



Microsoft ワークロードを AWSに移行するためのオプション、ツール、ベストプラクティス

AWS 規範ガイド



AWS 規範ガイド: Microsoft ワークロードを AWS に移行するためのオプション、ツール、ベストプラクティス

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
対象者	1
ターゲットを絞ったビジネス成果	2
Microsoft ワークロード AWS に を選択する理由	3
基本的なベストプラクティス	5
クラウドへの移行パス	7
移行戦略	7
主なトランスフォーメーション	7
移行戦略の選択	8
リホストが推奨される状況	9
リプラットフォーム/アーキテクチャの再設計が推奨される状況	9
リファクタリングが推奨される状況	9
Windows 移行プロセス	10
評価	10
準備	11
移行とモダナイズ	11
Windows 環境の検出	13
評価	13
エンタープライズアーキテクチャ	13
標準化と設定管理	13
適切なデータ	14
オートメーション	14
詳細な計画	14
準備	15
大規模移行の課題	15
レイテンシーの影響を受けやすい依存関係	15
IT 共有サービス	15
設定の更新	16
アプリケーションの機能テスト	16
アプリケーション依存関係検出用のツール	16
Microsoft ワークロードの移行	18
Active Directory の移行	18
評価	19
準備	20

移行	24
その他のリソース	26
Windows Server の移行	26
評価	26
準備	27
移行	28
ファイルサーバーの移行	29
評価	29
準備	31
移行	32
SQL Server の移行	32
評価	33
準備	34
移行	36
その他のリソース	40
.NET アプリケーションの移行	40
評価	40
準備	41
移行	42
リプラットフォーム	43
その他のリソース	46
Windows フェイルオーバークラスターの移行	46
評価	47
準備	49
移行	50
Microsoft のワークロードの監視	51
評価	51
準備	51
移行	52
移行ツール、プログラム、トレーニング	54
ツール	54
評価ツール	54
移行ツール	57
移行パートナーツール	59
管理ツール	60
プログラム	61

AWS Migration Acceleration プログラム	61
AWS Windows Migration Accelerator	62
AWS Windows 用移行アクセラレーションプログラム	62
AWS カウントダウン	62
トレーニング	63
自分のペースで進められる、インタラクティブな、クラスルーム形式のトレーニング	63
AWS パートナートレーニング	63
での Microsoft ライセンス AWS	64
評価	64
ライセンス込みオプション	65
BYOL オプション	67
準備	73
AWS License Manager	73
ライセンスに関する考慮事項	73
移行	74
AWS パートナー	75
AWS コンピテンシーパートナーを関与させるメリット	75
計画立案	75
コストの最適化	75
時間の節約	76
セキュリティの強化	77
次のステップ	78
リソース	79
Microsoft から AWS 移行へのガイドライン	79
一般的なガイドライン	79
動画	79
AWS ブログ投稿	79
寄稿者	80
オーサリング	80
レビューアー	80
テクニカルライター	80
ドキュメント履歴	81
用語集	83
#	83
A	84
B	86

C	88
D	91
E	95
F	98
G	99
H	100
I	102
L	104
M	105
O	109
P	112
Q	115
R	115
S	118
T	122
U	123
V	124
W	124
Z	125
.....	cxxvi

Microsoft ワークロードを に移行するためのオプション、ツール、ベストプラクティス AWS

Amazon Web Services ([寄稿者](#))

2026 年 3 月 ([ドキュメント履歴](#))

組織は、他のクラウドプロバイダーよりも 10 AWS 年以上にわたって、Microsoft ワークロードを に移行して実行しています。このガイドは、長年の移行とモダナイゼーションの取り組み AWS から得られた知識と専門知識に基づいて、Microsoft ワークロードの への移行を合理化するように設計されています AWS クラウド。このガイドを使用して、Windows のすべての移行フェーズを計画および実装できます。このガイドは、以下のようなさまざまな移行ユースケースに適用されます。

- 組織内のデジタルトランスフォーメーションとモダナイズジャーニーの一環として Windows の移行に着手しようとしている。
- Microsoft ワークロードを実行しているデータセンターのリース期限が近づいている。
- 可用性要件が異なるさまざまな Windows アプリケーションがあるが、地理的に離れた場所にワークロードをデプロイするリソースがない。

このガイドでは、移行ジャーニーの合理化に役立つさまざまな AWS ツールについて説明します AWS Transform AWS Application Migration Service。AWS ベストプラクティスに合わせて、このガイドは評価、動員、移行、モダナイズの [3 段階 AWS の移行プロセス](#)に従います。このプロセスは、Windows 移行の構造化と合理化に役立つ、実績ある移行フレームワークに基づいています。評価フェーズでは、クラウドでの運用の準備状況进行评估します。準備フェーズでは、移行計画の草案を作成し、評価フェーズで特定された準備状況のギャップを埋めます。次に、移行とモダナイズフェーズでワークロードの移行を開始します。ワークロードを体系的に移行し、ビジネス要件を満たすため、自動化ツールとテンプレートを組み合わせて使用します。

対象者

このガイドは、IT アーキテクト、移行リード、テクニカルリード、AWS パートナーチーム、および以下を担当するその他のロールを対象としています。

- Microsoft ワークロードをデータセンターから に移行する AWS クラウド
- での Windows 環境の管理 AWS クラウド

ターゲットを絞ったビジネス成果

このガイドは、お客様と組織が以下の目標を達成することに役立ちます。

1. Microsoft ワークロードの移行に使用できる戦略、プログラム、サービスについて説明します AWS。
2. Active Directory、Windows File Server、SQL Server、.NET ワークロードなど、特定の Microsoft ワークロードの AWS 移行パスを理解します。
3. セキュリティ、可用性、信頼性の要件を満たす AWS しながら、で Microsoft ワークロードを実行します。
4. Microsoft ワークロードを実行するためのライセンスのベストプラクティスを理解します AWS。

Microsoft ワークロード AWS にを選択する理由

AWS は、お客様の Microsoft ワークロードの移行とモダナイズを 14 年以上にわたって支援してきました。また、のサービス、プログラム、専門知識の最も幅広いポートフォリオを持ち、ビジネスを支える主要なアプリケーションの変革を加速しています。AWS を使用して移行およびモダナイズする場合、次の利点を期待できます。

- **イノベーションのロック解除** – 従来のモノリシックアーキテクチャからクラウドベースのマイクロサービスアーキテクチャに移行することで、迅速に適応して実験する自由が得られ、イノベーションをより迅速にロック解除できます。は、[Amazon Elastic Container Service \(Amazon ECS\)](#)、[Amazon Elastic Kubernetes Service \(Amazon EKS\)](#)、などの幅広いコンテナテクノロジー AWS を提供します [AWS Fargate](#)。また、が主導する包括的なサーバーレスポートフォリオも提供します [AWS Lambda](#)。深く統合された .NET サポート、オープンソースのデータベースオプション ([Amazon Aurora](#) など)、DevOps 自動化ツール、増え続ける生成 AI サービススイート AWS により、は、あらゆる規模でアプリケーションを構築およびモダナイズするために必要なツールを提供します。
- **コストの削減** – オープンソースのデータベースソリューションに移行することで、高価な Windows または SQL Server ライセンスにお金を払う必要がなくなります。例えば、Amazon Aurora は商用データベースと同じ機能を 10 分の 1 のコストで提供します。オープンソーステクノロジーに移行し、コンテナとサーバーレスソリューションを使用すると、総所有コスト (TCO) を削減し、コンピューティング消費量を最大化できます。
- **セキュリティの向上** – セキュリティ、コンプライアンス、ガバナンスサービスの幅広いポートフォリオ AWS を提供します。Active Directory の場合、は柔軟なオプション [AWS Directory Service](#) を提供します。を使用してクラウドに新しいフルマネージド Active Directory ドメインを作成するか [AWS Managed Microsoft AD](#)、[AWS Managed Microsoft AD \(ハイブリッドエディション\)](#) AWS を使用して既存のオンプレミス Active Directory をに拡張するか、[AD Connector](#) を使用して AWS アプリケーションをレプリケーションなしでオンプレミスディレクトリに直接プロキシします。[AWS アイデンティティサービス](#) を使用すると AWS アカウント、およびエンタープライズアプリケーション間でシングルサインオンが有効になり、ユーザーは同期や再入力なしで既存の認証情報を使用してリソースにアクセスできます。
- **信頼できるエキスパートとスキルを開発する** - AWS は、組織が移行目標をより早く達成できるよう支援した比類のない経験を持っています。[AWS Windows 用 Migration Acceleration Program \(MAP\)](#) は、AWS パートナーと AWS プロフェッショナルサービスからのサポートにより、クラウドへの移行の複雑さとコストを削減するためのベストプラクティス、ツール、財務上のインセンティブを提供します。AWS は、[戦略的クラウドプラットフォームサービスの 2025 年 " Magic Quadrant](#) で、実行能力の最高位のリーダーとして認識されています。

- 処理能力の価格とパフォーマンスを向上させる – は処理イノベーションを AWS リードし、AWS Graviton4-based インスタンスを提供します。これらのインスタンスは、[前世代と比較してパフォーマンスが最大 30% 向上し、データベースワークロードのパフォーマンスが最大 40% 向上](#)します。Amazon Aurora は、商用データベースと比較して、標準の MySQL のスループットの 5 倍、標準の PostgreSQL のスループットの 3 倍を 10 分の 1 のコストで提供します。
- 柔軟なライセンスオプションを活用する – AWS は、新規および既存の Microsoft ソフトウェアライセンスを使用するためのオプションをクラウドで最も多く提供します。ライセンス込みの [Amazon Elastic Compute Cloud \(Amazon EC2\)](#) または [Amazon Relational Database Service \(Amazon RDS\)](#) インスタンスを購入する場合、SQL Server ライセンスが含まれます。また、[Amazon EC2 Dedicated Hosts](#) または [ソフトウェアアシュアランスによる Microsoft License Mobility](#) AWS を使用して、既存のライセンスを に持ち込むこともできます。[AWS License Manager](#) を使用すると、AWS およびオンプレミス環境全体のライセンス使用状況を簡単に追跡できるため、コンプライアンス違反のリスクが軽減されます。

詳細については、AWS ドキュメントの「[Windows on AWS](#)」を参照してください。

基本的なベストプラクティス

AWS 移行のためのスケーラブルで安全な基盤を確立することで、Windows 環境を簡単に管理し、効率的に実行できます AWS。Microsoft ワークロードを に移行する前に AWS、以下の基本的なベストプラクティスを検討することをお勧めします。

- Microsoft ライセンスへの支出を最適化する – ライセンスはクラウド移行における重要な要素であり、以降のすべての決定に影響を与えます。ライセンスオプションをできるだけ早い段階で理解しておくことをお勧めします。ライセンスの詳細については、このガイドの「[Microsoft ワークロードのライセンス](#)」セクションを参照してください。
- クラウドアーキテクチャを合理化する – [AWS Well-Architected フレームワーク](#) は、ワークロードをクラウドで確実に実行できるようにします。このフレームワークに準拠して重大な問題を回避し、組織のニーズに合わせてスケーリングするためのガイダンスと戦略について説明します。このガイダンスでは、請求、アクセスコントロール、セキュリティコントロールについても説明します。
- 統合された管理しやすいクラウドネットワークを構築する – [AWS Transit Gateway](#) は、ネットワークをより簡単に管理し、Classless Inter-Domain Routing (CIDR) 範囲計画などの重複するネットワークがオンプレミスや他のクラウド環境で作成されないようにするのに役立ちます。これにより、トラフィックを必要に応じて各ネットワークにルーティングできます。アカウントが相互に、およびオンプレミス環境とインターネットにどのようにルーティングするかを決定する必要があります。これにより、ネットワークトラフィックを保護するために適切なコントロールを設定できます。たとえば、既存のオンプレミスデータセンター AWS アカウント を拡張し、ファイアウォール、侵入検知システム (IDS)、侵入防止システム (IPS) などの境界防御を使用するか、リソースを保護するために AWS これらの境界防御を含む AWS ネットワークアカウントを設定する必要があります。
- クラウドセキュリティの優先順位を付ける – 最小特権のアクセス許可を適用するセキュリティのベストプラクティスに従いながら、単一アカウントからマルチアカウント環境に移行することをお勧めします。また、[AWS 責任共有モデル](#) を十分に理解し、組織の俊敏性を維持しながら環境を保護する方法を計画することをお勧めします。セキュリティを強化および維持するために、Amazon API Gateway、AWS WAF Application Load Balancer、Amazon CloudWatch、AWS CloudTrail Amazon GuardDuty、およびその他の のサービスを使用できます。マルチアカウント戦略の詳細については、「[規範ガイド](#)」ドキュメントの「[複数の への移行 AWS アカウント](#)」を参照してください。 AWS
- クラウドで共有されている IT サービスを管理する – クラウドでワークロードを効率的に管理するには、ワークロードで使用されるすべての共有サービスを特定し、クラウドでどのように提供さ

れるかを計画することが重要です。これには例えば、Active Directory、ファイルサーバー、SQL データベース、DNS、仮想プライベートネットワーク (VPN)、Simple Mail Transfer Protocol (SMTP)、バックアップ、モニタリングサービスなどがあります。インベントリを作成したら、既存のサービスをクラウドに拡張するか、まったく新しいサービスインスタンスを設定するか、代替のマネージドクラウドサービスを使用するかを決定できます。このガイドの後続のセクションで、これらの考慮事項について詳しく説明します。

クラウドへの移行パス

このセクションでは、Windows アプリケーションを AWS に移行するためのベストプラクティスを実装するための大まかなアプローチについて説明します。これらの移行戦略とステップの詳細については、このガイドの後続のセクションで説明します。

移行戦略

移行戦略とは、ワークロードを AWS クラウドに移行するために使用するアプローチです。アプリケーションをクラウドに移行するには、次の 7 つの移行戦略があります。これらの戦略は 7 Rs と呼ばれ、ガートナーが 2019 年に特定した [7 Rs](#) に基づいて構築されています。

- リホスト (リフトアンドシフト) — クラウド機能を活用するための変更を加えずに、アプリケーションをクラウドに移行します。
- 再配置 (ハイパーバイザーレベルのリフトアンドシフト) — 新しいハードウェアを購入したり、アプリケーションを書き換えたり、既存の運用を変更したりすることなく、インフラストラクチャをクラウドに移行できます。
- リプラットフォーム (リフトアンドリシェイプ) — アプリケーションをクラウドに移行し、クラウド機能を活用するためある程度の最適化を導入します。
- 再購入 (ドロップアンドショップ) — 通常、従来のライセンスから Software as a Service (SaaS) モデルに移行して、別の製品に切り替えます。
- リファクタリング/アーキテクチャの再設計 — クラウドネイティブ特徴を最大限に活用して、俊敏性、パフォーマンス、スケーラビリティを向上させ、アプリケーションを移動させ、アーキテクチャを変更します。
- 保持 (再アクセス) — アプリケーションをお客様のソース環境で保持します。これには、主要なリファクタリングを必要とするアプリケーションや、お客様がその作業を後日まで延期したいアプリケーション、およびそれら移行するためのビジネス上の正当性がないため、お客様が保持するレガシーアプリケーションなどがあります。
- 廃止 — お客様のソース環境で不要になったアプリケーションを廃止または削除します。

主なトランスフォーメーション

レガシー Windows アプリケーションとデータベースをモダナイズすると、次の主なトランスフォーメーションが行われます。

- リホスト – 最初のステップは、オンプレミスのインフラストラクチャをクラウドインフラストラクチャに移行することです。この戦略は、多くの場合、「リフトアンドシフト」またはリホストと呼ばれます。リホストとは、既存のアプリケーションとデータベースをクラウドサーバーインスタンスに移行することを意味します。コードを変更する必要はなく、ユーザーはインスタンス設定、ソフトウェアイメージ、その他のリソースを管理する責任を負います。
- リプラットフォーム – クラウド環境への移行後に行う次のトランスフォーメーションは、アプリケーションとデータベースをより自動化されたマネージド環境にリプラットフォームすることです。アプリケーションの観点からは、仮想マシン (VMs) からコンテナまたはマネージドアプリケーションプラットフォームに移行することを意味します。アプリケーションのコンテナ化は、アプリケーションを迅速に開発、保守、デプロイし、移植性を向上させるのに役立ちます。または、は、容量のプロビジョニング、負荷分散、スケーリングを自動的に処理するマネージドプラットフォーム [AWS Elastic Beanstalk](#) を提供します。これにより、インフラストラクチャ管理を最小限に抑え、完全にコンテナ化することなく、アプリケーションをリプラットフォームできます。データベース側から見ると、セルフサービスモデルから Amazon RDS for SQL Server などのマネージドデータベースサービスに移行すると、プロビジョニング、パッチ適用、バックアップが不要になります。これにより、組織により多くの価値を追加できるアクティビティのリソースが解放されます。
- リファクタリング/アーキテクチャの再設計 – 3 番目のトランスフォーメーション領域は、商用ソフトウェアライセンスからオープンソースのオプションへの移行です。従来の商用ソフトウェアベンダーの多くは、顧客の囲い込みを目的としたソフトウェアライセンス契約に基づき、アップグレードと移行を強制するライセンスの罰則規定を利用してビジネスを構築しています。多くの場合、商用ソフトウェアライセンス料金は、同等のオープンソースのオプションに対してコストが一般的に 20~50% 増加します。コストを削減し、パフォーマンスを向上させ、最新のイノベーションにアクセスできるように、アプリケーションとデータベースをリファクタリングしてオープンソースのオプションを活用することをお勧めします。

これらの主なトランスフォーメーション領域は、アプリケーションおよびモダナイズの全体的な準備状況に応じて、段階的に、または一度にすべて完了できます。

移行戦略の選択

どの移行戦略を選ぶかは、組織のビジネス目標と IT 目標によって異なります。最も一般的なビジネス推進要因には、コストの削減、リスクの軽減、効率の向上、スキルギャップへの対応、イノベーションの加速などがあります。次のガイドンスを使用して、どの推進要因が組織にとって重要かを評価し、組織の推進要因に基づいて移行戦略を選択することをお勧めします。なお、3 つのアプローチ

はどれも、ジャーニーの各フェーズにおける優先順位に応じて、クラウドモダナイゼーションジャーニーで選択可能なことに注意してください。

リホストが推奨される状況

リホスト (リフトアンドシフト) は通常、アプリケーションでコードやアーキテクチャを変更する必要がないため、よりすばやく簡単に実行できます。また、リホストは、ビジネスのリスクと中断を最小限に抑えます。アプリケーションは変更されないため、運用チームは普段どおり事業運営をけることができます。これは、大規模な移行で、関与するワークロードの数が多いためわずかな変更でも重大な影響を及ぼす場合に特に当てはまります。ただし、リホストではクラウドのメリットを最大限に活用できないことを考慮することが重要です。例えば、既存のプラットフォームの問題があるアプリケーションを移行した場合、その問題は移行後も残ります。最後に、リホストの総保有コスト (TCO) と投資収益率 (ROI) は、他の移行アプローチと比較して低くなることを考慮してください。

リプラットフォーム/アーキテクチャの再設計が推奨される状況

一般的に、リプラットフォームはリホストよりもコスト効率が高くなります。リプラットフォームを使用すると、自動化を強化でき、アプリケーションは自動スケーリング、モニタリング、バックアップの実行などのクラウド機能をさらに活用できるようになります。リプラットフォームにより、クラウド運用チームの運用上のオーバーヘッドが軽減され、既存のプラットフォームの問題によるリスクが最小限に抑えられます。ただし、リプラットフォームにはリホストによる移行と比べて時間がかかります。また、リプラットフォームには、アプリケーションでコード変更を実行するための自動化を設定し、新しいプラットフォームを運用可能にするための追加スキルが必要です。

リファクタリングが推奨される状況

リファクタリングは通常、最も費用対効果の高い移行アプローチです。リファクタリングは、アプリケーションの耐障害性を向上させるためにアプリケーションコンポーネントをデカップリングすることで、アプリケーションが新しい要件に迅速に適應できるようにするクラウドネイティブなアプローチです。ただし、リファクタリングでは、より高度なコーディングと自動化のスキルが求められます。また、アプリケーションの再構築を伴うため、リファクタリングの実装には時間がかかります。

Windows 移行プロセス

既存の Windows 環境を に移行するには、慎重な計画と実装 AWS が必要です。このプロセスには、リソースの現在の使用状況の特定、への移行に伴うコスト削減の可能性の評価 AWS、セキュリティニーズの決定、組織のすべての要件を満たす明確に定義されたクラウドアーキテクチャの構築が含まれます。AWS を使用して、現在の Windows サーバーインフラストラクチャを迅速かつ簡単に移行し、システム効率を最大化しながら運用コストを削減できます。また、には、プロセス全体の制御を維持し、クラウド内の Windows 環境が最大限のパフォーマンスのために最適に設定されていることを確認するのに役立つさまざまな強力なツールとサービス AWS も用意されています。

このセクションでは、AWS クラウドへの複数のアプリケーションの移行を成功させるうえで組織を支援するために AWS が開発した 3 段階の移行プロセス (評価、準備、移行とモダナイズ) の概要を説明します。

評価

評価フェーズは、組織のクラウドへの移行準備状況を把握するのに役立ちます。ツールを使用して、オンプレミスのコンピューティングリソースを評価し、でアプリケーションを実行するためのコスト予測を構築することで、評価フェーズ AWS を支援できます AWS。次のツールを検討することをお勧めします。

- [移行準備評価](#) を使用すると、クラウドジャーニーのどの段階にいるのかを把握できます。
- [AWS 最適化とライセンス評価 \(AWS OLA\)](#) を使用して、実際のリソース使用率、サードパーティのライセンス、アプリケーションの依存関係に基づいて、現在のオンプレミス環境とクラウド環境を評価および最適化します。
- [Migration Evaluator](#) を使用すると、AWS への移行に関するデータ主導型のビジネスケースを構築できます。
- [Cloud Economics Center](#) を使用すると、信頼性の向上、コストの最適化、スケーラビリティなどの目標を定義して、移行のビジネスケースを作成できます。
- [AWS Transform](#) を使用すると、サーバーとアプリケーションのインベントリデータを収集できるため、移行の評価、計画、追跡に役立ちます。
- [移行検証ツールキット PowerShell モジュール](#) を使用すると、Microsoft ワークロードを検出し、AWS に移行できます。

準備

準備フェーズでは、移行計画を策定してビジネスプランを練り直し、評価フェーズで明らかになった準備状況におけるギャップに対処します。ベースライン環境の構築、運用準備の促進、クラウドスキルの開発に集中することが重要です。大規模なアプリケーションポートフォリオの移行は複雑な作業になる場合があります。このプロセスを容易にするために、は、一連のパイロットワークロードをクラウドに迅速、安全、コスト効率良く移行するためのさまざまなツールとサービス AWS を提供します。リホスト、再配置、リプラットフォーム、再購入、リファクタリング/再構築、保持、廃止という 7 つの一般的な移行戦略の 1 つ以上を使用してアプリケーションポートフォリオのデータを収集し、アプリケーションを合理化することで、意思決定の基盤を改善できます。は、Windows ベースのアプリケーションとワークロードをクラウドに移行するために使用できる一連の サービス AWS を提供します。

- [AWS Transform](#)
- [AWS Application Migration Service](#)
- [AWS Database Migration Service](#)
- [AWS 移行コンピテンシーパートナー](#)
- [の管理とガバナンス AWS](#)
- [AWS Control Tower](#)

移行とモダナイズ

移行とモダナイズの段階では、移行の対象となる各アプリケーションを慎重に設計、移行、検証する必要があります。Application Migration Service により、物理、仮想、またはクラウドインフラストラクチャから多数のサーバーを AWS に簡単に移行できます。Application Migration Service を使用すると、さまざまなアプリケーションに同じ自動化プロセスを適用し、既存の環境からクラウドにすばやくリフトアンドシフトできます。

[AWSでの Cloud Migration Factory](#) ソリューションは、多数のサーバーが関与する大規模な移行の手動プロセスを調整および自動化するように設計されています。このソリューションは、ワークロードを AWS 大規模に移行するためのオーケストレーションプラットフォームを提供することで、パフォーマンスを向上させ、長いカットオーバーウィンドウを防ぐのに役立ちます。[AWS プロフェッショナルサービス](#)、[AWS パートナー](#)、およびその他の企業はすでにこのソリューションを使用して、お客様が数千台のサーバーを AWS クラウドに移行できるよう支援しています。

お客様のチームは、Microsoft ワークロードをオンプレミスで構築して実行する専門家です。そのエクスペリエンスはクラウドでも強化できます。に移行すると、信頼される Windows の世界に、より効率的で信頼性の高いエクスペリエンスを提供 AWS できます。AWS では、既存の Microsoft ワークロードを簡単かつ迅速に移行できるように設計された幅広いクラウドサービスにアクセスできます。容量のスケラビリティの向上、ストレージオプションの強化、セキュリティの強化といったメリットがあります。

Windows 環境の検出

Windows Server AWS Application Migration Service、Linux、その他の x86 ベースのオペレーティングシステムとそのワークロードを に移行するなど、今日の利用可能なテクノロジー AWS を使用すると、かなり簡単です。ただし、これらのワークロードを適切に動作させること、それを大規模に実行することには、また別の課題があります。このセクションでは、Microsoft ワークロードの迅速かつ安全でスムーズな移行を可能にする、移行上の考慮事項について説明します。

評価

最小限の計画と自動化でも、小規模な (100 台のサーバーが関与するような) 移行を「力づくで」実行することはできますが、この方法論を使用して 500 台以上のサーバーを移行することはできません。以下の考慮事項は、大規模移行の成功に寄与する主な要因であり、[移行準備状況評価 \(MRA\)](#) を使用して、焦点を当てたい考慮事項を特定できます。

エンタープライズアーキテクチャ

環境内の技術的負債が多いほど、移行が難しくなります。健全なエンタープライズアーキテクチャプログラムを持つ組織は、その環境に使用するソフトウェアとシステムを最新バージョン (多くの場合、メジャーリリースの N および N-1 バージョン) に制限するよう努めています。これにより、考慮する必要があるシナリオの数を減らすだけでなく、新しいリリースにおける進歩や改善も活用できます。例えば、Windows Server 2012、Windows Server 2008 など、Windows Server のバージョンが古くなるにつれ、最新のバージョンと比べて Windows Server 環境での自動化が困難になります。古いバージョンやサポートされていないバージョンでは、ライセンス管理もさらに困難になります。

標準化と設定管理

環境の標準化についても考慮する必要があります。組織によって手動で構築および維持されている環境は、「ペット」に近いものと見なされます。1 つとして同じシステムはなく、標準化されたイメージ、Infrastructure as Code (IaC)、または継続的インテグレーションと継続的デリバリー (CI/CD) パイプラインを使用して構築された場合よりもはるかに多くの設定の組み合わせがあります。

例えば、移行時には個々のサーバーを手動で移行するのではなく、IaC または CI/CD を使用して一般的なウェブサーバーを再構築するのがベストプラクティスです。また、データベース、ファイル共有、リポジトリなどのすべての永続データをデータストアに保存することもベストプラクティスです。IaC または CI/CD を使用してシステムを再構築しない場合は、最低限、設定管理ツール (Puppet、Chef、Ansible など) を使用してサーバーを標準化する必要があります。

適切なデータ

適切なデータもまた、移行を成功させるための重要な要素です。現在のサーバーとそのメタデータに関する正確なデータは、自動化と計画に不可欠です。適切なデータがないと、移行を計画する際の難易度が増えます。適切なデータの例としては、サーバーの正確なインベントリ、サーバー上のアプリケーション、サーバー上のソフトウェアとバージョン、CPU の数、メモリ容量、ディスクの数などがあります。ウェーブプランナーが計画を立てるのに必要なデータや、移行プロセスの自動化の一環として使用する予定のデータをキャプチャすることをお勧めします。

オートメーション

大規模な移行には自動化が不可欠です。自動化の例としては、エージェントのインストール、.NET や PowerShell などの自動化に必要なユーティリティのソフトウェアバージョンの更新、AWS Systems Manager エージェント (SSM エージェント)、Amazon CloudWatch エージェント AWS などのソフトウェアのロードや更新、またはの実行に必要なその他のバックアップや管理ソフトウェアなどがあります AWS。

