



AWS クラウド への移行中の COTS および社内アプリケーションの再プラットフォーム化

AWS 規範ガイドンス



AWS 規範ガイド: AWS クラウド への移行中の COTS および社内アプリケーションの再プラットフォーム化

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

序章	1
ターゲットを絞ったビジネス成果	2
リプラットフォーム環境の選択	4
サポートされていない OS で実行されているアプリケーションコンポーネントの再プラット フォーム化	5
サポートされていない OS またはアプリケーションサーバーの置き換え	6
COTS アプリケーション用の OS のアップグレード	6
社内アプリケーション用の OS のアップグレード	6
アプリケーションライブラリと依存ソフトウェアの再プラットフォーム	7
バックエンドデータベースの再プラットフォーム	8
COTS アプリケーション用のバックエンドデータベースの再プラットフォーム	8
社内アプリケーション用のバックエンドデータベースの再プラットフォーム	9
ファイル共有の再プラットフォーム	10
ロギングおよびモニタリングコンポーネントの更新	11
アプリケーションのテストと検証	12
進行中の OS パッチの自動化	13
オートメーションとインフラストラクチャをコードとして使用する (iaC)	13
よくある質問	14
リホストする代わりに再プラットフォームを選択する必要があるのはいつですか？	14
COTS アプリケーションをオープンソース・データベースにリファクタリングすることはでき ますか？	14
サーバーを AWS クラウドにすばやくリホストするには、どのような AWS ツールを使用でき ますか？	14
アプリケーションのリプラットフォームにはどのような AWS ツールを使用できますか？	15
リソース	16
リファレンス	16
ツール	16
ドキュメント履歴	17
.....	xviii

AWS クラウドへの移行中の COTS と社内アプリケーションの再プラットフォーム

Anbu Selvan、Amazon Web Services (AWS)

2021 年 3 月 ([ドキュメント履歴](#))

このガイドでは、Amazon Web Services (AWS) クラウドで市販の既製品 (COTS) および社内アプリケーションを再プラットフォームする際に重点を置く 7 つの領域について説明します。このガイドでは、アプリケーションコンポーネントの再プラットフォームに役立つ戦略、ツール、AWS サービスも提供しています。COTS アプリケーションは、既製で、商用市場で購入できるサードパーティ製のアプリケーションです (例えば、[AWS Marketplace](#))。社内アプリケーションは、組織によって内部的に開発され、使用されます。

COTS または社内アプリケーションを に移行する場合は AWS クラウド、7 つの一般的な移行戦略 (7 R) のうちどれを使用するかを評価する必要があります。これらの戦略は、リファクタリング、再プラットフォーム、再購入、リホスト、再配置、保持、および使用停止です。サポート終了 (EOS) の日付に達した、またはそれに近いコンポーネントまたはデータベースを使用するアプリケーションを再プラットフォーム化することをお勧めします。EOS は、ベンダーが製品のテクニカルサポートを取り下げたときです。AWS クラウドでアプリケーションを再プラットフォームすることを選択した場合は、次の機能を利用できます。

- [デインプレースオペレーティングシステム \(OS\) のアップグレード](#)を自動化します AWS Systems Manager。
- Amazon EC2 (Amazon Elastic Compute Cloud) インスタンスから Amazon マシンイメージ (AMI) をすばやく作成するには、スナップショットストレージボリュームを使用します。
- プライベートサブネットを作成して、古いオペレーティングシステム (OS) で実行されるワークロードを分離します。
- 高速ネットワークを使用して、再プラットフォームをテストするための本番環境を迅速にレプリケーションします。
- 追加のオンプレミスハードウェアを使用せずに、オンデマンド EC2 インスタンスを使用して個別のアプリケーションスタックをすばやくセットアップできます。

AWS クラウドで利用できるこれらの機能やその他の機能を活用するには、まず を使用してアプリケーションをリホストすることをお勧めします [AWS Transform MGN](#)。その後、AWS クラウドでア

