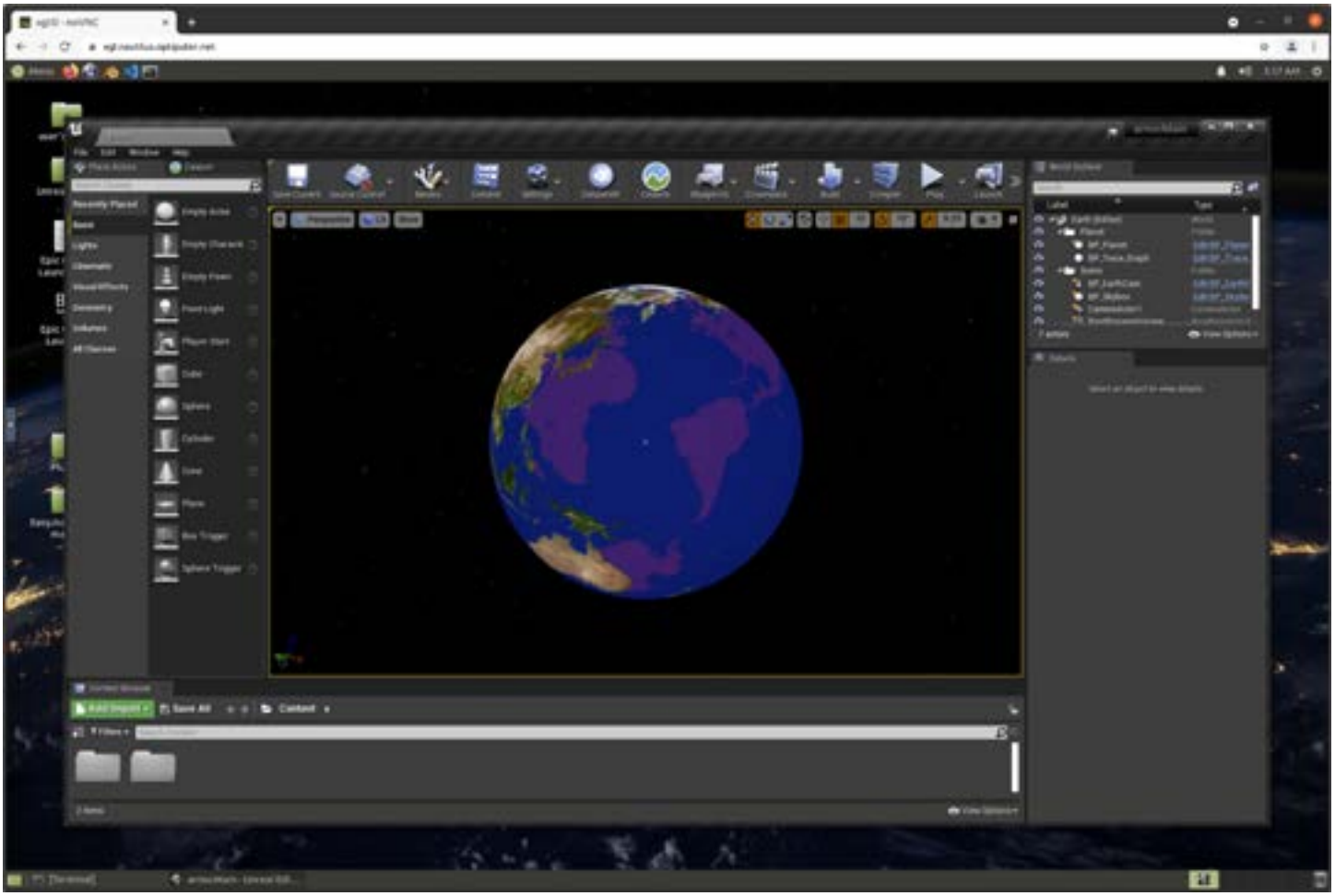


図 7：Pacific Research Platform の国際的 Kubernetes クラスタ、「Nautilus」でホストされ、仮想デスクトップ コンテナ イメージ内で実行中の Unreal Editor。画像提供：UCSD。

地理的に分散するリモートのチームが増加し、一般的になるにしたがって、クラウドでユーザーのワークステーションを統合しようと模索する組織がますます増えています。デスクトップ環境から仮想化されたクラウド デプロイメントに移行すると、ユーザーは一元的に管理およびプロビジョンされる高性能なハードウェアにリモートでアクセスできるようになるため、柔軟性が高まると同時に運用上のオーバーヘッドが低減されます。この種のインフラストラクチャは、高機能なソフトウェアを実行する計算リソースに対して他の方法ではアクセスできないユーザーにも役立ちます。テクノロジー プロバイダの Parsec は、このメリットを活かし、 新型コロナ感染症のパンデミックにおいて Unreal Engine を搭載したワークステーションに対する無料のリモート アクセスを学校に提供したことで Epic Games MegaGrant を獲得しました。

従来、仮想デスクトップ インフラストラクチャは仮想マシン上に構築されていましたが、このようなユース ケースでは、コンテナが実現する効率とスケーラビリティによって大きなメリットがもたらされます。米国のカリフォルニア大学サン・ディエゴ校および韓国の延世大学校医科大学の研究者は、Unreal Editor を含むグラフィック アプリケーションを備え、GPU アクセラレーションを全面的に活用する Linux デスクトップ環境のカプセル化に対応する コンテナ イメージ を開発しました。両大学は、このコンテナ イメージを使用し、学生や研修者にオンデマンドで動的にプロビジョニングされる仮想デスクトップリモート アクセスを提供しています。

このリモート デスクトップのコンテナ イメージの作成に関する研究プロジェクトの詳細については、Seungmin Kim 氏による資料「GPU Remote Desktop Container Project」を参照してください。

次のステップ

Unreal Engine コンテナの利用を開始するには、Unreal Engine ドキュメントの「コンテナ」のページを参照してください。

Unreal Engine コンテナのしくみと、実現可能なユース ケースの詳細については、Unreal Containers ハブのドキュメントの「Quickstart Guide」を参照してください。

まとめ

コンテナとクラウドネイティブのワークフローは、最新のソフトウェアの開発方法とデプロイ方法に変革をもたらしました。Unreal Engine 4.27 におけるコンテナの公式サポートの導入は、クラウドネイティブの開発による機能をすべての Unreal Engine ユーザーに提供し、オープン ソース インフラストラクチャの堅牢かつ実証済みの基盤上での構築を可能にします。

Unreal Engine による 3D レンダリングやシミュレーション エンジンとコンテナを組み合わせて既存のユース ケースを強化することで、革新的な新しい利用方法やワークフローを実現する道が拓け、幅広い業界で有効な次世代のクラウド ソリューションの誕生につながります。Unreal Engine コンテナは、ゲーム開発、映画やテレビ、研究、エンタープライズ、教育などに携わるクリエイターやプロフェッショナルをサポートします。

コンテナの公式サポートは、Unreal Engine のクラウドネイティブ機能の継続的統合における最初の一歩にすぎません。この機能は Unreal Engine 5 以降でも継続的に拡張され、かつてない可能性を実現して各業界に変革をもたらすことで、Unreal Engine は新しい世代のアプリケーションおよびワークフローに不可欠なプラットフォームとなるでしょう。

このドキュメントについて

著者
Adam Rehn

寄稿者
Luke Bermingham

編集者
Michele Bousquet

レイアウト制作者
Jung Kwak

Copyright © 2021 Epic Games, Inc. All rights reserved.