詳細な計画

詳細な計画の策定と管理も、大規模な移行に欠かせません。数週間かけて毎週 50 台のサーバーを移行するには、明確に定義された計画を立てる必要があります。効果的な計画には以下が含まれます。

- ウェーブプランニングを使用して、依存関係と優先順位に従ってサーバーをウェーブに整理します。
- 週次計画 (カットオーバーまで) を使用してアプリケーションチームとやり取りし、カットオーバー中に対処する必要があるネットワーク、DNS、ファイアウォール、その他の詳細を特定します。
- 詳細な 1 時間ごとの計画 (実際のカットオーバー付近) を使用して、カットオーバーメンテナンスウィンドウを説明します。
- 可否判定基準を使用して、どういった状況でアプリケーションが AWS にカットオーバーしたと見なされるか、またはソースリソースへのフェイルバックが必要かを記述します。
- クリーンアップアクティビティは、完了する必要があるフォローアップアクティビティとして使用します。これらのアクティビティは、カットオーバーメンテナンスウィンドウの期間外、または [ハイパーケア](#) の完了後に実行することができます。クリーンアップアクティビティには、バックアップやさまざまなエージェントの検証、サーバーからの Application Migration Service エージェントの削除、ソースサーバーと関連するリソースの削除が含まれます。

準備

準備フェーズでは、組織の複雑な問題やバリエーションをできるだけ多く検出し、移行計画中に考慮できるようにすることが重要です。理想的には、カットオーバーメンテナンスウィンドウ中にこのような複雑な問題やバリエーションに対処することは避け、フェイルバックを防ぐとよいでしょう。

大規模移行の課題

移行の失敗は、アプリケーションが新しい環境にカットオーバーされているが、移行メンテナンスウィンドウ内でパフォーマンス要件または機能要件を満たすことができない場合に発生します。この場合、アプリケーションは元の場所にフェイルバックされます。さらに、そのアプリケーションに依存する他のすべてのアプリケーションもフェイルバックする必要があります。アプリケーションを再スケジュールする必要があるため、移行の失敗は現在のウェブだけでなく、将来のウェブにも影響を与える傾向があります。

レイテンシーの影響を受けやすい依存関係

移行が失敗する主な理由は、レイテンシーの影響を受けやすい依存関係です。レイテンシーの影響を受けやすい依存関係を特定しないと、パフォーマンスの問題が発生し、許容できない応答時間やトランザクション時間が生じる可能性があります。

例えば、通常、アプリケーションではそのデータベースサーバーとアプリケーションサーバーを同時にクラウドに移動します。これは、データベースサーバーとアプリケーションサーバーが相互に頻繁に通信し、両方が同じデータセンターにある場合と同じミリ秒未満の応答時間が必要であるためです。データベースのみをクラウドに移動すると、それらのトランザクションに数秒のレイテンシーが発生し、アプリケーションのパフォーマンスに深刻な影響を与える可能性があります。これは、相互に大きく依存し、適切に動作するために同じデータセンターに存在する必要があるアプリケーションにも適用されます。

したがって、アプリケーションの依存関係を理解して対処することが、移行計画を立てる上で最も重要になります。相互に依存しているアプリケーションとサービスを特定し、一緒に移行できるようにする必要があります。

IT 共有サービス

ワークロードがクラウドに移ったら、適切かつ安全な機能と保守のために、さまざまなサービスが必要になります。これには、ランディングゾーン、ネットワークとセキュリティの境界、認証、パッチ適用、セキュリティスキャナー、IT サービス管理ツール、バックアップ、踏み台ホスト、その他の

リソースが含まれます。これらのサービスがないと、ワークロードが正しく動作せず、元の場所にフェイルバックされる可能性があります。

設定の更新

ほとんどの場合、ワークロードをクラウドに移動した後、ワークロードが適切に機能するようにいくつかの設定変更を行う必要があります。これらの設定変更は、多くの場合、ワークロードの以下の依存関係に関連付けられます。

- ファイアウォールルール
- 許可リスト
- DNS レコード
- 接続文字列

適切な設定更新を行わないと、ワークロード、そのユーザー、およびその依存システムが相互に通信できなくなる可能性があります。機能停止期間内にこれらの問題を解決することは可能かもしれませんが、このタイミングでの変更は時間がかかったり、時間内に間に合わせることができない変更レコードが必要になったりすることがあります。

アプリケーションの機能テスト

大規模移行のもう 1 つの課題は、アプリケーションの機能テストの必要性です。多くの組織は、レイテンシーの影響を受けやすい依存関係、IT 共有サービス、または必要な設定更新を特定するためにアプリケーションチームに依存しているため、これは特に重要です。アプリケーションが許容可能なパフォーマンスで完全に機能していることを検証するため、アプリケーションチームが、カットオーバーメンテナンスウィンドウ中に実行できるテストプランを文書化または自動化するのが理想的です。カットオーバーメンテナンスウィンドウを最小限に抑えるには、30 分以内にテストを完了できる必要があります。

アプリケーション依存関係検出用のツール

移行を成功させるには、レイテンシーの影響を受けやすい依存関係と接続設定項目の両方を検出するため、アプリケーション間の依存関係を決定することが重要です。マーケットプレイスには、検出ツール (エージェントベースの [AWS Transform ツール](#)) や [Cloudamize](#) (エージェントベースのツール) など、依存関係を検出するためのツールがいくつか用意されています。

アプリケーション依存関係検出用のツールを選択するときは、次の点を考慮してください。

- 期間 – 既知のピーク、月末、その他のイベントなどのアプリケーション固有のイベントをキャプチャするのに十分な期間にわたり検出ツールを実行することをお勧めします。推奨される最小日数は 30 日です。
- アクティブ (エージェントベース) – アクティブな依存関係検出ツールは、多くの場合、オペレーティングシステムのカーネルに埋め込まれ、すべてのトランザクションをキャプチャします。ただし、これは通常、最もコストと時間がかかる方法です。
- パッシブ (エージェントレス) – パッシブ依存関係検出ツールは、実装がはるかに安価で高速ですが、使用頻度の低い接続を見落とすリスクがあります。
- 組織の知識 – アプリケーション検出ツールは詳細で正確な情報を提供しますが、ほとんどの組織は、アプリケーションの依存関係を検出するためにアプリケーションチームとその組織が培ってきた知識に依存しています。アプリケーションチームは多くの場合、レイテンシーの影響を受けやすい依存関係についての知識は豊富ですが、接続設定、ファイアウォールルール、パートナーからの許可リスト要件などの詳細を見逃すことは珍しくありません。組織の知識を使用してアプリケーションの依存関係の検出を強化できますが、関連するリスクを考慮して軽減することもお勧めします。例えば、アプリケーションチームの知識のみに依存すると、接続設定項目やレイテンシーの影響を受けやすい依存関係を見落とすリスクがあります。これにより、機能停止や移行の失敗に至る可能性があります。このリスクを軽減するために、詳細なアプリケーション機能テストを実施することをお勧めします。

Microsoft ワークロードの移行

このセクションでは、特定の Microsoft ワークロードに関する規範的なガイドランスについて説明します。以下のワークロード固有のアプローチはすべて、評価、準備、移行とモダナイズのフレームワークに準拠しています。

トピック

- [Active Directory の移行](#)
- [Windows Server の移行](#)
- [ファイルサーバーの移行](#)
- [SQL Server の移行](#)
- [.NET アプリケーションの移行](#)
- [Windows フェイルオーバークラスターの移行](#)
- [Microsoft のワークロードの監視](#)

Active Directory の移行

Active Directory は、多くの企業環境における一般的な ID およびアクセス管理ソリューションです。DNS、ユーザー、マシン管理を組み合わせた Active Directory は、Microsoft と Linux の両方のワークロードでユーザー認証を一元的に行う理想的な選択肢となっています。クラウドまたはへの移行を計画する場合 AWS、Active Directory をに拡張 AWS するか、マネージドサービスを使用してディレクトリサービスインフラストラクチャの管理をオフロードするかの選択に直面することになります。組織にとって適切なアプローチを決定する際には、各オプションのリスクと利点を理解しておくことをお勧めします。

Active Directory 移行の適切な戦略とは、組織のニーズに合い、AWS クラウドを活用できるようにすることです。これには、ディレクトリサービス自体だけでなく、他のサービスとやり取りする方法も考慮する必要があります AWS のサービス。さらに、Active Directory を管理するチームの長期的な目標も考慮する必要があります。

Active Directory の移行に加えて、Active Directory の場所のアカウント構造、のネットワークポート AWS アカウント、Active Directory AWS のサービスを必要とする DNS 統合やその他の使用の可能性を決定する必要があります。アカウントトポロジの設計やその他の移行戦略上の考慮事項については、本ガイドの「[基本的なベストプラクティス](#)」セクションを参照してください。

評価

移行を成功させるには、既存のインフラストラクチャを評価し、環境に必要な主要な機能を理解することが重要です。移行方法を選択する前に、次の項目をお読みになることをお勧めします。

- 既存の AWS インフラストラクチャ設計を確認する – このガイドの [Windows 環境検出](#) セクションのガイダンスに従い、評価方法を使用して、フットプリントとインフラストラクチャ要件をまだ把握していない場合は、既存の Active Directory インフラストラクチャを確認します。Microsoft for Active Directory インフラストラクチャの所定のサイズを使用することをお勧めします AWS。Active Directory インフラストラクチャを に拡張する場合は AWS、Active Directory 認証フットプリントの一部のみが必要になる場合があります AWS。このため、Active Directory のフットプリントを完全に移動しない限り、環境をオーバーサイズにすることは避けてください AWS。詳細については、Microsoft ドキュメントの「[Active Directory ドメイン サービスのキャパシティ プランニング](#)」を参照してください。
- 既存のオンプレミスの Active Directory 設計の確認 — オンプレミス (自己管理型) Active Directory の現在の使用状況を確認します。Active Directory 環境を に拡張する場合は AWS、オンプレミス環境の拡張機能 AWS としても、 で複数のドメインコントローラーで Active Directory を実行することをお勧めします。これは、複数のアベイラビリティゾーンにインスタンスをデプロイすることで潜在的な障害に備えて設計する [AWS Well-Architected フレームワーク](#) に準拠しています。
- アプリケーションとネットワークの依存関係の特定 — 最適な移行戦略を選択する前に、組織が機能するために必要な Active Directory の機能をすべて理解しておく必要があります。マネージドサービスとセルフホスティングのどちらかを選択する際には、それぞれのオプションを理解することが重要であることを意味します。どの移行が適切かを判断するときは、以下の項目について検討します。
- アクセス要件 — Active Directory を制御するためのアクセス要件により、適切な移行方法が決まります。コンプライアンス規制のために任意のタイプのエージェントをインストールするために Active Directory ドメインコントローラーへのフルアクセスが必要な場合は、 が適切なソリューションではない AWS Managed Microsoft AD 可能性があります。代わりに、ドメインコントローラーから AWS アカウント内の Amazon Elastic Compute Cloud (Amazon EC2) へ Active Directory を拡張することを検討してください。
- 移行のタイムライン — 移行のスケジュールが延長され、完了期限が明確でない場合は、クラウド環境とオンプレミス環境のインスタンスを管理するための不測の事態が発生しているかどうかを確認してください。認証は、管理上の問題を回避するために Microsoft のワークロードに導入すべき重要なコンポーネントです。Active Directory の移行は、移行の早い段階で計画することをお勧めします。

- バックアップ戦略 — Active Directory ドメインコントローラーのシステム状態をキャプチャするために既存の Windows バックアップを使用する場合、AWS の既存のバックアップ戦略を引き続き使用できます。さらに、はインスタンスのバックアップに役立つテクノロジーオプション AWS を提供します。例えば、[Amazon Data Lifecycle Manager](#)、[AWS Backup](#)、[AWS Elastic Disaster Recovery](#) は、Active Directory ドメインコントローラーをバックアップするためのサポートされているテクノロジーです。問題を回避するには、Active Directory の復元に頼らないことが最善です。推奨されるベストプラクティスは耐障害性のあるアーキテクチャを構築することですが、復旧が必要な場合はバックアップ方法を用意しておくことが重要です。
- デザスタリカバリ (DR) のニーズ – Active Directory を に移行する場合は AWS 、災害発生時の耐障害性を設計する必要があります。既存のアクティブディレクトリを に移動する場合は AWS、セカンダリ を使用し AWS リージョン、 を使用して 2 つのリージョンを接続し AWS Transit Gateway でレプリケーションを実行できます。通常はこの方法が推奨されます。信頼性をテストするためにプライマリサイトとセカンダリサイト間の接続を何日も切断するような隔離された環境でのフェイルオーバーテストについて、さまざまな要件を課している組織もあります。これが組織の要件である場合は、Active Directory からスプリットブレインの問題をクリーンアップするのに時間がかかる可能性があります。[AWS Elastic Disaster Recovery](#) をアクティブ/パッシブの実装として使用できる場合があります。この場合、DR サイトをフェイルオーバー環境として残し、DR 戦略を個別に定期的にテストする必要があります。組織の目標復旧時間 (RTO) と目標復旧時点 (RPO) の要件を計画することは、移行を評価する際の重要な要素です AWS。実装を検証するためのテストとフェイルオーバーの計画とともに、要件が定義されていることを確認してください。

準備

Active Directory を AWS に移行または拡張する際には、組織上および運用上のニーズを満たす適切な戦略が重要な要素となります。の採用には、との統合方法の選択 AWS のサービスが不可欠です AWS。AWS Managed Microsoft AD ビジネス要件を満たす Active Directory または のメソッド拡張を必ず選択してください。Amazon Relational Database Service (Amazon RDS) などのサービスには、の使用に依存するいくつかの機能があります AWS Managed Microsoft AD。Amazon EC2 およびの Active Directory に互換性の制約があるかどうかを判断するには、必ず制限を評価し AWS のサービス てください AWS Managed Microsoft AD。計画プロセスの一環として、次の統合ポイントを検討することをお勧めします。

で Active Directory を使用する次の理由を考慮してください AWS。

- AWS アプリケーションが Active Directory と連携できるようにする
- Active Directory を使用して にログインする AWS マネジメントコンソール

AWS アプリケーションが Active Directory と連携できるようにする

AWS Managed Microsoft AD ディレクトリを使用するには、[AWS Client VPN](#)、[AWS マネジメントコンソール](#)[AWS IAM アイデンティティセンター](#)、[Amazon Connect](#)、[Amazon FSx for Windows File Server](#)、[Amazon Quick](#)、[Amazon RDS for SQL Server](#) (Directory Service にのみ適用)、[Amazon WorkMail](#)、[Amazon WorkSpaces](#) などの複数の AWS アプリケーションとサービスを有効にすることができます。ディレクトリで AWS アプリケーションまたはサービスを有効にすると、ユーザーは Active Directory 認証情報を使用してアプリケーションまたはサービスにアクセスできます。使い慣れた Active Directory の管理ツールを使用して、インスタンスを [AWS Managed Microsoft AD ディレクトリ](#) に結合することで、Active Directory のグループポリシーオブジェクト (GPO) を適用して Windows または Linux 用 Amazon EC2 インスタンスを一元管理できます。

ユーザーは Active Directory の認証情報でインスタンスにサインインできます。これにより、個々のインスタンスの認証情報を使用したり、プライベートキー (PEM) ファイルを配布する必要がなくなります。こうすることで、使い慣れた Active Directory のユーザー管理ツールを使用して、ユーザーに対してアクセスをすばやく許可または取り消すことができます。

Active Directory を使用して にログインする AWS マネジメントコンソール

AWS Managed Microsoft AD では、ディレクトリのメンバーに へのアクセスを許可できます AWS マネジメントコンソール。デフォルトでは、ディレクトリのメンバーが AWS リソースにアクセスすることはできません。ディレクトリメンバーに AWS Identity and Access Management (IAM) ロールを割り当てて、さまざまな AWS のサービス および リソースへのアクセスを許可します。IAM ロールでは、ディレクトリメンバーに付与するをサービス、リソース、およびアクセス権限のレベルを定義します。

たとえば、ユーザーが [Active Directory 認証情報](#) AWS マネジメントコンソール を使用して にサインインできるようにします。そのためには、ディレクトリ内のアプリケーションとして AWS マネジメントコンソール を有効にしてから、Active Directory ユーザーとグループを IAM ロールに割り当てます。ユーザーが にサインインすると AWS マネジメントコンソール、AWS リソースを管理する IAM ロールを引き受けます。これにより、別の SAML インフラストラクチャを設定および管理 AWS マネジメントコンソール しなくても、ユーザーに へのアクセスを簡単に付与できます。詳細については、AWS セキュリティブログの [AWS IAM アイデンティティセンター 「Active Directory 同期が AWS アプリケーションエクスペリエンスを強化する方法」](#) を参照してください。ディレクトリまたはオンプレミスの Active Directory 内のユーザーアカウントへのアクセスを許可できます。これにより、ユーザーは既存の認証情報とアクセス許可を使用して にサインイン AWS マネジメントコンソール したり AWS Command Line Interface、(AWS CLI) を通じて IAM ロールを既存のユーザーアカウントに直接割り当てて AWS リソースを管理したりできます。

ディレクトリメンバーにコンソールへのアクセス権限を付与する際には、ディレクトリにアクセスするための URL を設定しておく必要があります。ディレクトリの詳細を表示してアクセス URL を取得する方法の詳細については、AWS Directory Service ドキュメントの「[ディレクトリ情報を表示する](#)」を参照してください。アクセス URL の作成方法の詳細については、Directory Service ドキュメントの「[アクセス URL の作成](#)」を参照してください。ディレクトリメンバーに IAM ロールを作成して割り当てる方法の詳細については、ドキュメントの「[ユーザーとグループに AWS リソースへのアクセス権を付与する](#)」を参照してください。Directory Service

Active Directory の以下の移行オプションを検討してください。

- Active Directory の拡張
- への移行 AWS Managed Microsoft AD
- 信頼を使用して Active Directory を に接続する AWS Managed Microsoft AD
- Active Directory DNS と Amazon Route 53 の統合

Active Directory の拡張

Active Directory インフラストラクチャが既にあり、Active Directory 対応ワークロードを に移行するときに使用する場合は AWS クラウド、 がお手伝い AWS Managed Microsoft AD します。[信頼](#)を使用して、既存の Active Directory AWS Managed Microsoft AD に接続できます。つまり、ユーザーは、ユーザー、グループ、またはパスワードを同期することなく、オンプレミスの Active Directory 認証情報を使用して Active Directory 対応アプリケーションと AWS アプリケーションにアクセスできます。たとえば、ユーザーは既存の Active Directory ユーザー名とパスワードを使用して AWS マネジメントコンソールと WorkSpaces にサインインできます。また、 で SharePoint などの Active Directory 対応アプリケーションを使用すると AWS Managed Microsoft AD、ログインした Windows ユーザーは、認証情報を再度入力することなく、これらのアプリケーションにアクセスできます。

信頼を使用するだけでなく、Active Directory をデプロイして AWS の EC2 インスタンス上で実行することで Active Directory を拡張できます。これは自分で行うか、AWS を使用してプロセスに役立てることができる場合があります。Active Directory を AWS に拡張する場合は、少なくとも 2 つのドメインコントローラーを異なるアベイラビリティーゾーンにデプロイすることをお勧めします。使用しているユーザーとコンピュータの数に基づいて 3 つ以上のドメインコントローラーをデプロイする必要がある場合がありますが AWS、障害耐性の理由から推奨される最小数は 2 つです。Active Directory Migration Toolkit (ADMT) と [Password Export Server \(PES\)](#) を使用して移行を実行することで AWS、オンプレミスの Active Directory ドメインを に移行し、Active Directory インフラストラクチャの運用上の負担を軽減することもできます。 <https://aws.amazon.com/blogs/security/how-to->

[migrate-your-on-premises-domain-to-aws-managed-microsoft-ad-using-admt/Active Directory Launch Wizard](#) を使用して AWS に Active Directory をデプロイすることもできます。

への移行 AWS Managed Microsoft AD

AWS で Active Directory を使用するには、2 つのメカニズムを適用できます。1 つの方法は、Active Directory オブジェクトを に移行 AWS Managed Microsoft AD するために を採用することです AWS。これには、ユーザー、コンピュータ、グループポリシーなどが含まれます。2 つ目のメカニズムは、すべてのユーザーとオブジェクトを手動でエクスポートし、次に [Active Directory 移行ツール](#) を使用してユーザーとオブジェクトを手動でインポートする方法です。

AWS Managed Microsoft AD に移行する理由は他にもあります。

- AWS Managed Microsoft AD は、 で Microsoft [Remote Desktop Licensing Manager](#)、[Microsoft SharePoint](#)、Microsoft [SQL Server Always On](#) などの従来の Active Directory 対応ワークロードを実行できるようにする実際の Microsoft Active Directory ドメインです AWS クラウド。
- AWS Managed Microsoft AD は、グループマネージドサービスアカウント (gMSAs) と Kerberos 制約付き委任 (KCD) を使用して、Active Directory 統合 .NET アプリケーションのセキュリティを簡素化および改善するのに役立ちます。詳細については、AWS ドキュメントの「[を使用した Active Directory 統合 .NET アプリケーションの移行の簡素化とセキュリティの向上 AWS Managed Microsoft AD](#)」を参照してください。

複数の AWS Managed Microsoft AD 間で共有できます AWS アカウント。これにより AWS のサービス、各アカウントと各 [Amazon EC2](#) などの を管理できます。Amazon Virtual Private Cloud ディレクトリは、内の任意の Amazon VPC から、AWS アカウントまたは任意の Amazon VPC から使用できます AWS リージョン。 <https://aws.amazon.com/vpc/> この機能により、複数のアカウントと VPC 間で単一のディレクトリを使用して、ディレクトリ対応のワークロードを優れたコスト効果で簡単に管理できます。例えば、単一の AWS Managed Microsoft AD ディレクトリを使用して、複数のアカウントと VPCs にわたって EC2 インスタンスにデプロイされた [Microsoft ワークロード](#) を簡単に管理できるようになりました。ディレクトリを AWS Managed Microsoft AD 別の と共有する場合 AWS アカウント、Amazon EC2 コンソールまたは を使用して [AWS Systems Manager](#)、アカウントおよび内の任意の Amazon VPC からインスタンスをシームレスに結合できます AWS リージョン。

各アカウントや Amazon VPC でディレクトリをデプロイしたり、手動でドメインにインスタンスを結合したりする必要がなくなるため、EC2 インスタンスにディレクトリ対応のワークロードを迅速にデプロイできます。詳細については、Directory Service ドキュメントの「[ディレクトリを共有する](#)」を参照してください。AWS Managed Microsoft AD 環境を共有するにはコス

トがかかることに注意してください。Amazon VPC ピアまたは Transit Gateway ピアを使用して、他のネットワークまたはアカウントから AWS Managed Microsoft AD 環境と通信できるため、共有が不要な場合があります。このディレクトリを次のサービスで使用する場合は、ドメインを共有する必要があります: Amazon Aurora MySQL、Amazon Aurora PostgreSQL、Amazon FSx、Amazon RDS for MariaDB、Amazon RDS for MySQL、Amazon RDS for Oracle、Amazon RDS for PostgreSQL、Amazon RDS for SQL Server。

で信頼を使用する AWS Managed Microsoft AD

既存のディレクトリのユーザーに AWS リソースへのアクセスを許可するには、AWS Managed Microsoft AD 実装で信頼を使用できます。AWS Managed Microsoft AD 環境間で信頼関係を作成することもできます。詳細については、AWS セキュリティブログの「[投稿のある信頼について知っておくべきこと AWS Managed Microsoft AD](#)」を参照してください。

Active Directory DNS と Amazon Route 53 の統合

に移行するときは AWS、を使用してサーバーへのアクセス Amazon Route 53 Resolver を許可することで (DNS 名を使用して)、DNS を環境に統合できます。これを行うには、DHCP オプションセットを変更するのではなく、Route 53 リゾルバーエンドポイントを使用することをお勧めします。こうすることで、DHCP オプションセットを変更するよりも DNS 設定を一元的に管理できます。さらに、さまざまなリゾルバルールを活用できます。詳細については、「Networking & Content Delivery Blog」の「[Integrating your Directory Service's DNS resolution with Amazon Route 53 Resolvers](#)」投稿と、AWS 「Prescriptive Guidance」ドキュメントの「[Set up DNS resolution for hybrid network in a multi-account AWS environment](#)」を参照してください。

移行

への移行を開始するときは AWS、移行に役立つ設定とツールのオプションを検討することをお勧めします。環境の長期的なセキュリティと運用面を考慮することも重要です。

次のオプションも検討してください。

- クラウドネイティブセキュリティ
- Active Directory を に移行するツール AWS

クラウドネイティブセキュリティ

- Active Directory コントローラーのセキュリティグループ設定 – を使用している場合 AWS Managed Microsoft AD、ドメインコントローラーには、ドメインコントローラーへのアクセスを

制限するための VPC セキュリティ設定が付属しています。ユースケースによっては、セキュリティグループのルールを変更してアクセスを許可する必要がある場合があります。セキュリティグループ設定の詳細については、Directory Service ドキュメントの [AWS Managed Microsoft AD 「ネットワークセキュリティ設定の強化」](#) を参照してください。ユーザーがこれらのグループを変更したり、他の AWS のサービスに使用したりすることを許可しないことをお勧めします。他のユーザーにこれらの機能の使用を許可すると、ユーザーが必要な通信をブロックするように変更を行った場合、Active Directory 環境のサービスが中断される可能性があります。

- Active Directory イベントログ用の Amazon CloudWatch Logs との統合 – AWS Managed Microsoft AD を実行している場合、または自己管理型の Active Directory を使用している場合は、Amazon CloudWatch Logs を利用して Active Directory のロギングを一元化できます。CloudWatch ログを使用して、認証、セキュリティ、およびその他のログを CloudWatch にコピーできます。これにより、1 か所でログを簡単に検索できるようになり、コンプライアンス要件を満たすのに役立ちます。CloudWatch Logs との統合をお勧めします。これにより、環境内の将来のインシデントにより適切に対応できるようになります。詳細については、ドキュメントの Directory Service [「 の Amazon CloudWatch Logs の有効化 AWS Managed Microsoft AD 」](#) および AWS ナレッジセンターの [「Windows イベントログの Amazon CloudWatch Logs」](#) を参照してください。

Active Directory を に移行するツール AWS

Active Directory 移行ツール (ADMT) とパスワードエクスポートサーバー (PES) を使用して移行を行うことをお勧めします。これにより、あるドメインから別のドメインにユーザーとコンピューターを簡単に移動できます。PES を使用する場合は、管理対象の Active Directory ドメインから別のマネージドドメインに移行する場合は、次の点に注意してください。

- ユーザー、グループ、およびコンピュータ用のアクティブディレクトリ移行ツール (ADMT) – [ADMT](#) を使用して、自己管理型の Active Directory から AWS Managed Microsoft AD にユーザーを移行できます。重要な考慮事項は、移行スケジュールとセキュリティ識別子 (SID) 履歴の重要性です。SID 履歴は移行中には転送されません。SID 履歴をサポートすることが不可欠な場合は、SID 履歴を維持できるように、ADMT ではなく、Amazon EC2 上の自己管理型の Active Directory を使用することを検討してください。
- パスワードエクスポートサーバー (PES) – PES を使用してパスワードを に移行できますが、外に移行することはできません AWS Managed Microsoft AD。ディレクトリからユーザーとパスワードを移行する方法については、Microsoft ドキュメントの AWS [「セキュリティログ」](#) および [「パスワードエクスポートサーバーバージョン 3.1 \(x64\)」](#) の [「ADMT AWS Managed Microsoft AD を使用してオンプレミスドメインを に移行する方法」](#) を参照してください。