アプリケーションをアップグレードできます。次のリストに、アプリケーションを再プラットフォーム化するタイミングの例を示します。

- アプリケーションの OS、ランタイム (例えば、[Apache Tomcat](#)、[JBoss](#)、または [Oracle WebLogic Server](#))、データベース、またはランタイムコンポーネント (Java、Python、Perl など) は、サポート対象外です。
- アプリケーションの復元力が向上し、障害 (ソフトウェアのバグやインフラストラクチャの問題など) から自動的に回復する必要があります。
- 新しい顧客セグメントや負荷の増加をサポートするには、新しいアプリケーション機能が必要です。
- アプリケーションは不安定であり、オペレーションの安定性を高めるための改善が必要です。

再プラットフォームを開始する前に、アプリケーションの機能に代わるものを検討する必要があります。例えば、独立系ソフトウェアベンダー (ISV) から SaaS (Software as a Service) ソリューションに置き換えることができるかどうかを評価してください。また、[Amazon Cognito](#)、[AWS Lambda](#)、Amazon [Amazon MQ](#)、Amazon [AWS Glue Quick](#)、[Amazon Aurora](#) などの AWS サービスを使用してアプリケーション機能を再構築することもできます。

このガイドは、IT 管理者、アプリケーション所有者、アーキテクト、テクニカルリード、プロジェクトマネージャを対象としています。このガイドでは、AWS クラウドで COTS と社内アプリケーションを再プラットフォームするときに焦点を当てる次の 7 つの領域について説明します。

- [リプラットフォーム環境の選択](#)
- [サポートされていない OS で実行されているアプリケーションコンポーネントの再プラットフォーム化](#)
- [バックエンドデータベースの再プラットフォーム](#)
- [ファイル共有の再プラットフォーム](#)
- [ロギングおよびモニタリングコンポーネントの更新](#)
- [アプリケーションのテストと検証](#)
- [進行中の OS パッチの自動化](#)

ターゲットを絞ったビジネス成果

AWS クラウドで COTS と社内アプリケーションをリプラットフォームした後、次の 4 つの結果が期待されます。

- サポートされていないソフトウェアまたは OS を実行するレガシーアプリケーションによるセキュリティリスクを軽減します。
- 高価で必須ではないデータベースエディションを削除したり、オープンソースデータベースを採用することで、アプリケーション全体の所有コストを削減できます。
- AWS マネージドデータベース ([Amazon Relational Database Service \(Amazon RDS\)](#) や Aurora など) を使用して、アプリケーションの可用性と信頼性を高めることで、運用上のオーバーヘッドを削減します。
- [Amazon CloudWatch](#) モニタリングまたは Systems Manager ベースの OS のパッチ適用など、クラウドネイティブのオートメーションとモニタリング機能を採用することで、レガシーアプリケーションの復元力を高めます

リプラットフォーム環境の選択

[Amazon Elastic Block Store \(Amazon EBS\) スナップショット](#)や [EC2 インスタンスのクローン作成などの機能を使用して AMI を作成するには](#)、AWS クラウドでアプリケーションを再プラットフォームすることをお勧めします。これらの機能は、アップグレードやテストプロセスに役立ちます。

通常、まずアプリケーションを AWS クラウドにリホストしてから、リプラットフォームプロセスを開始します。ただし、脆弱性を持つアプリケーションを移行したり、サポートされていないソフトウェアや OS を実行したりする場合は、注意が必要です。これらのアプリケーションは、セキュリティの脆弱性を露呈する可能性があり、移行や今後のオペレーションに危険です。代わりに、[MGN](#)を使用して、出力が制限されたプライベートサブネットにアプリケーションコンポーネントをレプリケートすることをお勧めします。このアプローチにより、ワークロードが分離され、本番環境にデプロイされる前に安全に再プラットフォーム化およびテストを行うことができます。

組織が ISV から拡張 OS サポートを購入した可能性があります。これにより、OS のパッチ適用が正式な EOS の日付を超えて 3~5 年間延長されます。これは一時的な措置であり、製品のアップグレードまたは移行のタイムラインを COTS アプリケーションベンダーの製品リリーススケジュールにリファクタリングまたは調整するための追加の時間を提供します。ただし、[AWS プロフェッショナルサービス](#)または [AWS 移行コンピテンシーパートナー](#)の AWS ツールと専門知識を使用して、これらのワークロードを新しい OSs に再プラットフォームすることをお勧めします。これにより、高価な延長ライセンス契約を回避でき、移行をより迅速に完了できます。