- LDIF — LDAP データ交換フォーマット (LDIF) は、AWS Managed Microsoft AD ディレクトリのスキーマを拡張するために使用されるファイル形式です。LDIF ファイルには、ディレクトリに新しいオブジェクトや属性を追加するために必要な情報が含まれています。ファイルは LDAP の構文規格を満たしている必要があり、追加するオブジェクトごとに有効なオブジェクト定義が含まれている必要があります。LDIF ファイルを作成したら、ファイルをディレクトリにアップロードしてスキーマを拡張する必要があります。LDIF ファイルを使用して AWS Managed Microsoft AD ディレクトリのスキーマを拡張する方法の詳細については、Directory Service ドキュメントの「[AWS Managed Microsoft AD のスキーマの拡張](#)」を参照してください。
- CSVDE — 場合によっては、信頼を作成したり ADMT を使用したりせずに、ユーザーをディレクトリにエクスポートおよびインポートする必要がある場合があります。理想的ではありませんが、[Csvde](#) (コマンドラインツール) を使用すると Active Directory ユーザーをあるドメインから別のドメインに移行できます。Csvde を使用するには、ユーザー名、パスワード、グループメンバーシップなどのユーザー情報を含む CSV ファイルを作成する必要があります。その後、csvde コマンドを使用してユーザーを新しいドメインにインポートできます。このコマンドを使用して、ソースドメインから既存のユーザーをエクスポートすることもできます。これは、SAMBА ドメインサービスなどの別のディレクトリソースから Microsoft Active Directory に移行する場合に役立つ場合があります。詳細については、AWS セキュリティブログの「[Microsoft Active Directory ユーザーを Simple AD に移行する方法](#)」または [AWS Managed Microsoft AD](#)「」を参照してください。

その他のリソース

- [との信頼について知っておくべきこと AWS Managed Microsoft AD](#) (AWS セキュリティブログ)
- [ADMT AWS Managed Microsoft AD を使用してオンプレミスドメインを に移行する方法](#) (AWS セキュリティブログ)
- [ステップ 2: アクティブディレクトリのデプロイ](#) (AWS Windows ワークショップ)

Windows Server の移行

このセクションでは、Windows Server の への移行に使用できるさまざまなオプションに焦点を当てます AWS。

評価

まず、AWS に移行する必要があるアプリケーションとワークロードを特定します。[AWS Application Discovery Service](#) を使用して、オンプレミスのインフラストラクチャとアプリケーション間の依存

関係のマップを作成できます。これにより、AWS への移行が必要なサーバー、アプリケーション、サービスを特定できます。

[AWS Migration Hub](#) を使用してアプリケーションのインベントリを作成し、AWS との互換性を評価できます。Migration Hub はアプリケーションポートフォリオを一元的に表示するため、移行プロジェクトの計画、追跡、管理に役立ちます。Cloudamize や Evolve など AWS をサポートするサードパーティーの評価ツールを使用することもできます。

準備

大規模なインフラストラクチャをリホスト (リフトアンドシフト) するための適切な方法を見つけることは非常に難しい場合があります。役立つ [ベストプラクティス](#) は多数ありますが、どのツールを選択するかは、ワークロードのタイプ、許容できるダウンタイム、オペレーティングシステムの要件など、複数の要因によって決まります。リホストには [AWS Application Migration Service](#) を使用することをお勧めします。

AWS Application Migration Service

Application Migration Service を使用すると、互換性の問題、パフォーマンスへの影響、長いカットオーバー期間などに悩まされることなく、物理サーバー、仮想サーバー、またはクラウドサーバーを迅速にリフトアンドシフトできます。Application Migration Service は、ソースサーバーを継続的に AWS アカウントアカウントに複製します。その後、移行の準備が整うと、Application Migration Service は最小限のダウンタイムでサーバーを自動的に変換して起動します。詳細については、Application Migration Service ドキュメントの「[What Is AWS Application Migration Service?](#)」を参照してください。

AWS Transform for VMware

[AWS Transform](#) は、AI 主導のオーケストレーション AWS を使用して、サーバーとエンタープライズアプリケーションへの移行を簡素化および自動化します。移行ジョブを作成、実行、追跡するための単一のワークスペースを提供します。[AWS Transform for VMware](#) は、自動検出、インテリジェントなウェーブプランニング、リホスト機能を組み合わせて、中断を最小限に抑えながらワークロードを VMware 環境から Amazon EC2 に効率的に移行します。

AWS Transform は、以下を含む複数の移行ジョブタイプをサポートします。

- End-to-end の移行 – 検出、ウェーブプランニング、VPC 設定、サーバー移行について説明します。
- ネットワーク移行のみ – VPC ネットワーク設定を生成してデプロイします

- Network-and-server 移行 – VPC のセットアップとサーバーのリホストを組み合わせる
- 検出とサーバーの移行 – 検出を実行し、ウェーブプランを生成して、サーバーを移行します

AWS Transform は、VMware ネットワーク設定の Amazon VPC アーキテクチャへの AI 駆動型変換を使用し、アプリケーションのグループ化と推奨される移行ウェーブを含む移行計画を生成し、Windows および Linux サーバーのリホストを自動化して Amazon EC2 でネイティブに実行します。

VM Import/Export

[VM Import/Export](#) を使用すると、VM イメージを既存の仮想化環境から Amazon EC2 にインポートし、それを元の環境にエクスポートすることができます。この方法を使うと、アプリケーションおよびワークロードを Amazon EC2 へ移行したり、VM イメージカタログを Amazon EC2 にコピーしたり、バックアップと災害対策のために VM イメージのリポジトリを作成したりすることができます。詳細については、Amazon EC2 ドキュメントの「[What is VM Import/Export?](#)」を参照してください。

移行するワークロードを評価したら、移行戦略、タイムライン、移行プロセスに伴うコストを概説した移行計画を作成します。[AWS 価格設定/TCO ツール](#)を使用して、AWS でアプリケーションを実行することによるコスト削減を見積もることができます。

移行

Windows ワークロードを に移行するには、移行計画、準備状況評価、移行実装フェーズなど、いくつかのフェーズ AWS が必要です。移行フェーズは、Windows ワークロードを AWS に移行する最後のフェーズです。移行フェーズで考慮すべき手順は次のとおりです。

- AWS 環境の準備 – 移行プロセスを開始する前に、Amazon マシンイメージ (AMI) を作成し、ワークロードを移行する VPC を設定して環境を準備する AWS 必要があります。
- 移行ツールの選択 — Migration Hub、アプリケーション移行サービス、VM Import/Export など、さまざまな移行方法を選択できます。お客様のニーズに対して最適なものを選択してください。
- 移行の設定 — 移行元サーバーを選択し、移行先のインスタンスタイプ、ストレージ、ネットワーク設定を指定して移行を設定します。
- 移行の実行 — 設定が完了したら、移行を実行します。このプロセスには、データの複製、移行されたワークロードのテスト、移行されたワークロードへの切り替えのための最終的なカットオーバーの実行が含まれます。上記で選択した移行ツールが、これらの手順のガイドとなります。

- 移行の検証 — 移行が完了したら、移行したワークロードが想定どおりに機能していることを確認します。テストを実施し、セキュリティとコンプライアンスの要件が満たされていることを確認します。
- 移行したワークロードの最適化 — インスタンスのサイズを変更し、自動スケーリングを設定し、リザーブドインスタンスやスポットインスタンスなどのコスト削減戦略を実装して、移行したワークロードを最適化します。
- 移行されたワークロードの監視と管理 — 最適なパフォーマンスとセキュリティを確保するために、移行したワークロードを継続的に監視および管理します。モニタリングには [Amazon CloudWatch](#) を使用できます。

ファイルサーバーの移行

ストレージは、実行するワークロードに不可欠なコンポーネントです。には、ブロック、ファイル、オブジェクトストレージなど、クラウドにファイルを保存するための多くのオプション AWS があります。Microsoft ワークロードの場合、最も一般的なオプションはブロックとファイルストレージのオプションです。このセクションでは、Microsoft ワークロードのストレージを に移行するのに役立つ戦略 AWS クラウド を示し、ファイルサーバーの移行をガイドします。

評価

オブジェクト、ブロック、ファイルストレージの 3 つの主要なストレージタイプがあり、それぞれに分類できるストレージサービスの幅広いポートフォリオ AWS を提供します。移行を成功させるには、現在のニーズを理解し、それらをさまざまな AWS ストレージサービスと [比較して](#)、最適なものを判断する必要があります。ワークロードに適したテクノロジーを選択することが、長期的な成功の鍵です。現在使用しているストレージと完全に一致させようとすることは避けることをお勧めします。代わりに、利用可能なすべてのオプションを確認し、Microsoft ワークロードのコストとパフォーマンスを最適化するために最も理にかなったオプションを選択することをお勧めします。たとえば、ローカルのブロックストレージを必要とする大規模なオンプレミスファイルサーバーを考えてみましょう。では AWS、最適な選択肢として、ファイルサーバーとバックエンドストレージの管理に伴う差別化されていない手間を省きながら、[Amazon FSx](#) に移動してファイルサーバーと同じパフォーマンスを実現できます。

TCO は、どのストレージオプションが最適かを評価する際に評価すべき重要な項目です。AWS マネージドサービスを使用して運用コストを削減すると、適切な全体的なストレージソリューションを選択できることに注意してください AWS。ストレージ評価をリクエストするには、 [までお問い合わせ](mailto:migration-evaluator@amazon.com) ください migration-evaluator@amazon.com。ストレージスペシャリストは、ワークロードの

評価、最適な AWS ストレージサービスへのワークロードのマッピング、方向性のあるコスト見積もりの提供を支援します。ストレージ評価には次の 3 つのフェーズがあります。

1. 検出プロセスを開始するには、まずエージェントレスコレクターをインストールするか、既存のツールセットから出力をフラットファイルで取得します。
2. 検出プロセスを 7 ~ 60 日間実行します。
3. ストレージコレクターは検出ツールからのデータを分析し、ターゲットとなるストレージソリューションを提案して、そのソリューションにかかるコストの見積もりを提示します。

あるストレージオプションのコストが少し高い場合には、そのストレージオプションが長期的に全体のコストを削減できるかどうかを検討し、ストレージのセキュリティと信頼性を維持するためにチームが何をしなければならないかを調べます。これは、ワークロードに適した長期的ソリューションになる場合があります。

適切なソリューションを評価する際には、パフォーマンスとコストを見極めることが重要です。[Windows Performance Monitor](#) などのツールを使用して、ワークロードの IOPS、スループット、その他のパフォーマンスニーズを特定し、ワークロード用に選択した AWS ソリューションに同じテストを実装できます。さらに、Amazon CloudWatch エージェントを使用して [Windows サーバー上の Performance Monitor のメトリックスを表示](#) し、ワークロードを本番環境に移行する前にワークロードのメトリックスを分析できます。

ニーズに最適な AWS ストレージサービスを特定する

ストレージサービスの選択は通常、ユースケース、アプリケーションのニーズ、慣れ、パフォーマンスプロファイル、データ管理機能によって異なります。以下の点を考慮してください。

- [Amazon Simple Storage Service \(Amazon S3\)](#) – Amazon S3 はオブジェクトストレージであり、任意の場所の任意の量のデータを保存および取得するように構築されています。Amazon S3 では、ワークロードのデータアクセス、耐障害性、およびコスト要件に基づいて、幅広いストレージクラスを提供しています。[AWS Storage Gateway](#) を使用して、Amazon S3 へのファイルベースのアクセスを実装できます。これにより、サーバーメッセージブロック (SMB) を使用するアプリケーションを完全に書き直すことなく、Amazon S3 の低コストのストレージを活用できます。
- [Amazon Elastic Block Store \(Amazon EBS\)](#) – Amazon EBS は、Amazon EC2 インスタンスで使用するためのブロックレベルストレージボリュームを提供します。Amazon EBS ボリュームの動作は、未初期化のブロックデバイスに似ています。これらのボリュームは、デバイスとしてインスタンスにマウントできます。インスタンスにアタッチした Amazon EBS ボリュームは、インスタンスの有効期間とは無関係に存続するストレージボリュームとして公開されます。

- [Amazon FSx](#) – Amazon FSx には、NetApp ONTAP、OpenZFS、Windows ファイルサーバー、Lustre の 4 つの異なるファイルシステムがあります。適切なシステムを選択するためのガイドランスについては、「[Amazon FSx ファイルシステムの選択](#)」を参照してください。Amazon FSx は、さまざまなファイルシステムタイプのマネージドファイルストレージソリューションを提供し、Microsoft ワークロードを に移行 AWS したり、IT スタッフから運用上のオーバーヘッドの一部を削除したりできます。これにより、IT 部門は他の重要なビジネス推進要因に集中することができます。
- [AWS Snow Family](#) – ペタバイト単位のデータを移動する場合は AWS、Snow Family のストレージソリューションの使用を検討してください。ストレージはデータの長期存続のために Snow Family デバイスに依存しませんが、AWS Snowball Edge、AWS Snowball または AWS Snowmobile デバイスを使用して大規模なデータセットを AWS オフラインにシードするのに役立ちます。詳細については、「[を使用して大規模な SQL データベースをシームレスに移行する AWS Snowball](#)」および [AWS DataSync](#) AWS 「[ストレージブログ](#)」の投稿を参照してください。

本番データを移動する前に、ワークロードのストレージサービスを特定後、ストレス/負荷テストツールを使用してテストを実施することをお勧めします。たとえば、Amazon FSx for Windows File Server 上の SQL データベースを移動する場合、[Microsoft SQL Server Distributed Replay](#) を使用できます。同様に、[DISKSPD](#) は一般的な IOPS とスループットに使用できます。

準備

ストレージサービスを特定後の次のステップはデータ転送用のツールを選択することです。[Robocopy](#) のなどの古いソリューションや [AWS DataSync](#) などの最新のツールなど、いくつかのツールが利用可能です。DataSync には、スケジュールされた転送やネットワークスロットリングの簡単な制御など、Robocopy などのツールにはない多数の制御機能が含まれており、ネットワークトラフィック全体に影響を与えずにデータを移行できます。DataSync を使用して正常に完了した移行の詳細については、DataSync のお客様の「[お客様の声](#)」を参照してください。

Robocopy をより使い慣れている場合は、これを使用してデータを AWS に移行できます。[ファイル転送のパフォーマンス](#) を最適化する方法については、このガイドを確認することをお勧めします。このガイドは、移行中に問題が発生するのを防ぐのに役立ちます。重複排除が有効になっているファイルシステムで Robocopy を使用する場合は、データ破損の問題を回避するために、Amazon FSx for Windows File Server ドキュメントの「[Data deduplication](#)」と Microsoft ドキュメントの「[Troubleshooting Data Deduplication Corruptions](#)」を参照してください。

[AWS Storage Gateway](#) は、ファイル、ボリューム、仮想テープの AWS 3 つの方法でデータを に移行できます。Storage Gateway は、オンプレミスで実行されている VMware または Hyper-V ハイ

パーバイザー、Amazon VPC の Amazon EC2 インスタンス、または専用のハードウェアアプライアンスにインストールできます。

Storage Gateway は、オンプレミスからへのギャップを埋め AWS、コストを削減するのに役立ちます。Storage Gateway を使用して移行を段階的に実装し、それを使用してオンプレミスのバックアップデバイスとテープを仮想テープライブラリ (VTL) に置き換えることができます。Storage Gateway をアーカイブストレージソリューションとして使用して、ローカルの未使用のファイルのみを移行の最初のフェーズ AWS としてへの移行を開始することもできます。Storage Gateway を使用して Microsoft ワークロードをホストするには、いくつかのオプションがあります AWS。

移行

DataSync と Robocopy はどちらも、ネットワークアクセスコントロールリスト (ACL、Windows ACL とも呼ばれる) を保存する機能を備えています。移行を開始する前に、[icacls](#) を使用して ACL のバックアップコピーを作成し、以下のリソースを確認することをお勧めします。

- [オンプレミスから Amazon FSx for NetApp ONTAP へのファイル共有の移行](#) (AWS ストレージブログ)
- [Migrating existing file storage to Amazon FSx](#) (Amazon FSx for Windows File Server ドキュメント)
- [を使用して VPC を離れることなく、オンプレミスから AWS にファイルを転送する AWS DataSync](#) (AWS ストレージブログ)
- (AWS 規範ガイド) [を使用して、小規模なデータセットをオンプレミスから Amazon S3 に移行する AWS SFTP](#)

SQL Server の移行

クラウドへの移行には、SQL Server 環境を AWS に移行するための複数の選択肢があります。[移行](#)を成功させるには、SQL Server ワークロードとその依存関係の詳細なインベントリの作成、認証スキームの特定、高可用性とディザスタリカバリ (HADR) の要件の把握、パフォーマンス目標の評価、[ライセンスオプション](#)の評価が必要です。このインベントリは、ターゲットデータベースプラットフォームを決定し、移行オプションを定義するのに役立ちます。

SQL Server ワークロードをに移行するときに考慮すべき多くのオプションがあり AWS、それぞれが最適化された価格/パフォーマンス、より直感的なユーザーエクスペリエンス、TCO の削減につながります。SQL Server は、[Amazon EC2](#)、[Amazon RDS for SQL Server](#)、または [Amazon RDS Custom for SQL Server](#) にデプロイできます。

評価

移行を成功させるには、既存のインフラストラクチャを評価し、その環境に必要な主な機能を理解することが重要です。移行計画を選択する前に、次の主要なエリアを確認することをお勧めします。

- 既存のインフラストラクチャの確認 – 移行の検出フェーズで収集されたデータを使用して、既存の SQL Server インフラストラクチャを確認します。[AWS Migration Evaluator](#) を使用して、サーバー設定、SQL Server デプロイ、リソース使用率、アプリケーションの依存関係に関する詳細情報を自動的に収集できます。VMware ベースの環境では、[AWS Transform 検出ツール](#)は、クラウド接続を必要とせずに、エージェントレスのオンプレミス検出を提供します。その出力は、TCO 分析とビジネスケース生成 AWS Transform の評価に直接フィードされます。SQL Server インフラストラクチャには Microsoft の規定サイズ設定を使用することをお勧めします AWS。メモリ、CPU、IOPS、スループットなど、オンプレミス SQL Server インスタンスの現在の使用率を理解することは、SQL Server インスタンスのサイズを適切に設定するために重要です AWS。
- 既存のライセンスの確認 – 補完的な[AWS 最適化とライセンス評価 \(AWS OLA\)](#) を活用して、で移行とライセンス戦略を構築できます AWS。AWS OLA には、既存のライセンス使用権限を使用してデプロイオプションをモデル化するレポートが用意されています。これらの結果は、柔軟な AWS ライセンスオプション全体で実現可能なコスト削減の検討に役立ちます。SQL Server ワークロードを既に実行している場合 AWS、[AWS Compute Optimizer](#)は、実際の機能の使用状況に基づいて SQL Server エディションをダウングレードする機会を特定するなど、ライセンスに関する自動レコメンデーションを提供します。
- 既存の SQL Server アーキテクチャを確認する – 共有ストレージまたは SQL Server Always On 可用性グループアーキテクチャで SQL Server フェイルオーバークラスターを使用している場合、現在の高可用性アーキテクチャ要件を理解することで、[SQL Server のデプロイオプション](#)を定義できます AWS。

SQL Server Always On 可用性グループは、同期コミットモードと非同期コミットモードの両方をサポートし、単一 AWS リージョン (アベイラビリティゾーン間) 内の高可用性、またはリージョン間のディザスタリカバリに使用できます。SQL Server Always On フェイルオーバークラスターインスタンス (FCIs) には共有ストレージが必要です。共有ストレージは、[Amazon FSx for Windows File Server](#) または [Amazon FSx for NetApp ONTAP](#) を使用して提供できます。高可用性とディザスタリカバリオプションの完全な比較については、AWS 「規範ガイド」の「[高可用性とディザスタリカバリソリューションの選択](#)」を参照してください。

- バックアップ戦略の開発 – Amazon RDS for SQL Server では、point-in-timeリカバリ、手動スナップショット、ネイティブバックアップと復元で自動バックアップを使用できます。Amazon EC2 の SQL Server では、ネイティブ SQL Server のバックアップと復元を使用するか、スナップショットアプローチを使用するか、データベースを Amazon EBS、Amazon FSx for Windows

File Server、Amazon FSx for NetApp ONTAP、または Amazon S3 にバックアップできます。を使用して [AWS Backup](#)、Amazon EC2 上の Amazon RDS for SQL Server と SQL Server 間でバックアップをオーケストレーションおよび一元化できます。

Amazon FSx for NetApp ONTAP を使用した Amazon EC2 上の SQL Server 2022 は、プライマリホストへの影響を最小限に抑えながら、ほぼ瞬時に一貫した [バックアップのための T-SQL スナップショット](#) バックアップをサポートします。SQL Server 2025 は、Always On 可用性グループでセカンダリレプリカからのネイティブデータベースバックアップを有効にすることで、これをさらに拡張します。詳細については、(AWS ブログ記事) の [「Microsoft SQL Server 2025 の新機能 AWS」](#) を参照してください。

バックアップ戦略の詳細については、[「Amazon RDS for SQL Server のバックアップおよび復元戦略」](#) (AWS ブログ記事) および [Amazon EC2 上の SQL Server のバックアップおよび復元オプション](#) (AWS 規範ガイド) を参照してください。

- デザスタリカバリ (DR) のニーズを理解する – Amazon RDS for SQL Server の場合、クロスリージョン自動バックアップとリードレプリカは、SQL Server レベルのレプリケーション設定を必要とせずにマネージド DR オプションを提供します。

Amazon EC2 の SQL Server では、[AWS Transit Gateway](#) または を介して AWS リージョン 接続されたセカンダリを使用して [AWS Direct Connect](#)、レプリケーションを実行できます。DR オプションには、マルチリージョンデプロイ用の SQL Server 分散可用性グループ、RTO と RPO を使用したコスト効率の高いオプションのための数分以内のログ配信、アクティブ/パッシブ DR 実装としての継続的なブロックレベルのレプリケーション [AWS Elastic Disaster Recovery](#) などがあります。詳細については、AWS 「Database Blog」の [「Choose a high availability and disaster recovery solution on AWS Prescriptive Guidance」](#) と [「Architect a disaster recovery for SQL Server on AWS: Part 1」](#) を参照してください。

準備

[SQL Server ワークロードには、SQL Server データベース移行戦略](#) を検討することをお勧めします。

- リホスト (リフトアンドシフト) — オンプレミスの SQL Server データベースを AWS クラウドの Amazon EC2 インスタンス上の SQL Server に移行します。このアプローチは、への移行を高速化することが優先される場合 AWS に便利です。Bring Your Own License (BYOL) モデルを使用して既存の SQL Server ライセンスを持ち込むか、ライセンス込み (LI) インスタンスを購入できます AWS。 [AWS Launch Wizard for SQL Server](#) を使用して、Amazon EC2 での SQL Server のサイズ

設定、設定、デプロイをガイドすることもできます。シングルインスタンスデプロイと高可用性デプロイの両方をサポートします。

- **リプラットフォーム (リフトアンドリシェイプ)** – これには、オンプレミスの SQL Server データベースをマネージドデータベースサービスに移行することが含まれます AWS。このアプローチは、インストール、設定、パッチ適用、アップグレード、高可用性設定など、差別化されていないタスクをオフロードします。次の 2 つのマネージドオプションから選択します。
 - [Amazon RDS for SQL Server](#) – これは、すべてのデータベースインフラストラクチャ管理をオフロードする場合に最適なフルマネージドオプションです。
 - [Amazon RDS Custom for SQL Server](#) – これは、オペレーティングシステムとデータベースレベルのアクセスが保持されているマネージドサービスです。このオプションは、カスタムデプロイ要件を持つレガシーアプリケーションまたはパッケージ化されたアプリケーションに適しています。Amazon RDS Custom は、独自のメディア持ち込み (BYOM) オプションをサポートしています。これにより、Microsoft のライセンスモビリティ条件に従って既存の SQL Server ライセンスを使用できます。

Amazon EC2、Amazon RDS、Amazon RDS Custom での SQL Server の機能比較については、「[規範ガイドでの Amazon EC2 と Amazon RDS の選択](#)」を参照してください。AWS

- **リファクタリング (リアーキテクト)** – これには、通常、オープンソースデータベースまたはクラウド用に構築されたデータベースを使用してアプリケーションを変更し、モダナイズする必要があります。SQL Server から移行することで、ライセンスコストを削減し、ベンダーのロックインとライセンス監査を回避できます。SQL Server データベースをモダナイズして、以下を行うことができます。
 - [Amazon RDS for MySQL](#) または [Amazon RDS for PostgreSQL](#) – フルマネージド型のオープンソースデータベースサービス。
 - [Amazon Aurora](#) – MySQL と PostgreSQL の完全な互換性を持つクラウドネイティブのリレーショナルデータベース。商用グレードのデータベースのパフォーマンスと可用性をわずかなコストで実現します。
 - [Babelfish for Aurora PostgreSQL](#) – SQL Server 用に最初に記述されたアプリケーションが、最小限のコード変更で Aurora PostgreSQL と連携し、移行を加速し、リファクタリングリスクを軽減できるようにします。

SQL Server スキーマとコードを変換するには、() のフルマネージドスキーマ変換機能である [AWS DMS Schema Conversion](#) を使用できます AWS Database Migration Service AWS DMS。

移行

SQL Server ワークロードを に移行する際 AWS、以下のセクションでは、各移行戦略で使用できるツールとアプローチについて説明します。

リホスト

リホストは [同種移行アプローチ](#) です。データベースソフトウェアや設定を変更せずに SQL Server データベースをそのまま移行する場合は、このオプションを選択します。これは、速度が優先される大規模なレガシー移行の一般的な選択肢です。

Amazon EC2 を使用した SQL サーバーの移行

Amazon EC2 に移行する場合は、BYOL モデルを使用して既存の SQL Server ライセンスを持ち込むか、 から LI インスタンスを購入できます AWS。 [AWS License Manager](#) は、Amazon EC2 に SQL Server をデプロイするときに使用可能なライセンスの割り当てを制御し、ライセンスルールに準拠するのに役立ちます。

BYOL アプローチでは、 [Microsoft Software Assurance \(SA\) がある場合にのみ、SQL Server を共有テナンシー \(デフォルト\)](#) Amazon EC2 インスタンスにリホストできます。SQL Server ライセンスに SA がない場合は、ライセンスが 2019 年 10 月 1 日より前に購入されたか、その日より前に有効なアクティブなエンタープライズ登録でライセンスを補正として追加した場合、 [Amazon EC2 Dedicated Hosts](#) にリホストできます。詳細については、 [「での Microsoft Licensing AWS」](#) を参照してください。

SQL Server 機能または を使用して、SQL Server データベースを Amazon EC2 インスタンスに移行できます AWS のサービス。これらのオプションは、1 つのデータベースまたは一連のデータベースを Amazon EC2 の新しい SQL Server インスタンスに移行する場合に適しています。データベース移行に加えて、ログイン、ジョブ、データベースメール、リンクサーバーなどのオブジェクトの移行が必要になる場合があります。

で SQL Server データベースをリホストするには、次のアプローチを使用できます AWS。

- を使用したサーバーのリホスト [AWS Application Migration Service](#)
- [SQL Server のバックアップと復元](#)
- [SQL Server のトランザクションレプリケーション](#)
- [可用性グループをクラウドに拡張する](#)
- [AWS Database Migration Service \(AWS DMS\)](#)
- [ログ配布](#)

[AWS Launch Wizard for SQL Server](#) を使用して、単一インスタンスと高可用性の両方のデプロイをサポートする Amazon EC2 での Microsoft SQL Server のサイズ設定、設定、デプロイをガイドすることもできます。

を使用した SQL Server の移行 AWS Application Migration Service

[AWS Application Migration Service](#) は、ほぼゼロまたは最小限のダウンタイムでデータベース内の SQL Server バージョン、オペレーティングシステム、またはコードを変更 AWS せずに、1 つ以上の大規模なマシンをオンプレミス環境から リフトアンドシフトする場合に適しています。を使用すると AWS Application Migration Service、互換性の問題、パフォーマンスへの影響、または長いカットオーバーウィンドウなしに、物理サーバー、仮想サーバー、またはクラウドサーバーをすばやくリフトアンドシフトできます。Application Migration Service を使用してオンプレミス環境から Amazon EC2 インスタンスに SQL Server データベースを移行するガイドランスについては、AWS 「規範ガイド」の「[Microsoft SQL Server データベースの への移行 AWS クラウド](#)」を参照してください。Application Migration Service を使用して Microsoft SQL Server データベースのワークロードを AWS に移行する際の[ベストプラクティス](#)を参照することもできます。

Linux 上の SQL サーバー

SQL Server データベースエンジンは、Windows Server と Linux の両方で同様の方法で実行されます。ただし、Linux を使用する場合、特定のタスクにはいくつかの変更があります。[AWS Launch Wizard](#) は、これらの変更を調整し、可用性の高いソリューションを設定するのに役立ちます。社内に Linux 管理の専門知識がある場合は、Windows Server のライセンスコストを節約するために、Amazon EC2 Linux にリホストすることをお勧めします。Linux 上の SQL Server は、SQL Server 2017 以降でサポートされています。詳細については、「Linux on AWS Prescriptive Guidance」の「[Migrate an on Microsoft SQL Server on Amazon EC2](#)」を参照してください。