サポートされていない OS で実行されているアプリケーションコンポーネントの再プラットフォーム化

サポートされていない OS で実行されるアプリケーションコンポーネントの再プラットフォームアプローチは、アプリケーションコンポーネントごとに異なります。次のテーブルに、EOS に達したアプリケーションコンポーネントで使用可能な再プラットフォームオプションの概要を示します。

アプリケーションコンポーネント	COTS アプリケーション向けソリューション	社内アプリケーション向けのソリューション
アプリケーションサーバー	アプリケーションのベンダーが推奨するバージョンにアップグレードします。	最新のアプリケーションサーバーのバージョンを特定します。アップグレードする前に、開発環境で構築して検証してください。
OS	アプリケーションのベンダーが推奨するバージョンにアップグレードします。	最新の OS バージョンを特定します。アップグレードする前に、開発環境で構築して検証してください。
ランタイム・ライブラリ	アプリケーションのベンダーが推奨するバージョンにアップグレードします。	最新バージョンにアップグレードし、検証します。
その他のアプリケーションコンポーネント	アプリケーションのベンダーに新しいアプリケーションバイナリを要求します。	最新の OS、ランタイム、およびアプリケーションサーバーバージョンを使用して構築します。

以下のセクションでは、アプリケーションコンポーネントの再プラットフォームアプローチに関する詳細を示します。

サポートされていない OS またはアプリケーションサーバーの置き換え

サポートされていないアプリケーションサーバー (Apache Tomcat 6.0、Apache 2.2、IIS 7.x など) を置き換える場合、新しいアプリケーションサーバーのバージョンでは、基盤となる OS のアップグレードが必要になる場合があります。サポートされていない OS の大半は、Red Hat Enterprise Linux (RHEL) バージョン 5 と 6、CentOS のバージョン 5 と 6、または Windows 2008 R2 です。これらの OS を実行するアプリケーションには、次の手順をデプロイする必要があります。

1. 必要な OS バージョンで EC2 インスタンスを起動します。
2. 必要なアプリケーションサーバーのバージョンをインストールします。
3. 社内アプリケーションと COTS アプリケーションには、次の 2 つのアプローチがあります。
 - 社内アプリケーション — EC2 インスタンスにアプリケーションを再デプロイします。
 - COTS アプリケーション : アプリケーションのベンダーに連絡し、必要な OS またはアプリケーションサーバーのバージョンで認定されたアプリケーションバイナリを要求します。

COTS アプリケーション用の OS のアップグレード

ほとんどの COTS アプリケーションベンダーは Windows 2016 または RHEL 7 をサポートしています。レガシー COTS アプリケーションが Windows 2016 をサポートしていない場合は、[マイクロソフトが提供するインプレースアップグレードオプション](#) を使用して Windows 2008 R2 から Windows 2012 R2 へのインプレースアップグレードをお勧めします。[AWS Systems Manager オートメーションランブック](#) EC2 インスタンスで実行されている Windows Server を自動的にアップグレードします。アプリケーションのベンダーに連絡し、最新の OS バージョンのソフトウェアの認証を依頼することをお勧めします。