リプラットフォーム

リプラットフォームは[同種](#)のアプローチで、完全マネージド型のデータベース製品を使用してデータベースインスタンスの管理に費やす時間を短縮するのに最適です。Amazon RDS for SQL Server の完全マネージド型データベースでは、基盤となるオペレーティングシステム、システムボリューム、カスタムドライバーのインストールなどへのアクセスが制限されます。詳細については、「[Amazon RDS for Microsoft SQL Server](#)」を参照してください。OS レベルのアクセスまたは既存の SQL Server ライセンスが必要な場合は、[Amazon RDS Custom](#) for SQL Server への再プラットフォームを検討してください。

Amazon RDS Custom for SQL Server は BYOM ライセンスモデルをサポートしています。これにより、独自のインストールメディアとライセンスを使用できます。ライセンスは、Microsoft License

[Mobility](#) の条件に準拠している必要があります。SQL Server のプラットフォームを Amazon RDS for SQL Server または Amazon RDS Custom for SQL Server にリプラットフォームできます。選択は、基盤となるオペレーティングシステムに必要なカスタマイズレベル、Amazon RDS for SQL Server で必要な機能がサポートされているかどうか、BYOM を使用して既存の SQL Server ライセンスを使用するかどうかによって異なります。

SQL Server を Amazon RDS for SQL Server または Amazon RDS Custom for SQL Server に移行するために、次のオプションを使用できます。

- カスタムログ配信 – Amazon RDS for SQL Server と Amazon RDS Custom のカスタムスクリプトが必要です。リファレンス実装については、AWS データベースブログの「[Automate on-premises or Amazon EC2 SQL Server to Amazon RDS for SQL Server migration using custom log shipping](#)」を参照してください。
- SQL Server のバックアップと復元 – Amazon RDS for SQL Server のバックアップと復元については、「[ネイティブバックアップと復元を使用した SQL Server の Amazon RDS への移行](#)」を参照してください。Amazon RDS Custom については、「[Migrate on-premises SQL Server to Amazon RDS Custom for SQL Server using native backup and restore and Amazon S3](#)」を参照してください。
- [トランザクションレプリケーション](#)
- [AWS Database Migration Service \(AWS DMS\)](#)

詳細については、AWS 「規範ガイド」の「[SQL Server の移行方法](#)」を参照してください。

SQL Server データベースをリプラットフォームして Amazon RDS for SQL Server で実行するには、[Amazon RDS for SQL Server](#) で提供されているアプローチを使用することを検討してください。サポート終了ワークロードを移行する方法については、AWS データベースブログの「[サポート終了 Microsoft SQL Server データベースを Amazon RDS for SQL Server に確実に移行する](#)」を参照してください。オンプレミスデータベースを Amazon RDS Custom for SQL Server に移行する方法については、Amazon RDS ドキュメントの「[オンプレミスデータベースを Amazon RDS Custom for SQL Server に移行する](#)」を参照してください。

リファクタリング

リファクタリングは[異種混在](#)です。データベースとアプリケーションを再構築、書き換え、再構築する準備ができれば、このアプローチを選択して、オープンソースと built-for-the-cloud されたデータベースサービスを活用します。データベースとそれぞれのアプリケーションをリファクタリングすることに前向きな場合は、SQL Server ワークロードを Amazon RDS for MySQL、Amazon RDS for PostgreSQL、[Amazon Aurora MySQL 互換エディション](#)、または [Amazon Aurora PostgreSQL 互換](#)

[エディション](#)のいずれかにモダナイズできます。さまざまなモダナイズのスケジュールやパフォーマンス要件に応じてリファクタリングできます。

Amazon RDS for MySQL と Amazon RDS for PostgreSQL は、それぞれのオープンソースデータベース向けのフルマネージド型データベース製品です。Amazon Aurora は、MySQL と PostgreSQL との完全な互換性を備えたクラウド用に構築されたリレーショナルデータベース管理システム (RDBMS) です。Aurora は耐障害性に優れたストレージシステムを備え、商用グレードのデータベースのパフォーマンスと可用性を 10 分の 1 のコストで実現します。

[Amazon Aurora Serverless](#) を使用して、データベース容量を管理 AWS せずに データベースを実行することもできます。Amazon Aurora Serverless v2 は、ほんの一瞬で数十万件のトランザクションに瞬時にスケールリングできます。アプリケーションが消費する容量に対してのみ支払いを行い、ピーク負荷の容量をプロビジョニングするコストと比較して、データベースコストを最大 90% 削減できます。

SQL Server データベースをこれらのサービスのいずれかにリファクタリングするには、次のいずれかの使用を検討してください。

- [AWS Transform for SQL Server Modernization](#) は、SQL Server データベースおよび関連する .NET アプリケーションの Amazon Aurora PostgreSQL へのフルスタックのモダナイゼーションを自動化します。スキーマ変換、ストアドプロシージャ変換 (T-SQL から PL/pgSQL)、を介したデータ移行 AWS DMS、アプリケーションコードの更新 (エンティティフレームワーク、ADO.NET、接続文字列) など、移行ジャーニー全体をオーケストレーションします。また、重要な段階で human-in-the-loop チェックポイントも提供します。サポートされている SQL Server のバージョン、ソース、ターゲットの詳細については、ドキュメントの「[サポートされているバージョンとプロジェクトタイプ](#)」を参照してください。AWS Transform
- Amazon RDS for MySQL、Amazon RDS for PostgreSQL、またはその他の Aurora ターゲットへのスキーマのみの変換または移行の場合は、[AWS DMS スキーマ変換](#)の使用を検討してください。
- アプリケーションとデータベースの移行を高速化することが目標の場合は AWS、[Babelfish for Aurora PostgreSQL](#) の使用を検討してください。Babelfish を使用すると、SQL Server 用に最初に記述されたアプリケーションは、最小限のコード変更で Amazon Aurora と連携できます。その結果、SQL Server 2019 以前向けに開発された Babelfish for Aurora PostgreSQL アプリケーションを変更し、に移行するのに必要な労力が軽減され、より迅速でリスクが低く、費用対効果の高いリファクタリングが可能になります。

Babelfish を使用した移行には、以下のリソースを検討してください。

- [Babelfish を使用して SQL Server から Amazon Aurora に移行する](#) (AWS データベースブログ)
- [評価レポートによる AWS SCT Babelfish 移行の準備](#) (AWS データベースブログ)

- [SSIS と Babelfish を使用して SQL Server から Aurora PostgreSQL に移行する](#) (AWS データベースブログ)
- [AWS Database Migration Service のターゲットとしての Babelfish の使用](#) (AWS Database Migration Service ドキュメント)

詳細については、「[規範ガイド](#)」の「[異種データベース移行のツール](#)」を参照してください。AWS

その他のリソース

- [Microsoft SQL Server データベースの への移行 AWS クラウド](#) (AWS 規範ガイド)
- [での SQL Server ワークロードの移行とモダナイゼーション戦略 AWS](#) (AWS ブログ)
- [SQL Server データベースの移行方法](#) (AWS 規範ガイド)

.NET アプリケーションの移行

.NET アプリケーションを に移行 AWS することで、伸縮自在なスケーリング機能を備えた高可用性ワークロードを作成し、運用オーバーヘッドを削減し、差別化する価値に焦点を当てることでビジネスの俊敏性を高めることができます。

このセクションでは、.NET アプリケーションを AWS でホストするためのさまざまなオプションに焦点を当てます。VM、[AWS Elastic Beanstalk](#) などのマネージドソリューションの使用、コードのコンテナ化、マイクロサービスまたはサーバーレスベースのアーキテクチャへのコードのリファクタリングのいずれかを選択できます。

評価

.NET ワークロードの移行パスの選択は、次の重要な要素に依存します。

- 使用されている .NET バージョンを確認する — Microsoft がサポートしている .NET 実装には、.NET Framework (1.0—4.8) と .NET (.NET Core 1.0—3.1 と .NET 5 以降) の 2 種類があります。どちらも同じコンポーネントの多くを共有しており、異なる .NET プログラミング言語 (C#、F#、VB.NET など) を使用して記述されたアプリケーションコードを実行できます。.NET Framework は Windows 上で実行されますが、新しい .NET はマルチプラットフォームであるため、移行戦略とホスティングサービスの選択は、使用されるランタイムによって異なります。.NET Framework では、Windows OS でホストすることも、新しい .NET を使用するようにコードをリファクタリングすることもできます。新しい .NET は、Linux OS ベースのサービ

スでもホストできます。.NET Framework ベースのワークロードをモダナイズする場合、[AWS Transform for .NET](#) を使用してコードをスキャンし、互換性評価レポートを生成できます。プロジェクトで参照されている互換性のない .NET Framework API があるかどうかを確認することで、移行プロジェクトの複雑性を計画し、新しいランタイムを使用するようにコードをリファクタリングするかどうか、いつリファクタリングするかを決定できます。

- 現在のデプロイの確認 — 現在移行されているワークロードに、同じワークロードをクラウドにデプロイするように更新できる既存の CI/CD パイプラインがあるかどうかを確認します。既存のビルドおよびデプロイパイプラインを使用すると、ワークロードの構築、構成、デプロイに必要な手順が自動化され、アプリケーションをクラウドにデプロイするのにかかる時間を短縮できます。
- ロードマップの見直し — プロジェクトの現在の状態によっては、アプリケーションのリアーキテクトや再設計をすでに計画している場合があります。モダナイズを行う際には、製品ロードマップを考慮に入れる必要があります。たとえば、既存のコードをコンテナ化したり、モノリシックアーキテクチャをマイクロサービスにリファクタリングしたりすることは、製品ロードマップの一部であり、他の開発努力と連動するのが理想的です。

準備

.NET ワークロードを AWS に移行する際に考慮すべき移行パスは 3 つあります。既存のコードベースの複雑さ、移行に割り当てられる時間、移行作業をサポートするために割り当てられたチームの規模に応じて、さまざまなオプションを選択できます。移行の一環としてモダナイズを検討する場合、製品のロードマップに合わせて調整するのがベストプラクティスです。

- リホスト (リフトアンドシフト) - ほとんどまたはまったく変更せずに への移行を高速化することを優先する場合は AWS、このアプローチを選択できます。ASP.NET ベースのウェブサイトを Amazon EC2 インスタンスで実行されているインターネットインフォメーションサービス (IIS) にリホストできます。デスクトップベースのアプリケーション (Windows Presentation Foundation、Web Forms、.NET MAUI など) は、[Amazon WorkSpaces アプリケーション](#) や [Amazon WorkSpaces](#) などのエンドユーザーコンピューティングプラットフォームのいずれかにリホストできます。
- リプラットフォーム — リプラットフォームが最適であるケースは、コードを変更せずにマネージドサービスを使用してアプリケーションをホストしたいが、インストール、パッチ適用、アップグレード、インスタンス管理などの差別化されていない面倒な作業をオフロードすることで運用上のオーバーヘッドを削減したい場合です。この戦略は、コンテナベースのワークロードに移行したいチームにも適しています。既存のアプリケーションを [Elastic Beanstalk](#) にリプラットフォームすることも、[Amazon ECS](#)、[Amazon EKS](#)、または [AWS App Runner](#) でホストされている Docker コンテナを使用することもできます。

- リファクタリング – AWS クラウドネイティブサービスを使用して、運用オーバーヘッドを削減し、スケーリング、高可用性、ディザスタリカバリを向上させるコードとアーキテクチャの変更に時間と労力を費やすことができる場合は、このアプローチを選択します。リファクタリングでは、既存の .NET フレームワークアプリケーションを .NET (以前の .NET Core) に移植したり、既存のコードベースをモダナイズしてクラウドでの動作を向上したりして、コードベースをモダナイズする必要があります。[AWS SDK for .NET](#) を使用して、.NET コード内から多くの AWS クラウドサービス呼び出すことができます。[AWS Transform for .NET](#) などのツールを使用して、コードベースを .NET Framework から .NET に移植できます。既存の .NET ワークロードを [AWS Lambda](#) で実行するようにリファクタリングすることで、サーバーレスコンピューティングを使用してインフラストラクチャのプロビジョニングと管理を回避できます。

移行

.NET ワークロードの移行手順は、評価段階で選択した移行パスとアプリケーションの種類によって異なります。

.NET アプリケーションをリホストする

コードを変更せずにアプリケーションを移行したいが、クラウドでの自動スケーリング、負荷分散、伸縮性のメリットを利用したい場合は、この移行パスを選択してください。Windows ベースのウェブサイトの場合、リホストとは通常、AWS 上の Internet Information Services (IIS) で実行することを意味します。デスクトップベースのアプリケーションでは、アプリケーションをインストールして、ユーザーが外部からアプリケーションに接続できるようにする必要があります。

でのインターネットインフォメーションサービス AWS

Internet Information Services (IIS) は、Windows オペレーティングシステム上で動作する Microsoft Web サーバーで、Web サイトや Web サービスのホストに使用されます。IIS は、Windows Server を実行中の Amazon EC2 インスタンスにインストールできます。IIS を有効にして設定したら、オンプレミス環境と同じデプロイメカニズムを使用して ASP.NET ウェブサイトとサービスをデプロイできます。

EC2 Windows インスタンスで IIS をホストする場合、ワークロードと HADR のニーズに応じて、負荷分散、Auto Scaling グループ、マルチ AZ 配置を使用して [AWS Well-Architected フレームワーク](#) に従うことが重要です。IIS リソースを実行している Windows Server ワークロードのサイズ設定、設定、デプロイをガイド [AWS Launch Wizard](#) するため、を使用することをお勧めします AWS。Launch Wizard は、新しく作成された VPC または既存の VPC に必要なコンピューティン

グ、ネットワーク、ストレージコンポーネントを備えた 2 つの Availability Zones にまたがる高可用性アーキテクチャをデプロイします。

でのデスクトップアプリケーションのホスティング AWS

多くのクライアントは、Windows ベースのシッククライアントアプリケーションにアクセスする必要があります。次の 3 つのプラットフォームから選択できます。

- [Amazon EC2](#) — ユーザーが Microsoft リモートデスクトップを使用して Windows サーバーベースの環境に接続できるようにする場合、このオプションを選択します。このオプションでは、オペレーティングシステムにパッチを適用し、メンテナンスを行う必要があります。また、ユーザー用のリモートデスクトップサービスクライアントアクセスライセンス (RDS CAL) と [有効なソフトウェアアシュアランス \(SA\)](#) を追加購入する必要があります。詳細については、AWS ドキュメントの Microsoft [Licensing on AWS](#) を参照してください。
- [Amazon WorkSpaces](#) — ユーザー向けにフルマネージド型の仮想デスクトップインフラストラクチャ (VDI) が必要な場合は、このオプションを選択してください。WorkSpaces を使用すると、ユーザーに永続的な Windows デスクトップエクスペリエンスを提供できます。WorkSpaces 環境をカスタマイズし、カスタムイメージを使用して .NET アプリケーションをインストールしたり、[AWS Systems Manager](#) を使用して .NET アプリケーションを WorkSpaces 環境に配信したりすることもできます。ユーザーは、ブラウザまたは [Amazon WorkSpaces クライアント](#) を使用して接続できます。
- [Amazon WorkSpaces アプリケーション](#) — 任意の場所からアプリケーションや非永続的なデスクトップに、安全で信頼性が高く、スケーラブルなアクセスを提供するには、このオプションを選択します。WorkSpaces アプリケーションを使用して、ユーザーがウェブから .NET アプリケーションにアクセスできるようにします。既存の RDS CALs とアクティブな SA がすでにある場合は、ライセンスモビリティを使用して WorkSpaces アプリケーションでこれらのライセンスを使用できます。 <https://aws.amazon.com/windows/resources/licensemobility/>

リプラットフォーム

リプラットフォームでは、コードをほとんどまたはまったく変更せずにホスティング環境を変更します。運用上のオーバーヘッドを減らし、クラウドの機能とサービスを活用するには、この戦略を選択します。

AWS Elastic Beanstalk

[AWS Elastic Beanstalk](#) を使用して、.NET フレームワークのワークロードをリプラットフォームできます。ASP.NET ベースまたは ASP.NET Core ベースのアプリケーションをパッケージ化すると、

それらのアプリケーションを実行するインフラストラクチャについて知る AWS ことなく、アプリケーションをすばやくデプロイおよび管理できます。これによって、選択肢やコントロールを制限することなく、複雑さを緩和できます。アプリケーションをアップロードするだけで、Elastic Beanstalk は容量のプロビジョニング、ロードバランシング、スケーリング、およびアプリケーション状態モニタリングといった詳細を自動的に処理します。

詳細については、以下のリソースを参照してください。

- [Creating and deploying .NET applications on Elastic Beanstalk](#) (Elastic Beanstalk ドキュメント)
- [Working with .NET Core on Linux](#) (Elastic Beanstalk ドキュメント)
- [.NET および のカスタムドメインを使用したマルチアプリのサポート AWS Elastic Beanstalk](#) (AWS 開発者ツールブログ)

既存のアプリケーションをコンテナ化する

Amazon ECS または Amazon EKS を使用して、Docker ベースのコンテナ化されたアプリケーションをホストできます。は両方のサービス AWS を管理します。この 2 つのどちらを選択するかは、既存の知識と好みによります。どちらのオプションも、Linux ベースのコンテナまたは Windows ベースのコンテナのどちらでも実行できます。

詳細については、以下のリソースを参照してください。

- [Amazon EC2 Windows containers](#) (Amazon ECS ドキュメント)
- [Enabling Windows support for your Amazon EKS cluster](#) (Amazon EKS ドキュメント)
- [Running Windows Containers with Amazon ECS on AWS Fargate](#) (AWS ブログ)
- [EC2 Image Builder とイメージキャッシュ戦略による Windows コンテナの起動時間の短縮](#) (AWS ブログ)
- [Quick start: CI/CD for .NET Applications on AWS Fargate](#) (AWS ドキュメント)

.NET ベースのアプリケーションのコンテナ化は、使用する .NET ランタイムによって異なります。以下の点を考慮してください。

- .NET Framework ベースのアプリケーションを Windows コンテナ上で実行 — 既存のアプリケーションに Docker サポートを追加するには、アプリケーションをコンテナ化する方法の概要を記述した Docker ファイルを作成します。
- .NET または .NET Core — 新しい .NET ベースのウェブアプリケーションを Amazon ECS または Amazon EKS で実行することに加えて、[AWS App Runner](#) を使用することもできます。App

Runner は、コードまたはコンテナ イメージを実行し、負荷分散、自動スケーリング、ログ、証明書、およびネットワークを管理する、サーバーレスのフルマネージドソリューションです。

既存のコードのリファクタリング/リアーキテクト

アプリケーションの現在の環境で、他の方法では達成することが難しい機能、スケーリング、またはパフォーマンスを追加するというビジネス上の強いニーズがある場合は、このオプションを選択してください。アプリケーションのロードマップに応じて、最新のフレームワークやクラウドネイティブサービスを使用するようにコードを変更するか、クラウドでより適切に実行できるようにコードを再構築するかを選択できます。

最初に利用できるリファクタリングのオプションは、既存の .NET Framework アプリケーションを .NET に移行することです。 .NET に移行すると、Windows ではなく Linux で実行できるというメリットが得られます。これにより、総ライセンスコストが削減され、最新のフレームワークや、 .NET プログラミング言語の最新バージョンを利用できるようになります。

AWS SDK for .NET

[AWS SDK for .NET](#) は、 .NET 開発者に一貫性があり使い慣れたライブラリのセット AWS のサービスを提供することで、 の使用を簡素化します。 AWS SDK はクロスプラットフォームサポートを提供し、NuGet を使用して配布されます。開発者は AWS SDK を使用して .NET コードからクラウドサービスを簡単に呼び出し、アプリケーションのストレージ、キューイング、認証、および設定の要件を満たすことができます。

.NET Framework アプリケーションをモダナイズする

for .NET を使用して [.NET Transform NET Framework](#) から移行できます。これにより、コードファイルがスキャンされ、アプリケーションポートフォリオの移行ロードマップの計画に役立つレポートが作成されます。さらに、Porting Assistant for .NET は、互換性のない .NET Core API とパッケージを特定し、既知の代替品を見つけることで、移植のオーバーヘッドを削減するのに役立ちます。 .NET Framework アプリケーションを .NET に移行することで、ARM64-based Graviton プロセッサで実行し、price-to-performance比を向上させることができます。詳細については、GitHub の [Graviton 上の .NET](#)、および AWS Workshop Studio ドキュメントの [Graviton とコンテナ](#) を参照してください。

モノリスからマイクロサービスへ

多くの開発チームは、既存のモノリシックアプリケーションをマイクロサービスに再構築したいと考えています。マイクロサービスベースのアーキテクチャに移行することで、開発チームは開発の俊敏

性を高め、計算コストを削減し、サービスを個別にスケールし、デプロイ時間を短縮できます。コンポーネントを特定し、機能をグループ化することにより、開発チームは、.NET Framework モノリシックアプリケーションから機能を .NET サービスに段階的に抽出できます。

サーバーレスアプリケーションへのリファクタリング

[AWS Lambda](#) は、サーバーレスのイベント駆動型のコンピューティングサービスで、サーバーのプロビジョニングや管理を行わなくても、ほぼあらゆるタイプのアプリケーションやバックエンドサービスのコードを実行できるようになります。.NET と Lambda を使用して、既存のアプリケーションからロジックを抽出し、必要に応じて自動的にスケールするイベントベースのサーバーレスワークフローを作成できます。[Lambda の一般的な使用例](#)には、ファイル処理、分析、ウェブサイト、モバイルアプリケーションなど、さまざまなスケールニーズで数秒または数分間実行されるイベント駆動型ワークロードがあります。詳細については、Lambda ドキュメントの「[C# による Lambda 関数の構築](#)」を参照してください。

その他のリソース

- [AWS Toolkit for Azure DevOps](#) (AWS ドキュメント)
- [Jenkins を AWS CodeBuild および と統合して CI/CD パイプラインを設定する AWS CodeDeploy](#) (AWS DevOps ブログ)
- [.NET 用のデプロイツールについて AWS](#) (AWS GitHub)
- [.NET on AWS](#) (AWS ドキュメント)
- [aws/dotnet](#) (GitHub)

Windows フェイルオーバークラスターの移行

[Microsoft フェイルオーバークラスター](#) は、サーバーのグループであり、ストレージの大部分をサーバー間で共有しています。フェイルオーバークラスターを使用すると、アプリケーションやサービスの可用性を高めることができます。フェイルオーバークラスターを AWS クラウドに移行して、信頼性、パフォーマンス、TCO の削減というメリットを享受することもできます。

Windows フェイルオーバークラスターは、クラウドとオンプレミス環境では動作が異なります。クラウドにデプロイできるのはマルチサブネットクラスターのみであることに注意してください。オンプレミス環境とは異なり、Windows フェイルオーバークラスターの IP アドレスは、オペレーティングシステムレベルではなく Elastic Network Adapter (ENA) に割り当てられます。オンプレミス環境では、オペレーティングシステムが IP アドレスの割り当てを処理しますが、クラウドの IP アドレス割り当てはクラウドプロバイダー (AWS) が処理します。フェイルオーバークラスターリングはオ

ペレーティングシステムレベルの機能であるため、IP フェイルオーバーを制御することはできません。そのため、同じ IP アドレスをノード間でフェイルオーバーすることはできません。これを回避するには、クラスターがセカンダリ IP にフェイルオーバーするマルチサブネットクラスターを使用できます。セカンダリ IP は別のサブネットの ENA に割り当てられ、オンラインになることができます。詳細については、Microsoft ドキュメントの「[フェイルオーバークラスターリングネットワークの基本と基礎](#)」を参照してください。

Windows フェイルオーバークラスターを に移行する AWS のは複雑なプロセスですが、慎重な計画と実装により、ビジネス運用の中断を最小限に抑えて行うことができます。たとえば、フェイルオーバークラスターではアプリケーションごとに構成が異なるため、そのニーズを理解し、クラウドでどのように満たすことができるかを事前に確認することが不可欠です。このプロセスには、以下のステップが含まれます。

- すべてのクラスターノードが同じバージョンの Windows と必要な更新プログラムを実行していることを確認します。
- クラスタークォーラムの設定
- すべてのアプリケーションとデータがバックアップされ、移行中に復元できるようにする

評価

評価フェーズは、フェイルオーバークラスターを に移行するプロセスにおける重要なステップです AWS。このフェーズでは、現在の環境に関する情報を収集し、移行の実現可能性を判断し AWS、潜在的な課題やリスクを特定します。評価フェーズでは、次の手順を実行することをお勧めします。

- アプリケーションの準備状況の評価する – アプリケーションを変更 AWS せずに に移行できるかどうか、またはクラウドネイティブサービスを利用するためにアプリケーションを更新または書き換える必要があるかどうかを判断します。
- ネットワークとセキュリティの要件を評価する – ファイアウォール、ロードバランサー、VPN の設定など、ネットワークとセキュリティの要件を決定します。
- データ移行要件を評価する – データのサイズと場所 AWS、移行に必要な時間、データ転送コストなど、データの移行方法を決定します。オンプレミス環境では、JBOD、NAS、SAN などのさまざまなストレージテクノロジーが使用されている場合があります。それぞれが SAN ファイバーチャネル、iSCSI、SAS、SMB/NFS 共有など、さまざまなアクセス方法でアプリケーションにデータを提供できます。
- 潜在的なリスクと課題の特定 – ダウンタイム、互換性の問題、データ損失など、移行プロセスに影響を与える可能性のある潜在的なリスクや課題を特定します。

- コストの見積もり – Amazon EC2 インスタンス、ストレージ AWS、データ転送などの AWS のサービス 必要なコストなど、への移行にかかるコストを見積もります。
- 移行計画の作成 – 評価フェーズで収集された情報に基づいて、タイムライン、必要なリソース、移行に関連するステップを含む詳細な移行計画を作成します AWS。

現在の環境を評価する

ハードウェアとソフトウェアの構成を含む現在の環境を評価して、何を AWS に移行する必要があるかを判断します。アプリケーション、サーバー、データベース間の依存関係を特定します。

移行戦略を決定する

lift-and-shift アプローチや、クラウドネイティブサービスを活用するための環境の再設計など AWS、への移行のオプションを検討してください。

- 従来のフェイルオーバークラスターの移行 – Microsoft フェイルオーバークラスターを最初から手動で設定する場合は、「[Deploy SQL Server on Amazon EC2](#)」の手順に従います。共有ストレージは、フェイルオーバークラスターの移行において最も重要な考慮事項の 1 つです。Amazon EBS マルチアタッチでは SCSI-3 永続予約をサポートしていませんが、[Amazon FSx for Windows File Server](#) と [Amazon FSx for NetApp ONTAP](#) はどちらも共有ストレージオプションとして適切に機能します。最も一般的なユースケースの 1 つは、SQL Server クラスターの Always On フェイルオーバークラスターインスタンスと Amazon FSx for Windows File Server を使用することです。詳細については、AWS ストレージブログの「[Amazon FSx for Windows File Server を使用した Microsoft SQL Server の高可用性デプロイの簡素化](#)」の投稿を参照してください。次のステップは、ノードをクラウドに移行することです。これは、を使用して実現できます AWS Application Migration Service。詳細については、AWS ストレージブログの「[CloudEndure Migration AWS を使用した Microsoft Windows クラスターの への移行](#)」の投稿を参照してください。次に、アプリケーションのクラスター化されたロールを設定すると、高可用性を実現できます。
- ストレッチクラスターを使用して実質的なダウンタイムなしで移行 — ビジネスクリティカルなアプリケーションをクラウドに移行する必要があり、ダウンタイムが許されない場合には、ストレッチクラスターが適している可能性があります。[Microsoft ストレッチクラスター](#)では、サイト A とサイト B はネットワークを介して相互に通信する必要がありますが、それぞれ独自の共有ストレージを持つことができます。これを移行シナリオで活用できます。たとえば、ソース (オンプレミスか、別のプロバイダーのクラウドにあるかに関わらず) がサイト A であり、サイト B をデプロイする Amazon VPC とネットワーク接続しているとします。このアプローチでは、ソースのストレージ技術が機能するレプリケーション方法に関する制限要因を設けている可能性があるため、データレプリケーションのメカニズムが重要です。