社内アプリケーション用の OS のアップグレード

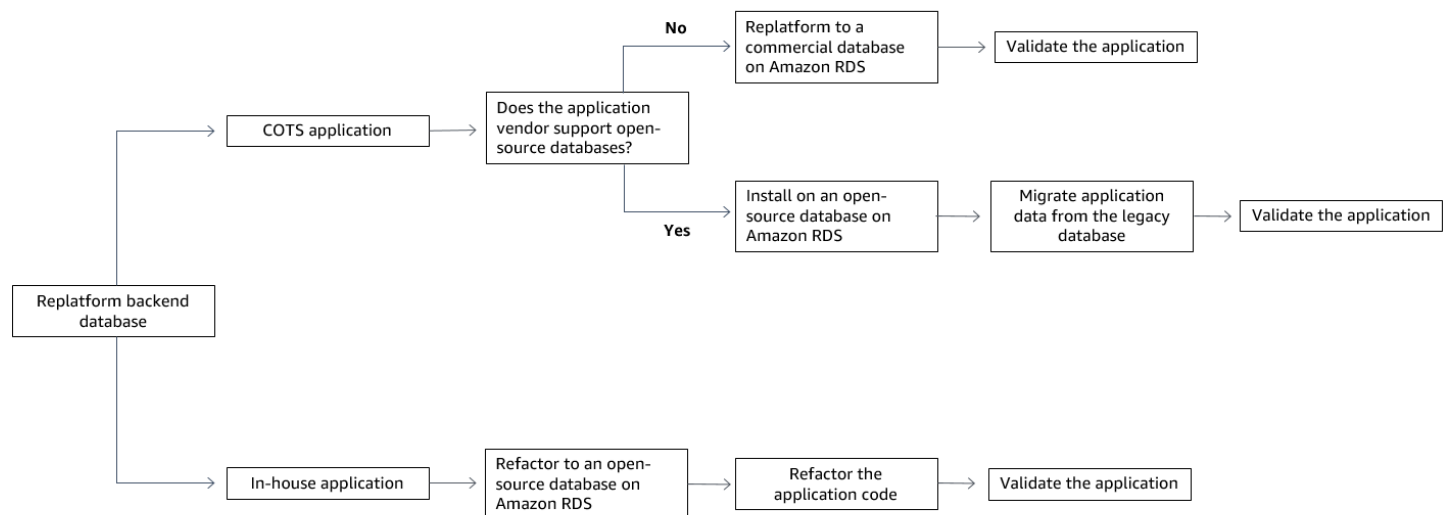
最新の OS およびソフトウェアランタイムバージョン (Java、C++、.NET、Python など) を使用して、社内アプリケーションのソフトウェアをコンパイルおよび再構築することをお勧めします。その後、本番環境にアップグレードする前に、既存のアプリケーション環境のクローンを作成し、機能を手動でデプロイして検証し、構築環境を最新の OS、ランタイムソフトウェアコンポーネント、およびライブラリに更新できます。

アプリケーションライブラリと依存ソフトウェアの再プラットフォーム

アプリケーションライブラリと依存ソフトウェアを再プラットフォームするアプローチは OS のアプローチに似ていますが、アップグレードするのはライブラリだけです。次に、アプリケーションの機能をテストし、プリプロダクションサーバーと本番サーバーに必要なライブラリを複製します。通常 COTS アプリケーションのベンダーは、継続的なソフトウェアリリースを通じてアプリケーションコンポーネントに必要な更新を処理します。

バックエンドデータベースの再プラットフォーム

バックエンドデータベースを再プラットフォームするアプローチは、COTS と社内アプリケーションでは異なります。これは通常ソースコードが社内アプリケーションでのみ使用できるためです。次の図は、アプリケーションのバックエンドデータベースで使用可能な再プラットフォームオプションを示しています。



次のセクションでは、COTS アプリケーションまたは社内アプリケーションに属するバックエンドデータベースの再プラットフォームアプローチについて説明します。

COTS アプリケーション用のバックエンドデータベースの再プラットフォーム

COTS アプリケーションがオープンソースデータベースをサポートしている場合は、Aurora データベースを使用することをお勧めします。オープンソースデータベースを使用すると、ライセンスコストの削減に役立ち、[AWS Schema Conversion Tool \(AWS SCT\)](#) および [AWS Database Migration Service \(AWS DMS\)](#) のようなツールも使用して、移行中のダウンタイムを最小限に抑えてカットオーバーを実現できます。

お使いの COTS アプリケーションがオープンソースデータベースをサポートしていない場合は、Amazon Relational Database Service (Amazon RDS) で [Amazon RDS for Oracle](#) または [Amazon RDS for Microsoft SQL Server](#) のような商用データベースに再プラットフォームすることをお勧めします。移行を開始する前に、アプリケーションで使用されるデータベース機能を評価し、

それらが Amazon RDS でサポートされていることを確認してください。詳細については、Amazon RDS ドキュメントの [Microsoft SQL Server データベースインスタンスの制限](#) を参照してください。

残りのデータベースライセンスを使用して、EC2 インスタンスで自己マネージド商用データベースを実行することもできます。この方法を選択した場合は、データベースのベンダーとライセンス検証プロセスを開始することをお勧めします。ライセンス検証プロセスが完了したら、アプリケーションの必要な目標復旧時間 (RTO) または目標復旧時点 (RPO) に対して、Amazon EC2 でマネージドデータベースの自己解決策を設計する必要があります。