- VMware オンプレミスにデプロイされたフェイルオーバークラスターを VMware Cloud on AWS に移行する AWS – VMware Cloud on AWS は、SCSI-3 永続予約をネイティブでサポートしています。これにより、AWS 上の VMware Cloud 上の仮想マシンディスク (VMDK) でフェイルオーバークラスターをホストできます。詳細については、VMware ドキュメントの「[Migrating SQL Server FCI cluster with shared disks to VMware Cloud on AWS](#)」を参照してください。

Note

2024 年 4 月 30 日現在、VMware Cloud on AWS は AWS またはそのチャネルパートナーによって再販されなくなりました。サービスは Broadcom を通じて引き続き提供されます。詳細については、AWS 担当者にお問い合わせください。

- Amazon EBS マルチアタッチボリュームを使用した SQL Server FCI の移行 – Amazon EBS マルチアタッチ予約と NVMe 予約を使用して、Windows Server フェイルオーバークラスターの共有ストレージとして Amazon EBS io2 ボリュームを持つ SQL Server フェイルオーバークラスターインスタンス (FCI) を作成できます。これらのボリュームは、インスタンスと同じアベイラビリティゾーンに限りアタッチできます。Amazon EBS io2 ボリュームを使用して Windows Server フェイルオーバークラスターをデプロイするには、SCSI 予約コマンドを NVMe 予約コマンドに変換する最新の Windows ドライバーが必要です。このアプローチを使用してオンプレミスの SQL Server FCI を単一のアベイラビリティゾーンで AWS に移行する方法の詳細については、AWS ブログ記事「[How to deploy a SQL Server failover cluster with Amazon EBS Multi-Attach on Windows Server](#)」を参照してください。

評価フェーズは、フェイルオーバークラスターへの移行を成功させるために重要です。AWS。時間をかけて情報を収集し、潜在的な課題を特定すれば、ダウンタイムを最小限に抑え、リスクを軽減し、AWS へのスムーズな移行を保証する包括的な移行計画を作成できます。

準備

フェイルオーバークラスターへの移行中 AWS、動員フェーズでは、クラスターへの移行の準備 AWS とテストを行い、クラスターが正常に機能することを確認します。準備フェーズには次のステップが含まれます。

1. ターゲット環境を準備する – このステップでは、フェイルオーバークラスターをホストするために必要な AWS リソースを作成します。これには、VPC、サブネット、セキュリティグループ、およびその他の必要なリソースの設定が含まれます。

2. ソース環境の準備 — このステップでは、既存のフェイルオーバークラスターを移行用に準備します。ネットワーク構成の変更、レプリケーションの構成、または必要なソフトウェアのインストールが行われる場合があります。
3. クラスターの検証 — ソース環境とターゲット環境の両方を準備すると、検証テストを実行してクラスターが正常に機能していることを確認できます。これには、クラスターがターゲット環境に正常にフェイルオーバーできることを確認するための一連のテストを実行する必要があります。
4. レプリケーションリンクの作成 — 検証テストの後、ソース環境とターゲット環境の間にレプリケーションリンクを作成できます。これにより、ソース環境に加えられたすべての変更がターゲット環境に確実に複製されます。
5. レプリケーションの監視 — レプリケーションリンクが確立されたら、レプリケーションプロセスを監視して、すべての変更が適切にレプリケートされていることを確認します。
6. クラスターのフェイルオーバー — レプリケーションが正常に動作していることを確認したら、ターゲット環境への最後のフェイルオーバーを実行します。ソース環境のクラスターサービスを停止し、ターゲット環境で起動します。
7. フェイルオーバーのテスト — フェイルオーバーが完了したら、テストを実行して、クラスター上で実行されているアプリケーションとサービスが新しい環境で正しく機能していることを確認します。

移行

Microsoft フェイルオーバークラスターの移行は複雑なプロセスになる場合があります、確実に成功させるには慎重な計画と実装が必要です。運用環境に変更を加える前に、既存の環境を徹底的に評価し、潜在的な問題を特定し、テストと検証を含む包括的な移行計画を策定することが不可欠です。移行フェーズでは、プロセスを注意深く監視し、問題や予期しない動作があればすぐに対処することが重要です。移行プロセスを円滑に進めるには、IT チーム、ビジネスユーザー、ベンダーを含むすべての利害関係者間のコミュニケーションとコラボレーションが不可欠です。

さらに、移行がフェイルオーバークラスター上で実行されているサードパーティのアプリケーションやサービスに及ぼす影響を考慮することも重要です。依存関係を特定し、それらのアプリケーションを徹底的にテストして、移行後も期待どおりに機能し続けることを確認します。移行フェーズのもう 1 つの重要な点は、移行プロセス中に予期しない問題や障害が発生した場合に備えて、ロールバック計画を立てることです。この計画には、運用環境への影響を最小限に抑えながら、移行を元に戻して元の環境を復元する手順が含まれているのが理想的です。

最後に、移行が完了し、新しい環境でフェイルオーバークラスターが正常に動作するようになったら、移行後の検証とテストを行い、すべてが意図したとおりに機能していることを確認することが重

要です。これには、パフォーマンスの監視、フェイルオーバー機能の検証、すべてのアプリケーションとサービスが適切に機能していることの確認が含まれます。

Microsoft のワークロードの監視

Microsoft のワークロードでは、通常、バックエンドで SQL Server を使用してデータを取得して保持します。クラウドへの移行の過程で、このようなソリューションのリホスト決定は、リフトアンドシフトというシンプルなアプローチで行われることがよくあります。このようなアプリケーションが Windows on Amazon EC2 プラットフォームでホストされている場合、ネイティブの Windows ベースのツールを使用して、これらのアプリケーションの状態をサーバーレベルで監視できます。ただし、ソリューションの一部としてデプロイされたさまざまなコンポーネントとサーバーを全体的に把握することは困難ですが、この課題は [Amazon CloudWatch Application Insights](#) で対処できます。

CloudWatch Application Insights は、AWS ワークロードのアプリケーションリソースのセットアップとモニタリングに役立つクラウドネイティブのモニタリングサービスです。企業のお客様は、さまざまなワークロードを扱うため、さまざまなソースからのテレメトリクスデータを相互に関連付けることができる監視サービスを必要としています。企業のお客様に対しては、CloudWatch Application Insights はリソース検出を自動化し、さまざまなリソースからアプリケーションを作成できるようにすることで、監視設定の複雑さを回避します。

評価

アプリケーションのパフォーマンスとバックエンドの状態を追跡することは、ほとんどの組織にとって不可欠です。その過程で、いつ、どこで異常が見つかったのか、なぜ発生したのかを知る必要があります。また、システムを監視してメンテナンスコストを削減する必要もあります。

CloudWatch は監視のニーズに対応し、CloudWatch Application Insights は CloudWatch メトリクス、アラーム、イベントを使用します。CloudWatch を使用して、多くの AWS リソースのメトリクス、テレメトリ、ログのモニタリングと管理を設定できます。[Amazon CloudWatch ServiceLens](#) では、アプリケーションの状態を監視するために必要なすべてのサービスを組み合わせて提供します。

準備

CloudWatch Application Insights には、アプリケーションに最適なテレメトリメトリクスとログを迅速かつ簡単に設定できる、クリック回数の少ないユーザーインターフェイスが用意されています。CloudWatch Application Insights は、特定のアプリケーションの問題の兆候を継続的に分析できるように、特定のワークロードに合わせてモニターを調整します。また、推奨ワークロードテレメトリの自動構成と分析も行います。例としては、.NET CLR、アプリケーション/Web サーバーテ

テクノロジーの 1 秒あたりのリクエスト数、.NET ガベージコレクションに関連する一般的な問題の特定、SQL Server のバックアップ失敗などがあります。

監視ソリューションの導入を検討する場合には、通常、CPU、メモリ、およびその他のしきい値要件を理解して構成する必要があります。ただし、CloudWatch Application Insights は、これらのリソースと関連メトリクスを自動的に検出します。CloudWatch Application Insights にアプリケーションを追加すると、リソースがスキャンされて、アプリケーションコンポーネントに対して推奨されるメトリクスとログが CloudWatch に設定されます。アプリケーションコンポーネントの例には、SQL Server バックエンドデータベースと Microsoft IIS/web 層があります。

選択したリソースグループに基づいて、CloudWatch Application Insights は各コンポーネントの監視を自動的に設定します。アカウントベースのアプリケーション監視の場合は、アカウント内で検出されるすべてのリソースが自動的に追加されます。CloudWatch Application Insights のリソース検出機能を利用することもできます。

CloudWatch Application Insights は、履歴データを使用してメトリクスのパターンを分析し、異常を検出します。また、アプリケーション、オペレーティングシステム、インフラストラクチャログからエラーや例外を継続的に検出します。これらの監視結果は、分類アルゴリズムと組み込みルールの組み合わせを使用して相互に関連付けられます。その後、自動作成されるダッシュボードに表示されるモニターリング結果と問題の重大度に関する情報に基づいて、対処するアクションの優先順位を決定できます。.NET および SQL アプリケーションスタックの一般的な問題 (アプリケーションのレイテンシー、SQL Server のバックアップの失敗、メモリリーク、大きくて無効な HTTP リクエスト、I/O オペレーションのキャンセルなど) については、CloudWatch Application Insights により根本原因を示唆する追加のインサイトと解決の手順が示されます。

組み込まれている [AWS Systems Manager OpsCenter](#) との統合により、関連する AWS Systems Manager Automation ドキュメントを実行して問題を解決できます。CloudWatch Application Insights は、各問題の重要度レベルを AWS Systems Manager OpsCenter に渡します。これにより、サポートチーム内のタスクの優先順位付けと割り当てがさらに容易になります。

移行

CloudWatch Application Insights は Windows on Amazon EC2 エコシステムの一部です。監視に CloudWatch Application Insights を使用することは、このサービスの重要な部分です。AWS へのワークロードの移行を開始後、CloudWatch Application Insights を利用して Microsoft ワークロードを監視できます。さらに、CloudWatch Application Insights は、SAP、Java、Oracle、MySQL、PostgreSQL、およびその他の AWS リソース (サーバーレスアプリケーションのサポートを含む) のサポートなど、Microsoft ワークロード以外のサポートを提

供します。CloudWatch Application Insights の使用を開始するには、CloudWatch ドキュメントの「[Getting set up](#)」を参照してください。

移行ツール、プログラム、トレーニング

このセクションでは、クラウド移行を支援するために利用できる AWS と AWS パートナーツール、クラウドへの移行と運用に必要なスキルをチームに提供するために利用できるトレーニングの機会、移行ジャーニーを加速し、移行コストを削減するために利用できる主要な移行プログラムの概要を説明します。

ツール

評価ツール

AWS 最適化とライセンスの評価

[AWS 最適化とライセンス評価 \(AWS OLA\)](#) を使用して移行とライセンス戦略を構築することをお勧めします。AWS OLA を使用して Windows AWS 環境を評価できます。この評価により、ライセンスコストを節約できる可能性を見極め、リソースをより効率的に運用する方法を見つけることができます。

AWS OLA は、新規および既存のお客様向けの無料プログラムです。AWS OLA を使用すると、実際のリソース使用率、サードパーティーのライセンス、アプリケーションの依存関係に基づいて、現在のオンプレミス環境とクラウド環境を評価および最適化できます。Enterprise [Strategy Group](#) と [Evolve Cloud Services](#) による 2022 年のサードパーティーの調査では、OLA AWS によって Microsoft SQL Server のライセンスコストが平均 45%、Windows Server が平均 77% 削減されたと計算されています。ライセンスコストは、これらのワークロードを実際に実行するコストの 3 倍に相当する AWS クラウドのため、コスト削減の可能性が TCO に大きな影響を与える可能性があります。

AWS OLA には、デプロイオプションをモデル化するレポートが用意されています。これらの結果は、[AWS OLA](#) が提供する柔軟なライセンスオプション全体で利用可能なコスト削減を検討するのに役立ちます。AWS OLA を Windows AWS 用 Migration Acceleration Program と組み合わせて使用して、クラウド移行中にサポートとリソースを取得することもできます。 [AWS](#)

移行前、移行中、移行後でも AWS OLA を使用できます。このツールベースのアプローチにより、実際の使用要件を判断することができます。AWS OLA は、各ワークロードの最小コストの EC2 インスタンスサイズとタイプに関するレコメンデーションを作成します。また、オンデマンドインスタンス、スポットインスタンス、Amazon EC2 Dedicated Hosts、Savings Plans、および環境固有の

その他のオプションを適切に組み合わせて見つけるのにも役立ちます。さらに、OLA AWS は移行計画、方向性のあるビジネスケース、ロードマップを提供します。

ライセンスの節約は TCO AWS の重要な部分であり、OLA は、既存のライセンス使用権限とワークロードに基づいて Bring Your Own License (BYOL) またはライセンス込みのレコメンデーションを提供することで、ライセンスコストを削減できます。AWS OLA は、アプリケーションの高いパフォーマンスを維持しながら、インスタンスが必要とするライセンス数を減らすように設定することで、ライセンスを最適化します。AWS OLA は、オンプレミスライセンスとクラウドでのライセンスの違いを理解するのに役立ちます。この知識を活かしてライセンス戦略を調整し、将来のコストをさらに削減することができます。

AWS OLA の範囲には、次のユースケースが含まれます。

- 方向性のあるビジネス ケース、EC2 インスタンスのコストを概説する推奨事項、実際のオンプレミスの使用状況とデータに基づく構成
- ホストレベルのライセンスに関する専有ホストモデリング
- SQL インスタンスの最適化と統合のための仮想 CPU (vCPU) の削減
- 業界平均に基づくオンプレミスの TCO の見積もり
- VMware Cloud on のモデリング AWS

Note

2024 年 4 月 30 日現在、VMware Cloud on AWS は AWS またはそのチャネルパートナーによって再販されなくなりました。サービスは Broadcom を通じて引き続き提供されます。詳細については、AWS 担当者にお問い合わせください。

- Microsoft のライセンス状況に基づく推奨事項 (ライセンスモビリティとライセンス削減の可能性について)
- T3 専有ホストのライセンス影響モデリング
- Amazon Relational Database Service (Amazon RDS) での SQL と Oracle のモデリング、エディションの最適化、Oracle Real Application Cluster (RAC) と Oracle Exadata の分析
- SQL 高可用性ライセンスへの影響に関するアクティブモデリングとパッシブモデリング
- モダナイズの評価

AWS は、内部の [Migration Evaluator](#) またはサードパーティーベンダー (または認定された OLA 移行パートナー) AWS の信頼できるツールを使用して、広範な検出を実行したり、既存のインベントリ

がある場合はエクスポートを安全にアップロードしたりします。使用するツールは、特定のニーズと要件によって異なります。は検出ツールの出力 AWS を使用し、サードパーティーのライセンスコンサルタントからの専門的な推奨事項と組み合わせて、信頼できる最適化された TCO を提供します。

詳細については、以下のリソースを参照してください。

- [AWS 最適化とライセンスの評価](#) (AWS ドキュメント)
- [Windows Workloads for AWS - AWS Online Tech Talks \(YouTube\) を最適化する](#) YouTube
- [Run Optimization and Licensing Assessment](#) (AWS ドキュメント)

AWS Migration Hub Strategy Recommendations

[AWS Migration Hub Strategy Recommendations](#) は、アプリケーションの実行可能なトランスフォーメーションパスに関する移行とモダナイズの戦略のレコメンデーションを提供することで、移行とモダナイズの取り組みを計画するのに役立ちます。Strategy Recommendations は、サーバーインベントリとランタイム環境の分析を行います。また、ソースコードやデータベース分析も実行できます。Strategy Recommendations は、この分析をビジネス目標および、指定されたアプリケーションやデータベースの変換に関する設定と組み合わせて、次のことを推奨します。

- 各アプリケーションにとって最も効果的な移行戦略
- 移行とモダナイズに使用できるツールまたはプログラム
- アプリケーションの非互換性と特定のオプションを解決するためのアンチパターン

Strategy Recommendations では、関連するデプロイ先、ツール、プログラムを使用して、リホスト、リプラットフォーム、およびリファクタリングを行うための移行とモダナイズ戦略が推奨されます。例えば、Strategy Recommendations が、AWS Application Migration Service を使用して Amazon EC2 にリホストするなどの簡単なオプションを推奨する場合があります。より最適化された推奨事項には、を使用したコンテナへのリプラットフォーム AWS App2Container や、.NET Core や PostgreSQL などのオープンソーステクノロジーへのリファクタリングなどがあります。

Strategy Recommendations を使用するには、「[Getting started with Strategy Recommendations](#)」の指示に従ってください。

移行検証ツールキット PowerShell モジュール

[移行検証ツールキット PowerShell モジュール](#) を使用して Microsoft ワークロードを検出し、AWS に移行することをお勧めします。このモジュールは、あらゆる Microsoft ワークロードに関連する一

一般的なタスクに対して複数のチェックと検証を実行することで機能します。移行検証ツールキット PowerShell モジュールを使用すると、組織は Microsoft ワークロードで実行されているアプリケーションやサービスの検出にかかる時間と労力を削減できます。このモジュールは、ワークロードの設定を特定するのにも役立ちます。これにより、その設定が AWS でサポートされているかどうかを確認できます。また、このモジュールでは、移行前、移行中、移行後に設定ミス回避できるよう、次のステップや緩和策に関する推奨事項も提供しています。

AWS クラウド準備状況評価

[AWS クラウド準備状況評価](#)を使用して、クラウドに移行するアイデアを、AWS プロフェッショナルサービスのベストプラクティスに従う詳細な計画に変換することをお勧めします。AWS クラウド準備状況評価を使用して、組織の規模に関係なく、クラウド導入とエンタープライズクラウド移行の効率的で効果的な計画を立てることができます。この 16 問のオンライン調査および評価レポートでは、ビジネス、人材、プロセス、プラットフォーム、運用、セキュリティなどの 6 つの観点から、クラウド移行の準備状況を詳しく説明しています。

評価の完了後、連絡先情報を入力すると、カスタマイズされたクラウド移行評価をダウンロードできます。この評価には、準備状況と改善のためにできることがまとめられています。概要レポートには、詳細なスコア情報とリソースを含むヒートマップとレーダーチャートが含まれており、準備状況スコアの向上に役立ちます。この重要なレポートは計画立案や利害関係者とのコミュニケーションに役立ちます。評価レポートのサンプルについては、「[AWS Cloud Adoption Readiness Assessment Report](#)」を参照してください。評価を受けるには、[AWS Cloud Adoption Readiness Assessment](#) にアクセスしてください。

移行ツール

AWS Migration Hub

[AWS Migration Hub](#) は、AWS への移行を評価、計画、追跡するためのサーバーとアプリケーションのインベントリデータを一元的に収集する場所です。Migration Hub は、移行後のアプリケーションのモダナイズを加速させるのにも役立ちます。Migration Hub のネットワークを視覚化すると、サーバーとその依存関係をすばやく特定し、サーバーの役割を特定し、サーバーをアプリケーションにグループ化することで、移行計画を迅速に進めることができます。ネットワークの視覚化を使用するには、[AWS Application Discovery Agent](#) をインストールし、データ収集を開始します。

AWS Migration Hub Orchestrator

[AWS Migration Hub Orchestrator](#) は、アプリケーションの移行を加速し、移行にかかる時間と労力を削減するのに役立ちます。定義済みのワークフローテンプレートを使用すると、移行ワークフローを

簡単に作成し、特定のニーズに合わせてワークフローをカスタマイズし、移行手順を自動化し、移行の進行状況を最初から最後まで 1 か所で追跡できます。Migration Hub Orchestrator は以下をサポートしています。

- SAP HANA データベースを使用した、SAP NetWeaver に基づくアプリケーションの移行
- あらゆるアプリケーションの Amazon EC2 へのリホスト
- SQL サーバーデータベースの Amazon EC2 へのリホスト
- SQL サーバーデータベースの Amazon RDS へのリプラットフォーム
- オープン仮想アプライアンス (OVA) または VMware 仮想マシンディスク (VMDK) の VM イメージの Amazon EC2 用 AMI へのインポート

AWS Migration Hub ダッシュボード

[Migration Hub ダッシュボード](#)には、リホストとリプラットフォームの移行に関する最新のステータスと指標が表示されます。このダッシュボードを使用すると、移行の進捗状況をすばやく把握し、問題を特定してトラブルシューティングできます。Migration Hub を使用すると、移行ツールで AWS リージョン サポートされている への移行のステータスを追跡できます。移行先のリージョンに関係なく、統合ツールを使用すると、移行ステータスが Migration Hub に表示されます。

AWS Application Migration Service

[AWS Application Migration Service](#) は、ネイティブに実行するソースサーバーの変換を自動化することで、時間のかかるエラーが発生しやすい手動プロセスを最小限に抑えます AWS。また、組み込みの最適化オプションとカスタム最適化オプションにより、アプリケーションのモダナイゼーションを簡素化します。Application Migration Service のユースケースには、次のようなものがあります。

- 物理サーバーまたは VMware vSphere、Microsoft Hyper-V、およびその他のオンプレミスインフラストラクチャ上で実行されている SAP、Oracle、SQL Server などのオンプレミスワークロード
- 他のパブリッククラウドから に実行されているクラウドベースのワークロード AWS

Application Migration Service を使用すると、コスト削減、可用性の向上、イノベーションの促進に役立つ 200 を超えるサービスにアクセスできます。さらに、これを使用して、Amazon EC2 ワークロードを AWS リージョン、アベイラビリティゾーン、またはアカウント間でより簡単に移動し、ビジネス、レジリエンス、コンプライアンスのニーズを満たすことができます。

あるいは、モダナイズ戦略として、カスタムモダナイゼーションアクションを適用するか、クロスリージョンディザスタリカバリ、CentOS コンバージョン、SUSE Linux サブスクリプションコン

バージョンなどのビルトインアクションを選択して、アプリケーションを最適化することもできます。

AWS Database Migration Service

[AWS Database Migration Service \(AWS DMS\)](#) は、データベースと分析のワークロードを、ダウンタイムを最小限に抑え、データ損失をゼロに、AWS 迅速かつ安全に移行するのに役立つマネージド移行およびレプリケーションサービスです。は、SQL Server を含む 20 以上のデータベースと分析エンジン間の移行 AWS DMS をサポートします。

AWS DMS では、マネージドデータベースモデルを使用して、シンプルな移行プロセスを通じてレガシーデータベースまたはオンプレミスデータベースからマネージドクラウドサービスに移行できます。これにより、開発者はイノベーションの時間を確保できます。また、AWS DMS を使用してライセンスコストから解放し、ビジネスの成長を加速し、専用データベースを使用して大規模なユースケースのイノベーションと構築を 10 分の 1 のコストで高速化することもできます。

AWS DMS を使用して、以下を実行することもできます。

- バックアップファイルを複製する
- ビジネスに不可欠なデータベースとデータストアを冗長化して、ダウンタイムとデータ損失を最小限に抑える
- データレイクを構築して、データストアからの変更データをリアルタイムで処理する
- データレイクを構築してデータマートを統合する
- データストアからの変更データをリアルタイムで処理する

移行パートナーツール

CloudBasix

[CloudBasix](#) は、クラウドネイティブなワークロード最適化およびデータ統合製品を提供しています。主力製品である [CLOUDBASIX for RDS SQL Server リードレプリカとディザスタリカバリ \(DR\)](#) を使用すると、次のことが可能になります。

- リージョン内リードレプリカ
- クロスリージョン DR
- クラウド間 Azure から AWS ディザスタリカバリへ
- AI 主導のデータレイクとデータハウス

• Amazon Redshift と Snowflake の統合

管理ツール

Amazon CloudWatch Application Insights

[Amazon CloudWatch Application Insights](#) は、アプリケーションと基盤となる AWS リソースのオペラビリティを容易にします。アプリケーションのリソースを監視する最適な条件を設定し、データを継続的に分析してアプリケーションの問題の徴候を検出できます。CloudWatch Application Insights は、Amazon SageMaker AI などの AWS テクノロジーを基盤とし、アプリケーションをモニタリングして検出した潜在的な問題を自動ダッシュボードに表示します。これにより、アプリケーションとインフラストラクチャで発生している問題をすばやく切り分けることができます。

CloudWatch Application Insights にアプリケーションを追加すると、アプリケーションのリソースがスキャンされて、アプリケーションコンポーネントに対して推奨されるメトリクスとログが CloudWatch に設定されます。アプリケーションコンポーネントの例には、SQL Server バックエンドデータベースと Microsoft IIS 層または Microsoft Web 層があります。CloudWatch Application Insights は、履歴データを使用してメトリクスのパターンを分析し、異常を検出します。また、アプリケーション、オペレーティングシステム、インフラストラクチャログからエラーや例外を継続的に検出します。これらの監視結果は、分類アルゴリズムと組み込みルールの組み合わせを使用して相互に関連付けられます。その後、CloudWatch Application Insights はダッシュボードを自動作成して、モニタリング結果と問題の重大度に関する情報を表示するため、対処するアクションの優先順位を決定するのに役立ちます。.NET および SQL アプリケーションスタックの一般的な問題 (アプリケーションのレイテンシー、SQL Server のバックアップの失敗、メモリリーク、大きな HTTP リクエスト、I/O オペレーションのキャンセルなど) については、根本原因を示唆する追加のインサイトと解決の手順が示されます。組み込まれている [AWS Systems Manager OpsCenter](#) との統合により、関連する Systems Manager Automation ドキュメントを実行して問題を解決できます。

AWS License Manager

[AWS License Manager](#) を使用すると、Microsoft、SAP、Oracle、IBM などのベンダーからのソフトウェアライセンスを、AWS およびオンプレミス環境全体で簡単に管理できます。License Manager を使用すると、ライセンスタイプを切り替えたり、既存のライセンスの検出、追跡、レポートを自動化したりすることで、ライセンス管理を効率化できます。また、Amazon EC2 専有ホストのコレクションを 1 つのエンティティとして管理し、割り当て、リリース、リカバリを自動化することで、Windows の BYOL エクスプレィエンスを簡素化できます。さらに、エンドユーザー AWS アカウント向けの全体のソフトウェア使用権限とワークロードの配布とアクティベーションを自動化することで、アカウント間でマーケットプレイスライセンスを処理できます。

AWS Backup

[AWS Backup](#) は費用対効果が高く、フルマネージド型のポリシーベースのサービスで、大規模なデータ保護を簡素化します。AWS Backup を使用して、バケット、ボリューム、データベース、ファイルシステムなどの主要なデータストアのクラウドネイティブバックアップを作成できます AWS のサービス。は、VMware ワークロード AWS Storage Gateway やボリュームなどのハイブリッド環境で実行されているアプリケーションのデータ保護管理を提供することで、データの保護を AWS Backup 一元化します。また、組織の、AWS アカウントリソース、および全体でバックアップアクティビティを設定、管理、管理するためのポリシーを一元管理することもできます AWS リージョン。

AWS Systems Manager フリートマネージャー

の一機能である [Fleet Manager](#) は AWS Systems Manager、AWS またはオンプレミスで実行されているノードをリモートで管理するための統合ユーザーインターフェイス (UI) エクスペリエンスです。Fleet Manager では、1 つのコンソールからサーバーフリート全体の正常性とパフォーマンスステータスを表示できます。個々のノードからデータを収集し、コンソールから一般的なトラブルシューティングと管理タスクを実行することもできます。これには、リモートデスクトッププロトコル (RDP) を使用した Windows インスタンスへの接続、フォルダとファイルのコンテンツの表示、Windows レジストリの管理、オペレーティングシステムのユーザー管理などが含まれます。Fleet Manager を使用すると、ノードフリートまたは Amazon Elastic Container Service (Amazon ECS) クラスターの管理を一元化できます。

プログラム

AWS Migration Acceleration プログラム

AWS [Migration Acceleration Program \(MAP\)](#) は、数千のエンタープライズ顧客をクラウドに移行した AWS 経験に基づく、包括的で実証済みのクラウド移行プログラムです。企業による移行は複雑で時間がかかる場合がありますが、MAP は成果重視の方法論により、クラウド移行とモダナイズの取り組みを加速させるのに役立ちます。