最後に、SQL Server データベースを使用するセキュリティに配慮した高性能の COTS アプリケーションを Amazon EC2 Linux インスタンスで実行されている SQL Server に再プラットフォーム化することをお勧めします。詳細については、[オンプレミス SQL Server Windows ワークロードを Amazon EC2 Linux に移行する](#) を参照してください。

社内アプリケーション用のバックエンドデータベースの再プラットフォーム

社内アプリケーションのバックエンドデータベースを AWS マネージドデータベース ([Amazon RDS for PostgreSQL](#)、[Amazon RDS for MySQL](#)、[Aurora](#)、[Amazon DynamoDB](#) など) にリプラットフォームすることで、データベースのライセンスコストを削減し、スケーラビリティを向上させることができます。

AWS マネージドデータベースは、データベースの定期的な管理タスク (バックアップの実行、データベースおよび OS へのパッチ適用など) を削減するのに役立ちます。Amazon RDS マルチ AZ 配置を使用している場合は、データベースハードウェアの違反による停止を防ぐことで、アプリケーションの可用性を高めることもできます。マルチ AZ データベースは、異なるアベイラビリティーゾーンに継続的に複製され、停止時にアプリケーションは複製されたデータベースに透過的にフェールオーバーします。

AWS DMS とを使用して商用データベースを Aurora AWS SCT に変換できます。Amazon RDS はデータベーススキーマ変換プロセス AWS SCT を自動化し、オンプレミスデータベースから Amazon RDS へのデータレプリケーション AWS DMS を有効にします。AWS DMS または、オンプレミスアプリケーションをクラウドに移行するときにダウンタイムのカットオーバーを最小限に抑えることもできます AWS。

ファイル共有の再プラットフォーム

社内アプリケーションでネットワークファイルシステム (NFS) やサーバーメッセージブロック (SMB) などのオンプレミスの外部ファイル共有を使用している場合は、アプリケーションのコードを Amazon Simple Storage Service (Amazon S3) および [Amazon CloudFront](#) を使用するように変更することをお勧めします。ただし、ブロックベースのアクセスが必要な場合は、[Amazon Elastic File System \(Amazon EFS\)](#) または [Amazon FSx for Lustre](#) を使用できます。Amazon EFS と Amazon FSx は、共有ファイルストレージを必要とするアプリケーションの AWS オプションです。

COTS アプリケーションでは、オンプレミスのファイル共有から Linux 上の Amazon EFS または [FSx for Windows File Server の FSx](#) にデータを移行することをお勧めします。オンプレミスのファイル共有が公開されるのと同じ方法で、ブロックベースのファイル共有をアプリケーションに公開する必要があります。Amazon EFS および Amazon FSx は、オンプレミスのファイル共有と同じ機能を持つ NFS および SMB ベースのファイル共有を提供します。ファイル共有に Amazon EFS または Amazon FSx を使用するには、COTS アプリケーションの設定の変更のみが必要です。

ロギングおよびモニタリングコンポーネントの更新

一部のレガシー環境は、一元化されたロギングおよびモニタリングツール ([Splunk](#)、[SolarWinds](#)、[Zabbix](#) など) は、インフラストラクチャとアプリケーションの監視に使用します。アプリケーションサポートチームは、SSH (Secure Shell) プロトコルまたはリモートデスクトッププロトコル (RDP) を使用してモニタリングとデバッグをする場合もあります。この手動および反復的なプロセスを回避するには、[CloudWatch](#) を使用して AWS クラウドでのモニタリングを自動化できます。

CloudWatch メトリクスを使用してインフラストラクチャをモニタリングし、CloudWatch エージェントを使用してアプリケーションログを CloudWatch に送信することをお勧めします。CloudWatch がアプリケーションログを受信した後、[CloudWatch メトリックスフィルター](#)を作成し、[CloudWatch アラーム](#)を使用してアプリケーションエラーをモニタリングし、サポートチームに自動的に通知できます。