MAP は、コストを削減し、実装を自動化および高速化するツール、カスタマイズされたトレーニングアプローチとコンテンツ、AWS パートナーネットワークのパートナーからの専門知識、グローバルパートナーコミュニティ、AWS 投資を提供します。MAP では、移行目標の達成に役立つ、実証済みの 3 段階のフレームワークも使用しています。MAP を使用すると、リスクを軽減し、生産性を高め、運用レジリエンスを向上させ、移行の初期コストを相殺しながら、強力な AWS クラウド基盤を構築できます。また、クラウドのパフォーマンス、セキュリティ、信頼性も活用できます。

AWS Windows Migration Accelerator

[AWS Windows Migration Accelerator](#) は、を使用して Windows サーバーの移行を加速するときに AWS プロモーションクレジットを使用することで、移行のコストを削減します [AWS Application Migration Service](#)。AWS Windows Migration Accelerator のインセンティブは、合意済みの他の販売インセンティブやプロモーションプログラムに加えて適用できます。Application Migration Service を使用して、最低 15 台の Windows サーバーを含む 40 台のサーバーを 1 か月 AWS で移行する場合、2023 年 12 月 31 日まで、Windows サーバーあたり 200 USD のプロモーションクレジットを受け取る資格があります。1 か月に少なくとも 25 台の Windows サーバーを含む 80 台以上のサーバーを移行すると、Application Migration Service AWS を使用して移行する Windows サーバーごとに割引が 250 USD AWS のプロモーションクレジットに引き上げられます。移行されたサーバーは、以外の場所から移行し、移行後少なくとも 4 AWS 週間は継続的に実行する必要があります。

AWS Windows 用移行アクセラレーションプログラム

既存の [AWS MAP プログラムの延長である Windows 用移行促進プログラム \(MAP\)](#) は、ベストプラクティス AWS のサービス、ツール、インセンティブを使用して、組織が移行目標をより迅速に達成できるように設計されています。は、クラウドへの移行の不確実性、複雑さ、コストを削減するための 3 ステップのアプローチ AWS を使用します。AWS さらに、MAP は、Linux 上で稼働する SQL Server、Aurora、コンテナベースのサービス、Lambda などのクラウドソリューションを使用することで、Windows Server と SQL Server のワークロードの現行バージョンとレガシーバージョンをモダナイズし、コストを削減するのに役立ちます。クラウドネイティブまたはオープンソースのソリューションを利用すると、高額な商用ライセンスから解放されます。

AWS カウントダウン

[AWS Countdown](#) は、ショッピングホリデー、製品のローンチ、移行などの予定されたイベントの準備と実施の際に、アーキテクチャとスケーリングのガイダンスと運用サポートを提供します。これらのイベントの場合、AWS Countdown は運用準備状況を評価し、リスクを特定して軽減し、ユーザー側で AWS エキスパートと自信を持ってイベントを実装するのに役立ちます。このプログラムはエンタープライズサポートプランに含まれており、ビジネスサポートのお客様は追加料金で利用できます。

AWS のエキスパートは、以下を実行するのに役立つ規範的な段階的なアプローチを使用して、計画されたイベントのアーキテクチャと運用のガイダンスを提供するため、重点を置いたエンゲージメントを主導します。

- 成功基準と期待されるビジネス成果を理解する

- AWS 環境の準備状況を評価し、リスクの特定と軽減を支援し、計画を文書化する
- AWS エキスパートと一緒に自信を持ってイベントをホストする
- イベント後に結果を分析し、サービスを通常の運用レベルまで拡張することにより、次のイベントの計画に集中できる

トレーニング

自分のペースで進められる、インタラクティブな、クラスルーム形式のトレーニング

AWS では、移行ジャーニーをサポートするために、デジタルトレーニングとクラスルームトレーニングの両方を提供しています。AWS の専門家が作成した何百ものセルフペースデジタルトレーニングコースから学習を始めることができます。その後、[AWS Skill Builder](#) でインタラクティブなトレーニングを修了することで、実践的なスキルを習得できます。クラスルームトレーニングでは、質問をしたり、ソリューションを直接実行したり、高度な技術知識を持つ AWS 認定インストラクターからフィードバックを取得したりできます。詳細については、[AWS トレーニングと認定サービス](#)をご覧ください。

AWS パートナートレーニング

AWS パートナーは、EdX や Coursera などのトップオンライン学習プラットフォームで、AWS クラウド 基礎から機械学習まで幅広いトピックをカバーするセルフペースコースとしてデジタルトレーニングも提供しています。詳細については、[AWS パートナートレーニングと認定サービス](#)をご覧ください。役割とソリューションごとに認定を受けることができます。たとえば、役割にはクラウドプラクティショナー、ソリューションアーキテクト、開発者、システム運用管理者などがあります。ソリューションには、高度なネットワーク、データ分析、データベース、機械学習、セキュリティ、ストレージなどがあります。

での Microsoft ライセンス AWS

このセクションでは、Microsoft ライセンスの仕組みについて説明し AWS、に Microsoft ワークロードをデプロイするためのライセンスのベストプラクティスと戦略を提供します。また AWS、コストを最適化しながら、Microsoft のライセンス条項への準拠を維持するのに役立ちます。ライセンスは移行コストに影響するため、Microsoft ライセンスオプションと Bring Your Own License (BYOL) オプションが、利用可能なデプロイオプションに影響することがよくあります。そのため、移行プロセスを開始する前に、ライセンスの仕組みを理解することが重要です。

評価

Microsoft ワークロードの移行先を評価するときは AWS、ライセンス要件を考慮することが重要です。Microsoft ワークロードでは、[AWS 最適化とライセンス評価 \(AWS OLA\)](#) を活用してオンプレミスまたはクラウドワークロードを評価し、ワークロードを実行するための適切なサイズと最適化されたロードマップを構築することをお勧めします AWS。AWS OLA は、ワークロードに適した Amazon Elastic Compute Cloud (Amazon EC2) インスタンスに対して最適化された提案を行うだけでなく、Microsoft のライセンス位置も調べます。結果として、コンピューティングコストとライセンスコストを節約するための今後の最善の道筋が推奨されます。OLA AWS は新規および既存のお客様が利用でき、完全に資金が供給され、義務はありません。詳細については、[AWS OLA チーム](#)にお問い合わせください。

現時点では OLA AWS がオプションでない場合は、Microsoft ライセンスの仕組みを理解することが重要です AWS。BYOL を検討している場合は、貴社の Microsoft ライセンス購入担当者に Microsoft ライセンスステートメント (MLS) の更新版をリクエストすることをお勧めします。これを使用して、お持ちのライセンス内容、購入日、SA 数量 (該当する場合) を確認してください。MLS のサポートについては、AWS 担当者にお問い合わせください。担当者が Microsoft についてのスペシャリストをご紹介します。

Microsoft 製品ごとにライセンス要件が異なるため、デプロイした Microsoft 製品を明確に把握することが重要です。AWS には、ライセンスモビリティを使用する製品の Amazon EC2 の共有テナンシー/デフォルトテナンシーや、ライセンスモビリティを使用しない製品の専用オプションなど、さまざまな Microsoft 製品のニーズを満たすためのさまざまなオプションが用意されています。AWS にはライセンス込みオプションもあり、ライセンスのコストは Amazon EC2 コンピューティングコストに含まれます。移行時には、混合ライセンスモデルが役立ちます AWS。混合ライセンスモデルでは、共有テナンシー EC2 インスタンスを、すべてまたは一部のライセンス込みオプションと共に使用します。混合ライセンスモデルは、変動するワークロードや、安定的で予測可能なワークロード

に専用 Amazon EC2 オプションを使用する場合に最適です。特に Windows Server データセンターや SQL Server Enterprise の BYOL を選択できる場合です。

Microsoft のボリュームライセンスプログラムを通じて購入した製品の現在の Microsoft ライセンス条件の詳細については、Microsoft の「[製品条項](#)」サイトを参照してください。

ライセンス込みオプション

ライセンス込みとは、コンピューティングコストにライセンスのコストが含まれる Amazon EC2 インスタンスを指します。Microsoft サーバーワークロードの場合、AWS は現在、Windows Server ([Amazon EC2](#)、[Amazon EC2 Dedicated Hosts](#)、[Amazon EC2 Dedicated Instances](#)、[AWS Outposts](#)) および SQL Server Enterprise、Standard、Web Edition ([Amazon EC2](#)) を提供しています。これらのサーバーライセンスは、ライセンス込みの EC2 インスタンスの特典として、vCPU ごと 1 秒あたりの従量制料金モデルで提供されています。EC2 インスタンスの計画的な停止や、需要に応じたスケールアップまたはスケールダウンの場合は、インスタンスが実行されている間のみ課金されます。オンデマンド料金では長期的な契約がないため、今後のモダナイゼーション計画に最適です。

現在のバージョンとレガシーバージョンにはライセンスが含まれており、サポートされているすべてのバージョンで Amazon マシンイメージ (AMI) を利用できます。Windows Server 2008 や SQL Server 2012 などのサポート終了バージョンもライセンス込みで供与可能ですが、独自のメディアを持ち込む必要があります。

ライセンス込みのオプションでは、ソフトウェアアップグレード料金はかかりません。Microsoft が製品の新しいバージョンをリリースするとすぐに、その新しいバージョンが Amazon EC2 コンソールで利用できるようになります。その際に、現在のライセンス込み費用を超える追加費用はありません。最も重要な AWS は、ライセンス込み EC2 インスタンスのライセンスコンプライアンスを担当することです。ライセンス遵守は複雑で難しい場合があるため、これにより時間と労力を大幅に節約できます。

SQL Server のライセンス込みオプションでは、クライアントアクセスライセンス (CAL) を必要としないコアベースのライセンスが提供されます。CAL の数やライセンスの有無に関係なく、ライセンス込みの Windows Server EC2 インスタンスにはユーザー数に制限なくアクセスできます。Windows Server のライセンス込みの EC2 インスタンスには、管理目的専用の 2 つの Microsoft リモートデスクトップ接続も含まれています。追加の Microsoft リモートデスクトップ接続が必要な場合は、Microsoft からソフトウェアアシュアランス (SA) 付きのリモートデスクトップサービスユーザー CALs を購入し、ライセンスモビリティのメリット AWS を通じてに持ち込むことができます。

AWS には、ユーザーベースのライセンス込みオプションも用意されています。Visual Studio 2022 の Enterprise Edition と Professional Edition ([Amazon EC2](#) と [AWS Lambda](#)) と Office LTSC Professional Plus 2021 ([Amazon EC2](#)) は、ユーザー 1 人あたりの月額料金が課金されます。これには、各ユーザーの Microsoft リモートデスクトップ接続が含まれます。[Amazon WorkSpaces](#) では、Office Professional Plus 2016 または 2019 もアドオンとして提供しており、ユーザー 1 人あたりの月額料金が課金されます。

AWS には、Microsoft ワークロード用の以下のライセンス込みオプションが用意されています。

製品	利用可能な状況	使用可能なバージョン
Windows サーバー	Amazon EC2、Amazon EC2 ハードウェア専用インスタンス、Amazon EC2 専用ホスト、AWS Outposts	すべて*
SQL Server Enterprise	Amazon EC2、Amazon EC2 専用インスタンス、AWS Outposts	すべて*
SQL Server Standard	Amazon EC2、Amazon EC2 専用インスタンス、AWS Outposts	すべて*
SQL Server Web**	Amazon EC2、Amazon EC2 専用インスタンス、AWS Outposts	すべて*
Visual Studio Enterprise	Amazon EC2、AWS Lambda、Amazon WorkSpaces	2022
Visual Studio Professional	Amazon EC2、AWS Lambda、Amazon WorkSpaces	2022
Office Professional Plus	Amazon WorkSpaces	2019、2016

Office LTSC Professional Plus	Amazon EC2、Amazon WorkSpaces	2021
Visio LTSC Professional	Amazon WorkSpaces	2021
Visio LTSC Standard	Amazon WorkSpaces	2021
Project Professional	Amazon WorkSpaces	2021
Project Standard	Amazon WorkSpaces	2021
リモートデスクトップサービス SAL	Amazon EC2	—

*サポート対象外のバージョンやサポート対象バージョンには、独自のメディアが必要です。

**SQL Server Web Edition には、Microsoft のライセンス条項に基づいてユースケースが制限されています。SQL Server Web Edition のライセンスは、パブリックアクセスやインターネットアクセスが可能なウェブページ、ウェブサイト、ウェブアプリケーション、およびウェブサービスをサポートします。基幹業務アプリケーション (顧客関係管理、企業リソース管理、その他の類似アプリケーションなど) のサポートには使用できない可能性があります。

ライセンス込みのオプションは、変動するワークロードに最適です。例えば、ワークロードをほとんどの時間実行する必要がない場合や、ワークロードのスケールアップとスケールダウンを頻繁に行う必要がある場合です。

BYOL オプション

Bring Your Own License (BYOL) モデルを使用することは、AWS クラウドの効率性を活用しながら、オンプレミスソフトウェアへの既存の投資を活用するための優れた方法です。BYOL を使用すると、以前のソフトウェアバージョンと購入のライフサイクルを拡張し、ライセンス込み AWS によって提供されていない製品をデプロイできます。独自のライセンスを持ち込む場合は、必ず、独自のメディアも持ち込む必要があります。つまり、Amazon が提供する Amazon マシンイメージ (AMI) を使用するのではなく、独自のメディアを使用して独自の AMI を作成する必要があります。[VM Import/Export](#) ツールは無料で使用でき、独自の AMI を作成できます。または、[AWS Application Migration Service](#) を使用して独自のメディアと AMI を作成することもできます。

ソフトウェアアシュアランスによるライセンスモビリティのある Microsoft 製品

AWS は [認定モビリティパートナー](#) であるため、アクティブな SA の対象となるライセンスモビリティを備えた Microsoft 製品は、共有テナント環境または専用テナント環境で AWS に持ち込むことができます。SA によるライセンスモビリティの対象となる製品には、SQL Server、SharePoint Server、Exchange Server、Project Server、Skype for Business Server、BizTalk Server、リモートデスクトップサービスユーザー CAL、および System Center Server が含まれます。ライセンスモビリティ権がある Microsoft 製品は、2019 年 10 月 1 日に Microsoft が行った [ライセンス変更](#) の影響を受けません。そのため、ライセンスモビリティのある製品には購入日やバージョンに関する制限はありません。ライセンスにアクティブな SA がある AWS 限り、BYOL は の対象となります。例えば、SA がアクティブな SQL Server 2022 ライセンスは、SA が維持されている限り、共有テナンシー (デフォルト) の EC2 インスタンス (専有インスタンスは不要) に持ち込むことができます。

SA によるライセンスモビリティを使用する製品は、System Center Server を除き、仮想化されたオンプレミス環境内 AWS と同じ方法でライセンスされます。System Center Server ライセンスには、AWS クラウドに持ち込まれる際に特別なライセンス数の数え方が適用されます。System Center Server Datacenter Edition の 16 コアごとに、(サイズを問わず) 最大 10 個の EC2 インスタンスを管理できます。System Center Server Standard Edition の 16 コアごとに、(サイズを問わず) 最大 2 個の EC2 インスタンスを管理できます。SQL Server は、ライセンスモビリティを使用する最も一般的な製品です AWS。SQL Server のアクティブ SA 対象コアライセンスまたはサブスクリプションライセンス (クラウドソリューションプロバイダー (CSP) プログラムを通じて購入したライセンスを除く) は、共有テナンシー (デフォルト) EC2 インスタンスの vCPU ごとにライセンスされます。最小 Microsoft ライセンス要件は EC2 インスタンスごとに 4 つの vCPU です。アクティブな SA の対象となる SQL Server/CAL ライセンスは、EC2 インスタンスごとに 1 つのサーバーライセンスが付与されます。さらに、アクセス権を持つすべてのユーザーまたはデバイスには、対応する CAL が割り当てられている必要があります。SQL Server には、アクティブな SA とサブスクリプションによるパッシブフェイルオーバーの利点もあります。Amazon EC2 上のライセンスされたアクティブな SQL Server ごとに、Amazon EC2 上のセカンダリのパッシブ SQL Server インスタンスを 1 つ利用できます。パッシブインスタンスの SQL Server 部分のライセンスは必要ありません。詳細については、[Microsoft ウェブサイトの Microsoft SQL Server 2022 ライセンスガイド](#) (ダウンロード可能 PDF) を参照してください。AWS は [認定モビリティパートナー](#) (ダウンロード可能 PDF) です。 [ライセンスモビリティ](#) のある Microsoft 製品を に持ち込む場合は AWS、ライセンスモビリティ検証フォームに記入して Microsoft に送信する必要があります。このフォームは Microsoft Word の簡潔なドキュメントで、以下の情報を求められます。

- 貴社の名称と連絡先情報
- Microsoft 契約番号

- クラウドパートナー
- ライセンスモビリティを通じて持ち込まれる製品
- 持ち込むライセンス数

製品を AWS に持ち込んでから 10 日以内に、直接、または Microsoft 販売代理店を通じてフォームを Microsoft に提出する必要があります。検証プロセスの詳細については、Microsoft ドキュメントの「[ソフトウェアアシュアランスによるライセンスモビリティ](#)」を参照してください。ライセンスモビリティ検証フォームには、認定モビリティパートナーに関する情報を記載するセクションがあります。を E メールアドレス `microsoft@amazon.com` として、Amazon Web Services をパートナー名として、をパートナーウェブサイト `aws.amazon.com` として使用できます。詳細なガイドランスについては、Microsoft ドキュメントの Microsoft の「[お客様向け検証ガイド](#)」(ダウンロード可能 PDF) を参照してください。ライセンスモビリティ検証フォームのコピーをダウンロードするには、Microsoft ドキュメントの「[ライセンスに関するリソースとドキュメント](#)」を参照してください。

Note

Microsoft が提供する Flexible Virtualization Program は、Microsoft によってリストされたプロバイダー* クラウドに AWS 指定されている AWS ため、では使用できません。Microsoft は、2019 年 10 月 1 日の[ライセンス変更](#)の一環として、Alibaba、Amazon、Google Cloud を[リストに記載されたプロバイダー](#)に指定しました。2019 年 10 月 1 日以降、SA およびライセンスモビリティの権利なしで購入したオンプレミスライセンスは、リストに記載されたプロバイダーが提供するホスト型クラウドサービスにデプロイできなくなります。

ライセンスモビリティのない Microsoft 製品

Windows Server、Visual Studio、Microsoft Developer Network (MSDN)、Windows デスクトップオペレーティングシステム、Microsoft Office、および Microsoft 365 アプリケーション (旧 Office 365) には、ライセンスに SA が付与されている場合やアクティブなサブスクリプションライセンスであっても、Microsoft 製品条項でライセンスモビリティ権が付与されていません。そのため、これらの製品のライセンスを持ち込むには、Amazon EC2 専有ホスト、Amazon EC2 ハードウェア専有インスタンス、VMware Cloud on AWS、AWS Outposts の専有ホストなどの専有インフラストラクチャが必要です。AWS への BYOL の対象となるには、他の特定の要件にも従う必要があります。これらの要件は、リストに記載されたプロバイダーのクラウドにデプロイする際のライセンスモビリティのない製品に関するライセンス条項を、Microsoft が変更した結果です (2019 年 10 月 1 日施行)。詳細に

については、Microsoft ドキュメントの「[専用ホストクラウドサービスに関するマイクロソフトライセンス条項の改定](#)」を参照してください。

BYOL の対象となるには AWS、ライセンスモビリティのない製品のライセンスが Microsoft の次の要件を満たしている必要があります。

- ライセンスは (サブスクリプションではなく) 永久使用権として購入されている必要があります。
- ライセンスの購入日が 2019 年 10 月 1 日より前であるか、2019 年 10 月 1 日より前に開始された Microsoft Enterprise 契約期間内にライセンスを購入している必要があります。
- デプロイされるバージョンは 2019 年 10 月 1 日より前に一般公開されている必要があります。
- 製品は専用インフラストラクチャにデプロイする必要があります。

ライセンスモビリティのない製品のサブスクリプションライセンスは、2019 年 10 月 1 日以降に購入または更新されると BYOL が失われます。

Note

注: ライセンスモビリティのない製品では、ライセンスが上記の要件を満たしている限り AWS、での BYOL のアクティブな SA は必要ありません。

ライセンスは複雑な場合があるため、[Amazon Web Services と Microsoft のよくある質問サイト](#)を参照して、ライセンスが BYOL to AWS オプションの対象であるかどうかを確認します。よくある質問で必要な情報が見つからない場合、または Microsoft ワークロードの移行先が不明な場合は AWS、にお問い合わせください Microsoft@Amazon.com。AWS には、必要なすべての情報を確実に入手できるように、Microsoft ワークロードとライセンスのスペシャリストが用意されています。

Note

Windows Server BYOL は物理コアによってライセンスされる必要があるため、Windows Server BYOL には専用ホストテナンシー (Amazon EC2 Dedicated Hosts や Dedicated Hosts on など AWS Outposts) が必要です。

サービスプロバイダーライセンス契約 (SPLA) の BYOL

サービスプロバイダーライセンス契約 (SPLA) プログラムは、2019 年 10 月 1 日に Microsoft が行った [ライセンス変更](#)の影響を受けませんでした。その結果、独自の SPLA ライセンスをお持ちの顧客

様は、購入日やバージョンの制限なしに、新規の Windows Server ライセンスを SPLA を通じて持ち込むことができます。SPLA を通じてライセンスされるコアベースまたはプロセッサベースの製品にはすべて Amazon EC2 専有ホストが必要で、ユーザーベースのサブスクリバークラウドアクセスライセンス (SAL) を共有テナンシー (デフォルト) の EC2 インスタンスに持ち込むことができます。これは、SPLA のユーザーベースの SAL が [サービスプロバイダー使用権 \(SPUR\)](#) のデータセンタープロバイダー (DCP) の対象となるためです。

Note

Microsoft は、2025 年 9 月 30 日以降、AWS または他の出品プロバイダークラウドで SPLA BYOL を許可しないことを [発表しました](#)。

Amazon EC2 専有ホスト

[Amazon EC2 専有ホスト](#) の主な機能には以下があります。

- 物理ソケットとコアを可視化できる、事前設定済みの Amazon EC2 Nitro および Xen ハイパーバイザー
- 同じ専有ホストでサポートされた同じファミリー内の複数のインスタンスサイズ (サポートされているインスタンスタイプの最新セットについては、Amazon EC2 ドキュメントの「[Amazon EC2 専有ホスト](#)」を参照してください)
- 自動管理、自動スケーリング、インスタンス配置制御
- 複数の間でホストを共有する機能 AWS アカウント
- ライセンスの使用状況と管理を追跡するための [AWS License Manager](#) との統合
- インスタンスとホストとの親和性を維持する機能
- ホスト復旧の自動化
- を使用した継続的なモニタリング AWS Config

Windows Server BYOL には専有インフラストラクチャ数と物理コア数が必要なため、Amazon EC2 専有ホストは次のように役立つ優れたオプションです。

- 大幅なコスト削減を実現
- SA またはライセンスモビリティに関係なく AWS、任意の Microsoft アプリケーションをに持ち込むことができます (2019 年 10 月 1 日の購入およびバージョン要件が適用されます)。

- Windows Server Datacenter Edition と SQL Server Enterprise Edition の物理コアライセンスを最大限に活用
- EC2 インスタンスごとではなく、ホストごとに課金 (つまり、専用ホストを使用すると、追加のコンピューティング料金を発生させることなく、ホストで利用可能な最大数のインスタンスを使用できます)

BYOL 対象の Windows Server ライセンスを Amazon EC2 専用ホストに持ち込むと、ホストのすべての物理コア (vCPUs ではない) のライセンスを取得できます。例えば、R5 Amazon EC2 専用ホストには 48 個の物理コアがあります。48 コアの Windows Server Datacenter Edition を R5 Amazon EC2 専用ホストに持ち込むと、技術的に可能な限りの多くの EC2 インスタンスをホストにデプロイできます。48 コアの Windows Server Standard Edition を持ち込むと、ホスト上に任意のサイズの EC2 インスタンスを最大 2 つ持ち込むことができます。

Windows Server Standard Edition のライセンスを積み上げると、同じホストで追加の EC2 インスタンスを作成できます。2 回目のライセンスでは、ホストのすべての物理コアで 2 つの EC2 インスタンスを追加できます (以下同様)。SQL Server Enterprise を物理コアでライセンスする場合は、ホストのすべての物理コアのライセンスを取得する必要があります。これにより、ライセンスされた物理コアの数と同じ数の SQL Server 用の EC2 インスタンスをホストにデプロイできます。例えば、48 コアの SQL Server Enterprise でライセンスされている R5 Amazon EC2 専用ホストでは、SQL Server を実行する最大 48 の EC2 インスタンスをそのホストにデプロイできます。BYOL 対象の Windows Server Datacenter ライセンスと SQL Server Enterprise ライセンスを持ち込み、ホストの物理コアの合計に対してライセンスを取得すると、同じ数とサイズの EC2 インスタンスについて、ライセンス込みよりも大幅にコストを節約できます。これは、ワークロードがホストの大部分を占め、ほとんどの時間稼働していることを前提としています。例えば、12 個の R5.2xlarge EC2 インスタンスをライセンス込みの Windows Server と SQL Server Enterprise BYOL を含む共有テナンシーインスタンスにデプロイすることもできます。ライセンスには合計 96 コアの SQL Server Enterprise が必要です。しかし、R5 Amazon EC2 専用ホスト (同じ 12 個の R5.2xlarge EC2 インスタンスに対応) をデプロイする場合は、48 コアの Windows Server Datacenter と 48 コアの SQL Server Enterprise BYOL 対象ライセンスを持ち込むことができます。Windows Server のライセンス込みのコストを節約できるだけでなく、持ち込みが必要な SQL Server Enterprise コアライセンスの数は半分だけで済みます。

Amazon EC2 専用ホストの BYOL は、ホストを 70% 以上使用でき、ほとんどの時間ワークロードが実行されている、安定した予測可能なワークロードに最適です。での Microsoft ライセンスの詳細については AWS、Microsoft ドキュメントの YouTube および Amazon Web Services での Microsoft [ライセンス AWS](#) および「Microsoft のよくある質問」を参照してください。 <https://aws.amazon.com/windows/faq/>

準備

AWS License Manager

Microsoft のライセンスに関する検討の準備段階の一環として、AWS のワークロードに割り当てる予定のライセンスを [AWS License Manager](#) に入力することをお勧めします。License Manager は、オンプレミスや他のクラウドのワークロード AWS だけでなく、Microsoft、Oracle、IBM、SAP などのベンダーからのソフトウェアライセンスを簡単に管理できる無料のツールです。

License Manager AWS に持ち込む Microsoft ライセンスを入力すると、次のことに役立ちます。

- ソフトウェアライセンスの使用状況の可視化や制御がしやすくなり、誤用を未然に防ぐことができます。
- ライセンスの追跡と管理の方法を含め、ライセンスを最大限に活用することでコストが低減します。
- ライセンス使用量の制限、新規起動のブロック、その他の制御の使用により、コンプライアンス違反のリスクが軽減します。
- ホストリソースグループを使用してホストの配置、リリース、復旧を自動化することで生産性が向上します。

License Manager の詳細については、License Manager ドキュメントの「[Working with AWS License Manager](#)」を参照してください。

ライセンスに関する考慮事項

ワークロードに現在割り当てられているライセンスに基づいて移行を計画することを移行前に検討してください。たとえば、複数のオンプレミスホストを に持ち込む場合は AWS、複数の異なるホストにまたがるワークロードをグループ化するのではなく、ホストごとに移行することを検討してください。これは、オンプレミスホストの使用を停止すると、そのホストに関連付けられているライセンスが AWS で使用できるように解放されるためです。また、移行中は Windows Server または SQL Server のライセンス込みのインスタンスを使用し、移行の完了後に BYOL オプションに切り替えることもできます。ただし、このオプションでは、最初から独自のメディアと AMI を使用する必要があります (ライセンス込みオプションでも)。で使用できる [ライセンス変換機能](#) AWS License Manager では、EC2 インスタンスが元のメディアと AMIs から作成された場合にのみ、ライセンス込みから BYOL に切り替えることができます。