CloudWatch は、アプリケーションの運用オペレーションを継続的にレビューするための運用ダッシュボードを構築するためのツールも提供します。サードパーティーの一元化されたモニタリングツールは CloudWatch やその他の AWS サービスと統合できるため、既存の運用プラクティスを AWS クラウド上のインフラストラクチャやアプリケーションに拡張できます。ただし、AWS クラウドでアプリケーションを運用する場合は、サポートチームまたは運用チームを再トレーニングする必要がある場合があります。詳細については、AWS Cloud Adoption Framework (AWS CAF) ホワイトペーパーの「[Operations perspective: Manage and scale](#)」セクションを参照してください。

アプリケーションのテストと検証

機能テストとパフォーマンステストは、アプリケーションの再プラットフォームの過程の重要な部分です。通常、レガシーアプリケーションは、機能の詳細が正しくまたは完全に文書化されていないため、テストにはアプリケーション所有者の知識に依存しています。ただし、動作テストと自動テストを使用してアプリケーションのユースケースを記録することをお勧めします。このアプローチは、再プラットフォームの前後にアプリケーションの機能を迅速かつ確実に検証します。自動テストツール ([Selenium](#)、[Tricentis](#)、[Gatling](#) など) を使用して、機能テストとパフォーマンステストを構築できます。ベースライン結果は、現在のアプリケーション環境で機能テストとパフォーマンステストを実行して生成する必要があります。現在のアプリケーション環境とターゲットアプリケーション環境間のテスト結果を比較し、受け入れ基準として使用できます。

お客様向けアプリケーションでは、canary テストを使用することをお勧めします。Canary テストでは、本番環境で重要なアプリケーションワークフローを定期的にテストし、サポートチームにエラーを通知します。詳細については、AWS Well-Architected フレームワークの [Canary デプロイ](#) セクションを参照してください。

進行中の OS パッチの自動化

オンプレミスのデータセンター内のレガシーアプリケーションは、多くの場合、進行中の OS パッチ適用とソフトウェアの更新のために、手動の運用プロセスに依存しています。リプラットフォームの移行中は、[Systems Manager パッチマネージャー](#) を使用して OS のパッチ適用を自動化することをお勧めします。Patch Manager は、運用上のインサイトを収集し、AWS クラウドリソースとオンプレミスリソースの両方で日常的な運用タスクを実装するための一元化された一貫したプロセスを提供します。

本番環境で使用されるパッチ適用時間枠よりも早い段階で開発環境にパッチを適用することをお勧めします。この件の詳細は、OS パッチを自動化用 [パッチマネージャーのRunbook](#) を参照してください。また、カナリアテストを展開して、実稼働前または実稼働環境で主要なアプリケーション機能を定期的にテストし、テストが失敗した場合はサポートチームに警告する必要があります。これにより、アプリケーションの予期しない停止を回避できます。

オートメーションとインフラストラクチャをコードとして使用する (iaC)

アプリケーションのリプラットフォームの過程の一環として、[Chef](#)、[Puppet](#)、[Ansible](#) などの構成管理ツールを使用してプラットフォームのビルドを自動化する必要があります。これらのツールは、アプリケーションスタックの反復可能なビルドを可能にし、スタックの構成を含む、アプリケーションインスタンスを生成するための手順を形式化します。

iaC のベストプラクティスを使用してインフラストラクチャをプロビジョニングすることをお勧めします。これには [AWS Cloud Development Kit \(AWS CDK\)](#)、[CloudFormation](#)、[Terraform](#) など、いくつかのオプションがあります。Chef、Ansible、およびPuppetの機能も制限されているため、ユースケースに十分な自動化を実現できます。

iaC と構成管理コードを使用する反復可能なビルドは、それらのリソースを再構築するオーバーヘッドやリスクなしにインフラストラクチャをテストするのに役立ちます。既存のインスタンスにパッチを適用して更新すると、問題の再現と特定が困難になる状態になることがあります。

COTS アプリケーションが自動インストールをサポートしていない場合は、[AWS パートナーネットワーク \(APN\)](#) に相談されることをお勧めします。詳細については、CAF ホワイトペーパーの「[プラットフォームの視点: アプリケーションとインフラストラクチャ](#) AWS」セクションを参照してください。

COTS と社内アプリケーションの再プラットフォームに関するよくある質問

このセクションでは、COTS と社内アプリケーションの再プラットフォームに関してよく寄せられる質問への回答を提供します。

リホストする代わりに再プラットフォームを選択する必要があるのはいつですか？

OS ベンダーがセキュリティ更新プログラムを切断した場合 (RHEL 5 や Windows 2008 など)、データベース以外のコンポーネントを再プラットフォーム化する必要があります。サポートされていない OS の場合、コンポーネントを乖離プライベートサブネットにリホストし、本番環境にデプロイする前にアップグレードできます。

データベースコンポーネントについては、Amazon RDS で商用データベースへの再プラットフォームが適切な選択であるかどうかを評価する必要があります。データベースのベンダーが AWS クラウドで現在のライセンスの使用を許可している場合は、Amazon EC2 で実行されているセルフマネージドデータベースを使用することもできます。

COTS アプリケーションをオープンソース・データベースにリファクタリングすることはできますか？

アプリケーションのベンダーがオープンソース・データベースをサポートしている場合は、COTS アプリケーションをオープンソース・データベースにリファクタリングできます。ベンダーのサポートを受けずに COTS アプリケーションにオープンソース・データベースを使用することはお勧めしません。

サーバーを AWS クラウドにすばやくリホストするには、どのような AWS ツールを使用できますか？

[MGN](#) をプライマリ移行サービスとして使用して、アプリケーションをリフトアンドシフトできます AWS。

アプリケーションのリプラットフォームにはどのような AWS ツールを使用できますか？

AWS クラウドには、以下を含むいくつかのツールがあります。

- [Systems Manager 自動アップグレード](#) – AWS クラウド上の Windows 2008 および SQL Server 2008 をアップグレードします。
- [AWS SCT](#) — データベーススキーマを商用データベースからオープンソース・データベースに変換します。
- [AWS DMS](#) – Amazon RDS データベースまたは Amazon EC2 で実行されているデータベースのいずれかで、オンプレミスデータベースから AWS クラウドにデータをレプリケートします。

リソース

リファレンス

- [AWS 移行ホワイトペーパーの AWS クラウド導入フレームワーク](#)
- [CloudWatch ダッシュボードを作成する](#)
- [Microsoft SQL Server DB インスタンスの制限](#)
- [自己マネージド Microsoft SQL Server データベースを Amazon RDS の完全マネージドデータベースに移行する](#)
- [AWS ワークショップ: Amazon FSx 上の Microsoft SQL Server フェイルオーバークラスターインスタンス](#)
- [Amazon EC2 Windows Server インスタンス OS をアップグレード](#)

ツール

- [AWS CodeBuild](#) および [AWS CodePipeline](#) は、AWS クラウドで継続的統合および継続的開発 (CI/CD) パイプラインを構築するのに役立ちます。
- [AWS DataSync](#) - DataSync はオンプレミスのファイルストアを AWS Cloud に複製するのに役立ちます。
- [AWS DMS](#) – AWS Database Migration Service (AWS DMS) は、オンプレミスデータベースから Amazon RDS データベースまたは Amazon EC2 上のデータベースにデータをレプリケートするのに役立ちます。
- [AWS Systems Managerベースの自動アップグレード](#)は AWS クラウド上の Windows 2008 および SQL Server 2008 をアップグレードするのに役立ちます。
- [AWS SCT](#) – AWS Schema Conversion Tool はデータベーススキーマを商用データベースからオープンソース・データベースに変換するのに役立ちます。
- [MGN](#) – AWS Transform MGN アプリケーションの移行を簡素化、迅速化、削減する高度に自動化されたlift-and-shift (リホスト) ソリューションです AWS。
- [Microsoft SQL Server データベースでの Windows から Linux への再プラットフォームのアシスタント](#)は、再プラットフォームツールです。

ドキュメント履歴

以下の表は、本ガイドの重要な変更点について説明したものです。今後の更新に関する通知を受け取る場合は、[RSS フィード](#) をサブスクライブできます。

変更	説明	日付
更新された AWS サービス名	アプリケーション移行サービスへの CloudEndure 移行を更新	2022 年 5 月 12 日
二	初版発行	2021 年 3 月 12 日

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。