移行

Microsoft ワークロードを にデプロイしてから 10 日以内に AWS、[ライセンスモビリティを使用するライセンスについて、必ずライセンスモビリティ検証フォーム](#)を Microsoft に送信してください AWS。このフォームは、移行のさまざまな段階に応じて複数回提出できます。このフォームでは以下の情報を求められます。

- 貴社の名称と連絡先情報
- Microsoft 契約番号
- クラウドパートナー
- ライセンスモビリティを通じて持ち込まれる製品
- 持ち込むライセンス数

検証プロセスの詳細については、Microsoft ドキュメントの「[ソフトウェアアシュアランスによるライセンスモビリティ](#)」を参照してください。詳細なガイドンスについては、Microsoft ドキュメントの Microsoft の「[お客様向け検証ガイド](#)」(ダウンロード可能 PDF) を参照してください。ライセンスモビリティ検証フォームのコピーをダウンロードするには、Microsoft ドキュメントの「[ライセンスに関するリソースとドキュメント](#)」を参照してください。

AWS パートナー

AWS コンピテンシーパートナーを関与させるメリット

Microsoft ワークロードをクラウドに効率的に移行するには、慎重な計画と効率的な実装が必要です。主なステップには、範囲の設定、クラウド移行ビジネスケースの作成、エグゼクティブスポンサーの合意形成、クラウド財務管理 KPI の設定、クラウド向けのセンターオブエクセレンスの構築、移行サービスの検証、大規模移行用自動化ツールのデプロイ、クラウドへのセキュリティ戦略の拡張が含まれます。

検証済みの [AWS コンピテンシーパートナー](#) を起用して、移行ジャーニーを通じて組織を主導することをお勧めします。AWS パートナーは、戦略的エキスパートであり、移行ジャーニーのすべてのフェーズを通じて前述の主要なステップとビジネス目標に対処するのに役立つ経験豊富なビルダーです。AWS パートナーコミュニティには、お客様のクラウドジャーニーをサポートし、イノベーションの推進、俊敏性の向上、コストの削減に集中できるように支援する 100,000 を超えるパートナーが 150 以上の国から参加しています。

計画立案

AWS パートナーは、準備状況評価を実行し、移行計画を作成し、移行ツールを導入してクラウドへの移行を加速できます。さらに、お客様のスキルギャップを埋め、コスト最適化戦略を推奨し、AWS への移行助成金を受けるための排他的な移行インセンティブの対象となるよう支援します。

コストの最適化

急速に進化する今日のテクノロジー環境では、デジタルトランスフォーメーションの過程で多くの組織がコスト面で深刻な課題に直面しています。よくある懸念事項の 1 つは、クラウドが高価すぎて、クラウドが提供するビジネス上の大きなメリットが見えづらいと認識されていることです。さらに、テクノロジースタックのモダナイズにかかるコストが、財務上の問題を引き起こす可能性があります。

[AWS Microsoft ワークロードコンピテンシーパートナー](#) と連携すると、Microsoft ワークロードを AWS にデプロイするための最も適切な資格を持つ AWS パートナーとつながることができます。これらのパートナーは、技術的能力を検証し、お客様が Microsoft ワークロードをに移行、管理、またはデプロイするのを支援することに成功しています AWS。これらのパートナーでサポートされるワークロードには、Windows Server、Microsoft SQL Server、Windows File Server、SharePoint、および .NET アプリケーションなどがあります。

AWS パートナーは AWS ベストプラクティスを使用して、安全性、可用性、信頼性、パフォーマンス、コスト最適化のアーキテクチャを構築します。パートナーは、 が提供する資金を最大限に活用 AWS してコストを最適化し、専門知識を活用して価値実現までの時間を短縮することもできます。最後に、AWS パートナーは [AWS Windows 用 Migration Acceleration Program](#) を活用して、移行コストを相殺できます AWS。

時間の節約

Note

2024 年 4 月 30 日現在、VMware Cloud on AWS は AWS またはそのチャンネルパートナーによって再販されなくなりました。サービスは Broadcom を通じて引き続き提供されます。詳細については、AWS 担当者にお問い合わせください。

多くの企業はオンプレミスインフラストラクチャに重点的に投資しています。組織がオンプレミスインフラストラクチャを管理するために VMware ソフトウェアに多額の投資を行ったため、AWS でインフラストラクチャを管理するのも同じオンプレミスツールを使用したいと考えることはあり得ます。それ自体をクラウドに移行するのは難しいが、移行済みのワークロードに依存している特殊なワークロードやインフラストラクチャがある場合もあります。また、インフラストラクチャの一部が従来のオンプレミスデータセンターにあり、他の部分がクラウドにデプロイされているハイブリッドインフラストラクチャのパターンもあります。

時間が最重要事項である場合は、スキルを持つ人材、洗練されたプロセス、技術力により、大規模な移行を幅広く提供してきた実績を持つ [AWS 移行コンピテンシーパートナー](#) と連携することをお勧めします。サポートされているワークロードカテゴリには、Windows、SAP、Oracle、VMware on AWS、データベース、分析、ストレージ、モノのインターネット (IoT)、機械学習、Software as a Service などがあります。

AWS パートナーは、への移行 AWS は all-or-nothing の動きを意味し、現在の投資を排除するものではないことを理解します。インフラストラクチャの最適化と合理化に長けており、オンプレミスでどのパートが最適か、どのパートがクラウドに最適かを最適化します。AWS には、Amazon Virtual Private Cloud (Amazon VPC) や AWS Direct Connect などのハイブリッドクラウドソリューションが幅広く用意されています AWS Storage Gateway。

AWS パートナーは、数千のエンタープライズ顧客をクラウドに移行した経験に基づいて AWS、包括的で実証済みのクラウド移行プログラムである [AWS 移行促進プログラム \(MAP\)](#) の対象となる顧客を認定できます。MAP は、包括的なツール、サービス、ガイダンス、トレーニングなどのインセン

タイプを通じて、特殊なワークロードをサポートします。特殊なワークロードサポートは、メインフレーム、Windows、ストレージ、VMware Cloud on、SAP AWS、データベース、Amazon Connect で利用できます。

セキュリティの強化

データのプライバシーとセキュリティが懸念される場合があります。さらに、データ処理のプラクティスが、海外データ合法的使用明確化法 (CLOUD) と一般データ保護規則 (GDPR) に準拠しているという保証が必要な場合もあります。特定のワークロードやユースケース向けにセキュリティに重点を置いたソリューションを提供するために、セキュリティエキスパートチームを提供できる [AWS セキュリティコンピテンシーパートナー](#) を採用することをお勧めします。AWS パートナーソリューションにより、ワークロードの自動化と俊敏性、スケーリングが可能になります。

公開時に、は PCI-DSS、HIPAA/HITECH、FedRAMP、GDPR、FIPS 140-2、NIST 800-171 など、幅広いセキュリティ標準とコンプライアンス認定 AWS をサポートしています。世界中のほとんどの規制機関のコンプライアンス要件を満たすようサポートします。

ヘルスケア、銀行、法律、製薬など、セキュリティの影響を受けやすい業種の一部の民間組織や公共部門組織は、セキュリティ体制の改善 AWS を信頼しています。中小企業、大企業、公共部門を問わず、ビジネスを前進させるのに役立つ適切なスキルと経験を持つ AWS パートナーがいます。AWS パートナースペシャリストは、ビジネスニーズに合った適切なクラウドパートナーを見つけて接続するのに役立ちます。詳細については、[AWS パートナースペシャリスト](#) にお問い合わせください。世界中のお客様が を使用してクラウドの導入を加速し、イノベーションを促進する方法については AWS Partner Network、[「AWS パートナーによるカスタマーサクセス」](#) を参照してください。

次のステップ

次の手順を実行することをお勧めします。

1. 具体的な移行とモダナイズのシナリオについて学びます。詳細については、「[Microsoft SQL Server データベースの AWS クラウド への移行](#)」、「[RDBMS から Amazon DynamoDB への移行によるアプリケーションの最新化](#)」、「[.NET アプリケーションを最新化するためのアプローチの選択](#)」を参照してください。
2. 大規模移行が組織に与える影響について学びます。大規模移行は、テクノロジーの変換であるだけでなく、組織のロール、プロセス、優先順位の変更も伴います。詳細については、「[AWS の大規模移行のための戦略とベストプラクティス](#)」を参照してください。
3. 「[AWS for Microsoft Workloads Self-Study Guide](#)」を確認してください。
4. [Migrating Microsoft Workloads to AWS Hands-on Workshop](#) を完了します。

リソース

Microsoft から AWS 移行へのガイドライン

- [Microsoft ワークロードの への移行 AWS: 自己学習ガイド](#)
- [Microsoft ワークロードの への移行 AWS: 実践ラボ](#)
- [Microsoft SQL Server データベースの への移行 AWS クラウド](#)
- [Modernizing your application by migrating from an RDBMS to Amazon DynamoDB](#)
- [Choosing an approach for modernizing .NET applications](#)
- [AWS 大規模な移行の戦略とベストプラクティス](#)

一般的なガイドライン

- [の Windows AWS](#)
- [AWS 大規模な移行の戦略とベストプラクティス](#)
- [AWS ドキュメント](#)

動画

- [AWS re:Invent 2020: Microsoft ワークロードの への移行 AWS](#)
- [仮想ワークショップで Windows ワークロードをリホスト AWS Application Migration Service AWS する](#)

AWS ブログ投稿

- [を使用してオンプレミスワークロードを移行する方法 AWS Application Migration Service](#)
- [を使用して Windows ワークロードを移行する理由 AWS \(およびその方法\)](#)

寄稿者

オーサリング

- Dror Helper、AWS シニアソリューションアーキテクト
- Christine Megit、AWS シニアスペシャリスト
- AWS シニアソリューションアーキテクト、ダニエル・マルドナド
- Mani Pachnanda、AWS シニアソリューションアーキテクト
- Siddharth Mehta、AWS Principal Solutions Architect
- Reut Almog Talmim、AWS ソリューションアーキテクト
- Rob Higareda、AWS プリンシパルソリューションアーキテクト
- Saleha Haider、AWS シニアデリバリーコンサルタント
- Siavash Irani、AWS プリンシパルソリューションアーキテクト
- Yogi Barot、AWS テクニカルリーダー

レビューアー

- Jake Ignatius、AWS ソリューションマネージャー

テクニカルライター

- " AbouHarb, AWS Senior Technical Writer

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	SQL Server データベースの使用に関する情報を追加 AWS Transform し、移行に関する情報を更新しました。使用できなくなった AWS のサービスた を削除しました。	2026 年 3 月 13 日
更新	「 AWS での Microsoft ライセンシング 」セクションに新しいライセンス込みのオプションを追加しました。	2025 年 2 月 27 日
更新	Amazon EBS マルチアタッチに関する情報を「 Windows フェイルオーバークラスタの移行 」セクションに追加しました。	2024 年 4 月 1 日
更新	移行検証ツールキット PowerShell モジュールへのリンクを追加しました。 「 Windows フェイルオーバークラスタの移行 」セクションの「チュートリアル: Amazon EC2 で Windows HPC クラスタを設定する」の使い方を明確にしました。	2023 年 12 月 14 日

更新	「Windows フェイルオーバー クラスターの移行」 セクションを更新しました。	2023 年 12 月 8 日
更新	「AWSでの Microsoft ライセンシング」 ページの「Amazon EC2 専有ホスト」セクションで、専有ホストでサポートされているインスタンスタイプのリストを更新しました	2023 年 11 月 16 日
更新	サポートされているインスタンスファミリーの全リストを、 「AWSでの Microsoft ライセンシング」 ページの「Amazon EC2 専有ホスト」セクションに追加しました	2023 年 7 月 31 日
更新	「SQL サーバーの移行」 ページの「リプラットフォーム」セクションに BYOM ガイダンスを追加しました	2023 年 6 月 23 日
初版発行	—	2023 年 6 月 9 日

AWS 規範ガイドの用語集

以下は、AWS 規範ガイドが提供する戦略、ガイド、パターンで一般的に使用される用語です。エントリを提案するには、用語集の最後のフィードバックの提供リンクを使用します。

数字

7 Rs

アプリケーションをクラウドに移行するための 7 つの一般的な移行戦略。これらの戦略は、ガートナーが 2011 年に特定した 5 Rs に基づいて構築され、以下で構成されています。

- リファクタリング/アーキテクチャの再設計 — クラウドネイティブ特徴を最大限に活用して、俊敏性、パフォーマンス、スケーラビリティを向上させ、アプリケーションを移動させ、アーキテクチャを変更します。これには、通常、オペレーティングシステムとデータベースの移植が含まれます。例: オンプレミスの Oracle データベースを Amazon Aurora PostgreSQL 互換エディションに移行する。
- リプラットフォーム (リフトアンドリシェイプ) — アプリケーションをクラウドに移行し、クラウド機能を活用するための最適化レベルを導入します。例: お客様のオンプレミスの Oracle データベースを AWS クラウドの Oracle 用の Amazon Relational Database Service (Amazon RDS) に移行する。
- 再購入 (ドロップアンドショップ) — 通常、従来のライセンスから SaaS モデルに移行して、別の製品に切り替えます。例: 顧客関係管理 (CRM) システムを Salesforce.com に移行する。
- リホスト (リフトアンドシフト) — クラウド機能を活用するための変更を加えずに、アプリケーションをクラウドに移行します。例: お客様のオンプレミスの Oracle データベースを AWS クラウドの EC2 インスタンス上の Oracle に移行する。
- 再配置 (ハイパーバイザーレベルのリフトアンドシフト) — 新しいハードウェアを購入したり、アプリケーションを書き換えたり、既存の運用を変更したりすることなく、インフラストラクチャをクラウドに移行できます。オンプレミスプラットフォームから同じプラットフォームのクラウドサービスにサーバーを移行します。例: Microsoft Hyper-V アプリケーションをに移行します AWS。
- 保持 (再アクセス) — アプリケーションをお客様のソース環境で保持します。これには、主要なリファクタリングを必要とするアプリケーションや、お客様がその作業を後日まで延期したいアプリケーション、およびそれらを移行するためのビジネス上の正当性がないため、お客様が保持するレガシーアプリケーションなどがあります。
- 廃止 — お客様のソース環境で不要になったアプリケーションを停止または削除します。

A

ABAC

[「属性ベースのアクセス制御」](#) をご覧ください。

抽象化されたサービス

[「マネージドユーザー」](#) をご覧ください。

ACID

[「原子性、一貫性、分離性、耐久性 \(ACID\)」](#) をご覧ください。

アクティブ/アクティブ移行

(双方向レプリケーションツールまたは二重書き込み操作を使用して) ソースデータベースとターゲットデータベースを同期させ、移行中に両方のデータベースが接続アプリケーションからのトランザクションを処理するデータベース移行方法。この方法では、1 回限りのカットオーバーの必要がなく、管理された小規模なバッチで移行できます。[アクティブ/パッシブ移行](#) よりも柔軟な方法ですが、さらに多くの作業が必要となります。

アクティブ/パッシブ移行

ソースデータベースとターゲットデータベースを同期させながら、データがターゲットデータベースにレプリケートされている間、接続しているアプリケーションからのトランザクションをソースデータベースのみで処理するデータベース移行方法。移行中、ターゲットデータベースはトランザクションを受け付けません。

集計関数

複数行に処理を行い、グループ全体を対象に単一の戻り値を計算する SQL 関数。集計関数の例としては、SUM や MAX などがあります。

AI

[「人工知能」](#) をご覧ください。

AIOps

[「AI オペレーション」](#) をご覧ください。

匿名化

データセット内の個人情報を完全に削除するプロセス。匿名化は個人のプライバシー保護に役立ちます。匿名化されたデータは、もはや個人データとは見なされません。

アンチパターン

繰り返し起こる問題に対して頻繁に用いられる解決策で、その解決策が逆効果であったり、効果がなかったり、代替案よりも効果が低かったりするもの。

アプリケーション制御

マルウェアからシステムを保護するために、承認されたアプリケーションのみを使用できるようにするセキュリティアプローチ。

アプリケーションポートフォリオ

アプリケーションの構築と維持にかかるコスト、およびそのビジネス価値を含む、組織が使用する各アプリケーションに関する詳細情報の集まり。この情報は、[ポートフォリオの検出と分析プロセス](#)の重要な要素であり、移行、モダナイズ、最適化するアプリケーションを特定し、優先順位を付けるのに役立ちます。

人工知能 (AI)

コンピューティングテクノロジーを使用し、学習、問題の解決、パターンの認識など、通常は人間に関連づけられる認知機能の実行に特化したコンピュータサイエンスの分野。詳細については、「[人工知能 \(AI\) とは何ですか?](#)」をご覧ください。

AI オペレーション (AIOps)

機械学習技術を使用して運用上の問題を解決し、運用上のインシデントと人の介入を減らし、サービス品質を向上させるプロセス。AWS 移行戦略での AIOps の使用方法については、[オペレーション統合ガイド](#)を参照してください。

非対称暗号化

暗号化用のパブリックキーと復号用のプライベートキーから成る 1 組のキーを使用した、暗号化のアルゴリズム。パブリックキーは復号には使用されないため共有しても問題ありませんが、プライベートキーの利用は厳しく制限する必要があります。

原子性、一貫性、分離性、耐久性 (ACID)

エラー、停電、その他の問題が発生した場合でも、データベースのデータ有効性と運用上の信頼性を保証する一連のソフトウェアプロパティ。

属性ベースのアクセス制御 (ABAC)

部署、役職、チーム名など、ユーザーの属性に基づいてアクセス許可をきめ細かく設定する方法。詳細については、AWS Identity and Access Management (IAM) ドキュメントの「[の ABAC AWS](#)」を参照してください。

信頼できるデータソース

最も信頼性のある情報源とされるデータのプライマリーバージョンを保存する場所。匿名化、編集、仮名化など、データを処理または変更する目的で、信頼できるデータソースから他の場所にデータをコピーすることができます。

アベイラビリティゾーン (AZ)

他のアベイラビリティゾーンの障害から AWS リージョン 隔離され、同じリージョン内の他のアベイラビリティゾーンへの低コストで低レイテンシーのネットワーク接続を提供する 内の別の場所。

AWS クラウド導入フレームワーク (AWS CAF)

組織がクラウドへの移行を成功させるための効率的で効果的な計画を立て AWS するための、のガイドラインとベストプラクティスのフレームワークです。AWS CAF は、ビジネス、人材、ガバナンス、プラットフォーム、セキュリティ、運用という 6 つの重点分野にガイダンスを整理しています。ビジネス、人材、ガバナンスの観点では、ビジネススキルとプロセスに重点を置き、プラットフォーム、セキュリティ、オペレーションの視点は技術的なスキルとプロセスに焦点を当てています。例えば、人材の観点では、人事 (HR)、人材派遣機能、および人材管理を扱うステークホルダーを対象としています。この観点から、AWS CAF は、クラウド導入を成功させるための組織の準備に役立つ人材開発、トレーニング、コミュニケーションのガイダンスを提供します。詳細については、[AWS CAF ウェブサイト](#)と [AWS CAF のホワイトペーパー](#) を参照してください。

AWS ワークロード認定フレームワーク (AWS WQF)

データベース移行ワークロードを評価し、移行戦略を推奨し、作業見積もりを提供するツール。AWS WQF は AWS Schema Conversion Tool (AWS SCT) に含まれています。データベーススキーマとコードオブジェクト、アプリケーションコード、依存関係、およびパフォーマンス特性を分析し、評価レポートを提供します。

B

不正なボット

個人や組織に混乱や損害を与えることを目的とした [ボット](#)。

BCP

「[ビジネス継続性計画 \(BCP\)](#)」をご覧ください。

動作グラフ

リソースの動作とインタラクションを経時的に示した、一元的なインタラクティブビュー。Amazon Detective の動作グラフを使用すると、失敗したログオンの試行、不審な API 呼び出し、その他同様のアクションを調べることができます。詳細については、Detective ドキュメントの「[動作グラフのデータ](#)」を参照してください。

ビッグエンディアンシステム

最上位バイトを最初に格納するシステム。「[エンディアン性](#)」もご覧ください。

二項分類

バイナリ結果 (2 つの可能なクラスのうちの一つ) を予測するプロセス。例えば、お客様の機械学習モデルで「この E メールはスパムですか、それともスパムではありませんか」などの問題を予測する必要があるかもしれません。または「この製品は書籍ですか、車ですか」などの問題を予測する必要があるかもしれません。

ブルームフィルター

要素がセットのメンバーであるかどうかをテストするために使用される、確率的でメモリ効率の高いデータ構造。

ブルー/グリーンデプロイ

それぞれが独立しているが、同一の環境を 2 つ作成するデプロイ戦略。現在のアプリケーションバージョンを 1 つの環境 (ブルー) で実行し、新しいアプリケーションバージョンを別の環境 (グリーン) で実行します。この戦略は、最小限の影響で迅速にロールバックするのに役立ちます。

ボット

インターネット経由で自動タスクを実行し、人間のアクティビティややり取りをシミュレートするソフトウェアアプリケーション。インターネット上の情報のインデックスを作成するウェブクローラーなど、一部のボットは有用または有益です。悪質なボットと呼ばれる他のボットの中には、個人や組織を混乱させたり、損害を与えたりすることを意図したものもあります。

ボットネット

[マルウェア](#)に感染しており、ボットハーダーまたはボットオペレーターと呼ばれる単一の当事者によって制御されている[ボット](#)のネットワーク。ボットネットは、ボットとその影響力を拡大する仕組みとして、非常によく知られています。

ブランチ

コードリポジトリに含まれる領域。リポジトリに最初に作成するブランチは、メインブランチといます。既存のブランチから新しいブランチを作成し、その新しいブランチで機能を開発した

り、バグを修正したりできます。機能を構築するために作成するブランチは、通常、機能ブランチと呼ばれます。機能をリリースする準備ができたなら、機能ブランチをメインブランチに統合します。詳細については、「[ブランチの概要](#)」(GitHub ドキュメント)を参照してください。

ブレイクグラスアクセス

例外的な状況では、承認されたプロセスを通じて、ユーザーが AWS アカウント 通常アクセス許可を持たないにすばやくアクセスできるようにします。詳細については、AWS Well-Architected ガイドの「[ブレイクグラス手順の実装](#)」インジケータを参照してください。

ブラウнフィールド戦略

環境の既存インフラストラクチャ。システムアーキテクチャにブラウнフィールド戦略を導入する場合、現在のシステムとインフラストラクチャの制約に基づいてアーキテクチャを設計します。既存のインフラストラクチャを拡張している場合は、ブラウнフィールド戦略と[グリーンフィールド](#)戦略を融合させることもできます。

バッファキャッシュ

アクセス頻度が最も高いデータが保存されるメモリ領域。

ビジネス能力

価値を生み出すためにビジネスが行うこと (営業、カスタマーサービス、マーケティングなど)。マイクロサービスのアーキテクチャと開発の決定は、ビジネス能力によって推進できます。詳細については、[AWSでのコンテナ化されたマイクロサービスの実行](#)ホワイトペーパーの「[ビジネス機能を中心に組織化](#)」セクションを参照してください。

ビジネス継続性計画 (BCP)

大規模移行など、中断を伴うイベントが運用に与える潜在的な影響に対処し、ビジネスを迅速に再開できるようにする計画。

C

CAF

「[AWS クラウド導入フレームワーク](#)」を参照してください

カナリアデプロイ

エンドユーザーへのバージョンリリースを、時間をかけて段階的に行うこと。確信が持てたら新規バージョンをデプロイして、現在のバージョン全体を置き換えます。

CCoE

「[Cloud Center of Excellence](#)」を参照してください。

CDC

「[変更データキャプチャ](#)」を参照してください。

変更データキャプチャ (CDC)

データソース (データベーステーブルなど) の変更を追跡し、その変更に関するメタデータを記録するプロセス。CDC は、ターゲットシステムでの変更を監査またはレプリケートして同期を維持するなど、さまざまな目的に使用できます。

カオスエンジニアリング

障害や破壊的なイベントを意図的に導入して、システムの耐障害性をテストすること。[AWS Fault Injection Service \(AWS FIS\)](#) を使用して、AWS ワークロードにストレスを与え、その応答を評価する実験を実行できます。

CI/CD

「[継続的インテグレーションと継続的デリバリー](#)」を参照してください。

分類

予測を生成するのに役立つ分類プロセス。分類問題の機械学習モデルは、離散値を予測します。離散値は、常に互いに区別されます。例えば、モデルがイメージ内に車があるかどうかを評価する必要がある場合があります。

クライアント側の暗号化

ターゲットが AWS のサービス 受信する前に、ローカルでデータを暗号化します。

Cloud Center of Excellence (CCoE)

クラウドのベストプラクティスの作成、リソースの移動、移行のタイムラインの確立、大規模変革を通じて組織をリードするなど、組織全体のクラウド導入の取り組みを推進する学際的なチーム。詳細については、AWS クラウド エンタープライズ戦略ブログの [CCoE 投稿](#) を参照してください。

クラウドコンピューティング

リモートデータストレージと IoT デバイス管理に通常使用されるクラウドテクノロジー。クラウドコンピューティングは、一般的に、[エッジコンピューティング](#) に接続されています。

クラウド運用モデル

IT 組織において、1 つ以上のクラウド環境を構築、成熟、最適化するために使用される運用モデル。詳細については、「[クラウド運用モデルの構築](#)」を参照してください。

導入のクラウドステージ

組織が、AWS クラウドへの移行時に通常実行する 4 つの段階。

- プロジェクト — 概念実証と学習を目的として、クラウド関連のプロジェクトをいくつか実行する
- 基礎固め — お客様のクラウドの導入を拡大するための基礎的な投資 (ランディングゾーン の作成、CCoE の定義、運用モデルの確立など)
- 移行 — 個々のアプリケーションの移行
- 再発明 — 製品とサービスの最適化、クラウドでのイノベーション

これらのステージは、AWS クラウド エンタープライズ戦略ブログのブログ記事「[クラウドファーストへのジャーニー](#)」と「[導入のステージ](#)」で Stephen Orban によって定義されました。移行戦略との関連性については、AWS「[移行準備ガイド](#)」を参照してください。

CMDB

「[構成管理データベース \(CMDB\)](#)」を参照してください。

コードリポジトリ

ソースコードやその他の資産 (ドキュメント、サンプル、スクリプトなど) が保存され、バージョン管理プロセスを通じて更新される場所。一般的なクラウドリポジトリには、GitHub や Bitbucket Cloud があります。コードの各バージョンはブランチと呼ばれます。マイクロサービスの構造では、各リポジトリは 1 つの機能専用です。1 つの CI/CD パイプラインで複数のリポジトリを使用できます。

コールドキャッシュ

空である、または、かなり空きがある、もしくは、古いデータや無関係なデータが含まれているバッファキャッシュ。データベースインスタンスはメインメモリまたはディスクから読み取る必要があり、バッファキャッシュから読み取るよりも時間がかかるため、パフォーマンスに影響します。

コールドデータ

めったにアクセスされず、通常は過去のデータです。この種類のデータをクエリする場合、通常は低速なクエリでも問題ありません。このデータを低パフォーマンスで安価なストレージ階層またはクラスに移動すると、コストを削減することができます。

コンピュータビジョン (CV)

機械学習を使用してデジタルイメージやビデオといった、ビジュアル形式の情報を分析および抽出する [AI](#) の分野。例えば、Amazon SageMaker AI では、CV 用の画像処理アルゴリズムを利用できます。

設定ドリフト

ワークロードにおいて、設定が想定した状態から変化すること。これによって、ワークロードが非準拠になる可能性があります。この状態は、徐々に生じ、意図的なものではありません。

構成管理データベース (CMDB)

データベースとその IT 環境 (ハードウェアとソフトウェアの両方のコンポーネントとその設定を含む) に関する情報を保存、管理するリポジトリ。通常、CMDB のデータは、移行のポートフォリオの検出と分析の段階で使用します。

コンフォーマンスパック

コンプライアンスチェックとセキュリティチェックをカスタマイズするためにアセンブルできる AWS Config ルールと修復アクションのコレクション。YAML テンプレートを使用して、コンフォーマンスパックを AWS アカウント および リージョンの単一のエンティティとしてデプロイすることも、組織全体にデプロイすることもできます。詳細については、AWS Config ドキュメントの「[コンフォーマンスパック](#)」を参照してください。

継続的インテグレーションと継続的デリバリー (CI/CD)

ソフトウェアリリースプロセスのソース、ビルド、テスト、ステージング、本番の各ステージを自動化するプロセス。CI/CD は一般的にパイプラインと呼ばれます。プロセスの自動化、生産性の向上、コード品質の向上、配信の加速化を可能にします。詳細については、「[継続的デリバリーの利点](#)」を参照してください。CD は継続的デプロイ (Continuous Deployment) の略語でもあります。詳細については「[継続的デリバリーと継続的なデプロイ](#)」を参照してください。

CV

[「コンピュータビジョン」](#) を参照してください。

D

保管中のデータ

ストレージ内にあるデータなど、常に自社のネットワーク内にあるデータ。

データ分類

ネットワーク内のデータを重要度と機密性に基づいて識別、分類するプロセス。データに適した保護および保持のコントロールを判断する際に役立つため、あらゆるサイバーセキュリティのリスク管理戦略において重要な要素です。データ分類は、AWS Well-Architected フレームワークのセキュリティの柱のコンポーネントです。詳細については、「[データ分類](#)」を参照してください。

データドリフト

実稼働データと ML モデルのトレーニングに使用されたデータとの間に有意な差異が生じたり、入力データが時間の経過と共に有意に変化したりすることです。データドリフトは、ML モデル予測の全体的な品質、精度、公平性を低下させる可能性があります。

転送中のデータ

ネットワーク内 (ネットワークリソース間など) を活発に移動するデータ。

データメッシュ

非一元的で分散型のデータ所有権を持つとともに、一元的な管理およびガバナンスを行えるアーキテクチャフレームワーク。

データ最小化

厳密に必要なデータのみを収集し、処理するという原則。でデータ最小化を実践 AWS クラウドすることで、プライバシーリスク、コスト、分析のカーボンフットプリントを削減できます。

データ境界

AWS 環境内の一連の予防ガードレール。信頼できる ID のみが、期待されるネットワークから信頼できるリソースにアクセスできるようにします。詳細については、「[でのデータ境界の構築 AWS](#)」を参照してください。

データの前処理

raw データをお客様の機械学習モデルで簡単に解析できる形式に変換すること。データの前処理とは、特定の列または行を削除して、欠落している、矛盾している、または重複する値に対処することを意味します。

データ出所

データの生成、送信、保存の方法など、データのライフサイクル全体を通じてデータの出所と履歴を追跡するプロセス。

データ件名

データを収集、処理している個人。

データウェアハウス

分析などのビジネスインテリジェンスをサポートするデータ管理システム。データウェアハウスには、一般的に、大量の履歴データが含まれており、多くの場合、それらはクエリや分析に使用されます。

データベース定義言語 (DDL)

データベース内のテーブルやオブジェクトの構造を作成または変更するためのステートメントまたはコマンド。

データベース操作言語 (DML)

データベース内の情報を変更 (挿入、更新、削除) するためのステートメントまたはコマンド。

DDL

「[データベース定義言語](#)」を参照してください。

ディープアンサンブル

予測のために複数の深層学習モデルを組み合わせます。ディープアンサンブルを使用して、より正確な予測を取得したり、予測の不確実性を推定したりできます。

深層学習

人工ニューラルネットワークの複数層を使用して、入力データと対象のターゲット変数の間のマッピングを識別する機械学習サブフィールド。

多層防御

一連のセキュリティメカニズムとコントロールをコンピュータネットワーク全体に層状に重ねて、ネットワークとその内部にあるデータの機密性、整合性、可用性を保護する情報セキュリティの手法。この戦略を採用するときは AWS、AWS Organizations 構造の異なるレイヤーに複数のコントロールを追加して、リソースの安全性を確保します。たとえば、多層防御アプローチでは、多要素認証、ネットワークセグメンテーション、暗号化を組み合わせることができます。

委任管理者

では AWS Organizations、互換性のあるサービスが AWS メンバーアカウントを登録して組織のアカウントを管理し、そのサービスのアクセス許可を管理できます。このアカウントを、そのサービスの委任管理者と呼びます。詳細、および互換性のあるサービスの一覧は、AWS

Organizations ドキュメントの「[AWS Organizations で使用できるサービス](#)」を参照してください。

トラブルシューティング

アプリケーション、新機能、コードの修正をターゲットの環境で利用できるようにするプロセス。デプロイでは、コードベースに変更を施した後、アプリケーションの環境でそのコードベースを構築して実行します。

開発環境

「[環境](#)」を参照してください。

検出管理

イベントが発生したときに、検出、ログ記録、警告を行うように設計されたセキュリティコントロール。これらのコントロールは副次的な防衛手段であり、実行中の予防的コントロールをすり抜けたセキュリティイベントをユーザーに警告します。詳細については、「[AWS でのセキュリティコントロールの実装](#)」の「[検出的コントロール](#)」を参照してください。

開発バリューストリームマッピング (DVSM)

ソフトウェア開発ライフサイクルのスピードと品質に悪影響を及ぼす制約を特定し、優先順位を付けるために使用されるプロセス。DVSM は、もともとリーンマニファクチャリング・プラクティスのために設計されたバリューストリームマッピング・プロセスを拡張したものです。ソフトウェア開発プロセスを通じて価値を創造し、動かすために必要なステップとチームに焦点を当てています。

デジタルツイン

建物、工場、産業機器、生産ラインなど、現実世界のシステムを仮想的に表現したものです。デジタルツインは、予知保全、リモートモニタリング、生産最適化をサポートします。

ディメンションテーブル

[スタースキーマ](#)において、ファクトテーブルの定量データに関するデータ属性が含まれる小さいテーブル。ディメンションテーブルの属性は、通常、テキストフィールド、またはテキストのように扱える個別の数値で示されます。これらの属性は、一般的に、クエリの制約、フィルタリング、結果セットのラベル付けに使用されます。

ディザスタ

ワークロードまたはシステムが、導入されている主要な場所でのビジネス目標の達成を妨げるイベント。これらのイベントは、自然災害、技術的障害、または意図しない設定ミスやマルウェア攻撃などの人間の行動の結果である場合があります。

ディザスタリカバリ (DR)

[ディザスタ](#)によるダウンタイムとデータ損失を最小限に抑えるための戦略とプロセス。詳細については、AWS Well-Architected フレームワークの「[Disaster Recovery of Workloads on AWS: Recovery in the Cloud](#)」を参照してください。

DML

「[データベース操作言語](#)」を参照してください。

ドメイン駆動型設計

各コンポーネントが提供している変化を続けるドメイン、またはコアビジネス目標にコンポーネントを接続して、複雑なソフトウェアシステムを開発するアプローチ。この概念は、エリック・エヴァンスの著書、Domain-Driven Design: Tackling Complexity in the Heart of Software (ドメイン駆動設計:ソフトウェアの中心における複雑さへの取り組み) で紹介されています (ポストン: Addison-Wesley Professional、2003)。strangler fig パターンでドメイン駆動型設計を使用する方法の詳細については、「[コンテナと Amazon API Gateway を使用して、従来の Microsoft ASP.NET \(ASMX\) ウェブサービスを段階的にモダナイズ](#)」を参照してください。

DR

「[ディザスタリカバリ](#)」を参照してください。

ドリフト検出

ベースライン設定からの偏差を追跡します。たとえば、AWS CloudFormation を使用して[システムリソースのドリフトを検出](#)したり、を使用して AWS Control Tower、ガバナンス要件への準拠に影響する[ランディングゾーンの変更を検出](#)したりできます。

DVSM

「[開発バリューSTREAMマッピング](#)」を参照してください。

E

EDA

「[探索的データ分析](#)」を参照してください。

EDI

「[電子データ交換](#)」を参照してください。

エッジコンピューティング

IoT ネットワークのエッジにあるスマートデバイスの計算能力を高めるテクノロジー。[クラウドコンピューティング](#)と比較すると、エッジコンピューティングは通信レイテンシーを短縮し、応答時間を改善できます。

電子データ交換 (EDI)

組織間で行う、ビジネスドキュメントの自動交換。詳細については、[「電子データ交換とは」](#)を参照してください。

暗号化

人間が読み取り可能なプレーンテキストデータを暗号文に変換するコンピューティング処理。

暗号化キー

暗号化アルゴリズムが生成した、ランダム化されたビットからなる暗号文字列。キーの長さは決まっておらず、各キーは予測できないように、一意になるように設計されています。

エンディアン

コンピュータメモリにバイトが格納される順序。ビッグエンディアンシステムでは、最上位バイトが最初に格納されます。リトルエンディアンシステムでは、最下位バイトが最初に格納されます。

エンドポイント

[「サービスエンドポイント」](#)を参照してください。

エンドポイントサービス

仮想プライベートクラウド (VPC) 内でホストして、他のユーザーと共有できるサービス。を使用してエンドポイントサービスを作成し AWS PrivateLink、他の AWS アカウント または AWS Identity and Access Management (IAM) プリンシパルにアクセス許可を付与できます。これらのアカウントまたはプリンシパルは、インターフェイス VPC エンドポイントを作成することで、エンドポイントサービスにプライベートに接続できます。詳細については、Amazon Virtual Private Cloud (Amazon VPC) ドキュメントの [「エンドポイントサービスを作成する」](#)を参照してください。

エンタープライズリソースプランニング (ERP)

エンタープライズの主要なビジネスプロセス (会計、[MES](#)、プロジェクト管理など) を自動化および管理するシステム。

エンベロープ暗号化

暗号化キーを、別の暗号化キーを使用して暗号化するプロセス。詳細については、AWS Key Management Service (AWS KMS) ドキュメントの「[エンベロープ暗号化](#)」を参照してください。

環境

実行中のアプリケーションのインスタンス。クラウドコンピューティングにおける一般的な環境の種類は以下のとおりです。

- **開発環境** — アプリケーションのメンテナンスを担当するコアチームのみが利用できる、実行中のアプリケーションのインスタンス。開発環境は、上位の環境に昇格させる変更をテストするときに使用します。このタイプの環境は、テスト環境と呼ばれることもあります。
- **下位環境** — 初期ビルドやテストに使用される環境など、アプリケーションのすべての開発環境。
- **本番環境** — エンドユーザーがアクセスできる、実行中のアプリケーションのインスタンス。CI/CD パイプラインでは、本番環境が最後のデプロイ環境になります。
- **上位環境** — コア開発チーム以外のユーザーがアクセスできるすべての環境。これには、本番環境、本番前環境、ユーザー承認テスト環境などが含まれます。

エピック

アジャイル方法論で、お客様の作業の整理と優先順位付けに役立つ機能カテゴリ。エピックでは、要件と実装タスクの概要についてハイレベルな説明を提供します。例えば、AWS CAF セキュリティエピックには、ID とアクセスの管理、検出コントロール、インフラストラクチャセキュリティ、データ保護、インシデント対応が含まれます。AWS 移行戦略のエピックの詳細については、[プログラム実装ガイド](#)を参照してください。

ERP

「[エンタープライズリソース計画](#)」を参照してください。

探索的データ分析 (EDA)

データセットを分析してその主な特性を理解するプロセス。お客様は、データを収集または集計してから、パターンの検出、異常の検出、および前提条件のチェックのための初期調査を実行します。EDA は、統計の概要を計算し、データの可視化を作成することによって実行されます。

F

ファクトテーブル

[スタースキーマ](#)の中央にあるテーブル。ビジネスオペレーションに関する定量的データが保存されます。一般的に、ファクトテーブルは、2種類の列で構成されます。1つは測定値が含まれる列、もう1つはディメンションテーブルへの外部キーが含まれる列です。

フェイルファスト

開発ライフサイクルを短縮するために、頻繁かつ段階的にテストを行う哲学であり、アジャイルアプローチでは、この考え方がきわめて重要です。

障害分離境界

では AWS クラウド、アベイラビリティゾーン AWS リージョン、コントロールプレーン、データプレーンなどの境界で、障害の影響を制限し、ワークロードの耐障害性を向上させるのに役立ちます。詳細については、「[AWS 障害分離境界](#)」を参照してください。

機能ブランチ

「[ブランチ](#)」を参照してください。

特徴量

お客様が予測に使用する入力データ。例えば、製造コンテキストでは、特徴量は製造ラインから定期的にキャプチャされるイメージの可能性もあります。

特徴量重要度

モデルの予測に対する特徴量の重要性。これは通常、Shapley Additive Deskonations (SHAP) や積分勾配など、さまざまな手法で計算できる数値スコアで表されます。詳細については、「[を使用した機械学習モデルの解釈可能性 AWS](#)」を参照してください。

機能変換

追加のソースによるデータのエンリッチ化、値のスケーリング、単一のデータフィールドからの複数の情報セットの抽出など、機械学習プロセスのデータを最適化すること。これにより、機械学習モデルはデータの恩恵を受けることができます。例えば、「2021-05-27 00:15:37」の日付を「2021年」、「5月」、「木」、「15」に分解すると、学習アルゴリズムがさまざまなデータコンポーネントに関連する微妙に異なるパターンを学習するのに役立ちます。

数ショットプロンプト

[LLM](#) に、タスクと望ましい出力を示す例を少数提示した後に、類似のタスクを実行させること。この手法は、プロンプトに記述された例(ショット)からモデルが学習する「インコンテキスト学

習」の一種です。数ショットプロンプトは、特定のフォーマット、推論、専門知識が必要なタスクに効果的です。「[ゼロショットプロンプト](#)」も参照してください。

FGAC

「[きめ細かなアクセス制御](#)」を参照してください。

きめ細かなアクセス制御 (FGAC)

複数の条件を使用してアクセス要求を許可または拒否すること。

フラッシュカット移行

[変更データのキャプチャ](#)による継続的なデータ複製を利用して、段階的なアプローチではなく、可能な限り短時間でデータを移行するデータベース移行方法。目的はダウンタイムを最小限に抑えることです。

FM

「[基盤モデル](#)」を参照してください。

基盤モデル (FM)

大規模な深層学習ニューラルネットワークであり、一般化およびラベル付けされていないデータからなる大規模データセットでトレーニングされています。FM により、言語理解、テキストおよび画像生成、自然言語での会話といった、一般的な各種タスクを実行できます。詳細については、「[基盤モデルとは何ですか?](#)」を参照してください。

G

生成 AI

[AI](#) モデルのサブセット。大量のデータでトレーニングされており、シンプルなテキストプロンプトを使用して、画像、動画、テキスト、オーディオなどの新しいコンテンツやアーティファクトを作成できます。詳細については、「[生成 AI とは何ですか?](#)」を参照してください。

ジオブロッキング

「[地理的制限](#)」を参照してください。

地理的制限 (ジオブロッキング)

特定の国のユーザーがコンテンツ配信にアクセスできないようにするための、Amazon CloudFront のオプション。アクセスを許可する国と禁止する国は、許可リストまたは禁止リスト

を使って指定します。詳細については、CloudFront ドキュメントの「[コンテンツの地理的ディストリビューションの制限](#)」を参照してください。

Gitflow ワークフロー

下位環境と上位環境が、ソースコードリポジトリでそれぞれ異なるブランチを使用する方法。Gitflow ワークフローは古いと見なされている方法であり、[トランクベースのワークフロー](#)は推奨されている新しい方法です。

ゴールデンイメージ

システムまたはソフトウェアのスナップショットであり、システムまたはソフトウェアの新規インスタンスをデプロイするテンプレートとして使用されます。製造の例で言えば、ゴールデンイメージを使用すると、複数のデバイスにソフトウェアをプロビジョニングして、デバイス製造オペレーションの速度、スケーラビリティ、生産性を向上させることができます。

グリーンフィールド戦略

新しい環境に既存のインフラストラクチャが存在しないこと。システムアーキテクチャにグリーンフィールド戦略を導入する場合、既存のインフラストラクチャ (別名 [ブラウンフィールド](#)) との互換性の制約を受けることなく、あらゆる新しいテクノロジーを選択できます。既存のインフラストラクチャを拡張している場合は、ブラウンフィールド戦略とグリーンフィールド戦略を融合させることもできます。

ガードレール

組織単位 (OU) 全般のリソース、ポリシー、コンプライアンスを管理するのに役立つ概略的なルール。予防ガードレールは、コンプライアンス基準に一致するようにポリシーを実施します。これらは、サービスコントロールポリシーと IAM アクセス許可の境界を使用して実装されます。検出ガードレールは、ポリシー違反やコンプライアンス上の問題を検出し、修復のためのアラートを発信します。これらは AWS Config、AWS Security Hub CSPM、Amazon GuardDuty、AWS Trusted Advisor Amazon Inspector、およびカスタム AWS Lambda チェックを使用して実装されます。

H

HA

「[高可用性](#)」を参照してください。

異種混在データベースの移行

別のデータベースエンジンを使用するターゲットデータベースへお客様の出典データベースの移行 (例えば、Oracle から Amazon Aurora)。異種間移行は通常、アーキテクチャの再設計作業の一部であり、スキーマの変換は複雑なタスクになる可能性があります。[AWS は、スキーマの変換に役立つ AWS SCT を提供します。](#)

高可用性 (HA)

課題や災害が発生した場合に、介入なしにワークロードを継続的に運用できること。HA システムは、自動的にフェイルオーバーし、一貫して高品質のパフォーマンスを提供し、パフォーマンスへの影響を最小限に抑えながらさまざまな負荷や障害を処理するように設計されています。

ヒストリアンのモダナイゼーション

製造業のニーズによりよく応えるために、オペレーションテクノロジー (OT) システムをモダナイズし、アップグレードするためのアプローチ。ヒストリアンは、工場内のさまざまなソースからデータを収集して保存するために使用されるデータベースの一種です。

ホールドアウトデータ

[機械学習](#) モデルのトレーニング用データセットから保留される、ラベル付き履歴データの一部。ホールドアウトデータを使用すると、モデル予測をホールドアウトデータと比較して、モデルのパフォーマンスを評価できます。

同種データベースの移行

お客様の出典データベースを、同じデータベースエンジンを共有するターゲットデータベース (Microsoft SQL Server から Amazon RDS for SQL Server など) に移行する。同種間移行は、通常、リホストまたはリプラットフォーム化の作業の一部です。ネイティブデータベースユーティリティを使用して、スキーマを移行できます。

ホットデータ

リアルタイムデータや最近の翻訳データなど、頻繁にアクセスされるデータ。通常、このデータには高速なクエリ応答を提供する高性能なストレージ階層またはクラスが必要です。

ホットフィックス

本番環境の重大な問題を修正するために緊急で配布されるプログラム。緊急性が高いため、通常の DevOps のリリースワークフローからは外れた形で実施されます。

ハイパーケア期間

カットオーバー直後、移行したアプリケーションを移行チームがクラウドで管理、監視して問題に対処する期間。通常、この期間は 1~4 日です。ハイパーケア期間が終了すると、アプリケーションに対する責任は一般的に移行チームからクラウドオペレーションチームに移ります。

I

laC

「[Infrastructure as Code](#)」を参照してください。

ID ベースのポリシー

AWS クラウド 環境内のアクセス許可を定義する 1 つ以上の IAM プリンシパルにアタッチされたポリシー。

アイドル状態のアプリケーション

90 日間の平均的な CPU およびメモリ使用率が 5~20% のアプリケーション。移行プロジェクトでは、これらのアプリケーションを廃止するか、オンプレミスに保持するのが一般的です。

IIoT

「[インダストリアル IoT](#)」を参照してください。

イミュータブルインフラストラクチャ

既存インフラストラクチャの更新、パッチ適用、変更などを行わずに、本番環境ワークロードに使用する新規インフラストラクチャをデプロイするモデル。本質的に、イミュータブルインフラストラクチャは、[ミュータブルインフラストラクチャ](#)よりも一貫性、信頼性、予測性に優れています。詳細については、AWS Well-Architected フレームワークにある「[イミュータブルインフラストラクチャを使用してデプロイする](#)」のベストプラクティスを参照してください。

インバウンド (受信) VPC

AWS マルチアカウントアーキテクチャでは、アプリケーションの外部からネットワーク接続を受け入れ、検査し、ルーティングする VPC。[AWS Security Reference Architecture](#) では、アプリケーションとより広範なインターネット間の双方向のインターフェイスを保護するために、インバウンド、アウトバウンド、インスペクションの各 VPC を使用してネットワークアカウントを設定することを推奨しています。

I

増分移行

アプリケーションを 1 回ですべてカットオーバーするのではなく、小さい要素に分けて移行するカットオーバー戦略。例えば、最初は少数のマイクロサービスまたはユーザーのみを新しいシステムに移行する場合があります。すべてが正常に機能することを確認できたら、残りのマイクロサービスやユーザーを段階的に移行し、レガシーシステムを廃止できるようにします。この戦略により、大規模な移行に伴うリスクが軽減されます。

インダストリー 4.0

2016 年に [Klaus Schwab](#) 氏が提唱した用語で、接続、リアルタイムデータ、オートメーション、分析、AI/ML の進歩による、ビジネスプロセスのモダナイズを意味します。

インフラストラクチャ

アプリケーションの環境に含まれるすべてのリソースとアセット。

Infrastructure as Code (IaC)

アプリケーションのインフラストラクチャを一連の設定ファイルを使用してプロビジョニングし、管理するプロセス。IaC は、新しい環境を再現可能で信頼性が高く、一貫性のあるものにするため、インフラストラクチャを一元的に管理し、リソースを標準化し、スケールを迅速に行えるように設計されています。

インダストリアル IoT (IIoT)

製造、エネルギー、自動車、ヘルスケア、ライフサイエンス、農業などの産業部門におけるインターネットに接続されたセンサーやデバイスの使用。詳細については、「[インダストリアル IoT \(IIoT\) デジタルトランスフォーメーション戦略の構築](#)」を参照してください。

インスペクション VPC

AWS マルチアカウントアーキテクチャでは、VPC (同一または異なる 内 AWS リージョン)、インターネット、オンプレミスネットワーク間のネットワークトラフィックの検査を管理する一元化された VPCs。 [AWS Security Reference Architecture](#) では、アプリケーションとより広範なインターネット間の双方向のインターフェイスを保護するために、インバウンド、アウトバウンド、インスペクションの各 VPC を使用してネットワークアカウントを設定することを推奨しています。

IoT

インターネットまたはローカル通信ネットワークを介して他のデバイスやシステムと通信する、センサーまたはプロセッサが組み込まれた接続済み物理オブジェクトのネットワーク。詳細については、「[IoT とは](#)」を参照してください。

解釈可能性

機械学習モデルの特性で、モデルの予測がその入力にどのように依存するかを人間が理解できる度合いを表します。詳細については、[「を使用した機械学習モデルの解釈可能性 AWS」](#) を参照してください。

IoT

[「IoT」](#) を参照してください。

IT 情報ライブラリ (ITIL)

IT サービスを提供し、これらのサービスをビジネス要件に合わせるための一連のベストプラクティス。ITIL は ITSM の基盤を提供します。

IT サービス管理 (ITSM)

組織の IT サービスの設計、実装、管理、およびサポートに関連する活動。クラウドオペレーションと ITSM ツールの統合については、[オペレーション統合ガイド](#) を参照してください。

ITIL

[「IT 情報ライブラリ」](#) を参照してください。

ITSM

[「IT サービス管理」](#) を参照してください。

L

ラベルベースアクセス制御 (LBAC)

強制アクセス制御 (MAC) の実装で、ユーザーとデータ自体にそれぞれセキュリティラベル値が明示的に割り当てられます。ユーザーセキュリティラベルとデータセキュリティラベルが交差する部分によって、ユーザーに表示される行と列が決まります。

ランディングゾーン

ランディングゾーンは、スケーラブルで安全な、適切に設計されたマルチアカウント AWS 環境です。これは、組織がセキュリティおよびインフラストラクチャ環境に自信を持ってワークロードとアプリケーションを迅速に起動してデプロイできる出発点です。ランディングゾーンの詳細については、[「安全でスケーラブルなマルチアカウント AWS 環境のセットアップ」](#) を参照してください。

大規模言語モデル (LLM)

大量のデータで事前トレーニングされた深層学習 AI モデル。LLM では、質問への回答、ドキュメントの要約、他言語へのテキスト翻訳、文を完成させるなど、さまざまなタスクを実行できます。詳細については、「[大規模言語モデル \(LLM\) とは何ですか?](#)」を参照してください。

大規模な移行

300 台以上のサーバの移行。

LBAC

「[ラベルベースアクセス制御](#)」を参照してください。

最小特権

タスクの実行には必要最低限の権限を付与するという、セキュリティのベストプラクティス。詳細については、IAM ドキュメントの「[最小特権アクセス許可を適用する](#)」を参照してください。

リフトアンドシフト

「[7 Rs](#)」を参照してください。

リトルエンディアンシステム

最下位バイトを最初に格納するシステム。「[エンディアン性](#)」もご覧ください。

LLM

「[大規模言語モデル](#)」を参照してください。

下位環境

「[環境](#)」を参照してください。

M

機械学習 (ML)

パターン認識と学習にアルゴリズムと手法を使用する人工知能の一種。ML は、モノのインターネット (IoT) データなどの記録されたデータを分析して学習し、パターンに基づく統計モデルを生成します。詳細については、「[機械学習](#)」を参照してください。

メインブランチ

「[ブランチ](#)」を参照してください。

マルウェア

コンピュータのセキュリティやプライバシーを侵害するように設計されたソフトウェア。マルウェアは、コンピュータシステムの中断、機密情報の漏洩、不正アクセスを招く可能性があります。マルウェアの例には、ウイルス、ワーム、ランサムウェア、トロイの木馬、スパイウェア、キーロガーなどがあります。

マネージドサービス

AWS のサービスはインフラストラクチャレイヤー、オペレーティングシステム、プラットフォーム AWS を運用し、エンドポイントにアクセスしてデータを保存および取得します。マネージドサービスの例として、Amazon Simple Storage Service (Amazon S3) と Amazon DynamoDB が挙げられます。このサービスは、抽象化されたサービスとも呼ばれます。

製造実行システム (MES)

生産プロセスを追跡、モニタリング、文書化、制御するソフトウェアシステムであり、工場では、これによって、原材料から製品を完成させます。

MAP

[「Migration Acceleration Program」](#) を参照してください。

メカニズム

ツールを作成してその導入を推進し、導入結果を調べて調整を行うための包括的なプロセス。メカニズムとは、運用中にそれ自体を強化し改善するサイクルを意味します。詳細については、AWS 「Well-Architected フレームワーク」の [「メカニズムの構築」](#) を参照してください。

メンバーアカウント

組織の一部である管理アカウント AWS アカウント 以外のすべて AWS Organizations。アカウントが組織のメンバーになることができるのは、一度に 1 つのみです。

MES

[「製造実行システム」](#) を参照してください。

Message Queuing Telemetry Transport (MQTT)

[発行/サブスクリプション](#) のパターンに基づく、軽量のマシンツーマシン (M2M) 通信プロトコルであり、リソースに限りのある [IoT](#) デバイスに使用されます。

マイクロサービス

明確に定義された API を介して通信し、通常は小規模な自己完結型のチームが所有する、小規模で独立したサービスです。例えば、保険システムには、販売やマーケティングなどのビジネス

機能、または購買、請求、分析などのサブドメインにマッピングするマイクロサービスが含まれる場合があります。マイクロサービスの利点には、俊敏性、柔軟なスケーリング、容易なデプロイ、再利用可能なコード、回復力などがあります。詳細については、[AWS「サーバーレスサービスを使用したマイクロサービスの統合」](#)を参照してください。

マイクロサービスアーキテクチャ

各アプリケーションプロセスをマイクロサービスとして実行する独立したコンポーネントを使用してアプリケーションを構築するアプローチ。これらのマイクロサービスは、軽量 API を使用して、明確に定義されたインターフェイスを介して通信します。このアーキテクチャの各マイクロサービスは、アプリケーションの特定の機能に対する需要を満たすように更新、デプロイ、およびスケーリングできます。詳細については、「[でのマイクロサービスの実装 AWS](#)」を参照してください。

Migration Acceleration Program (MAP)

組織がクラウドに移行するための強力な運用基盤を構築し、移行の初期コストを相殺するのに役立つコンサルティングサポート、トレーニング、サービスを提供する AWS プログラム。MAP には、組織的な方法でレガシー移行を実行するための移行方法論と、一般的な移行シナリオを自動化および高速化する一連のツールが含まれています。

大規模な移行

アプリケーションポートフォリオの大部分を次々にクラウドに移行し、各ウェーブでより多くのアプリケーションを高速に移動させるプロセス。この段階では、以前の段階から学んだベストプラクティスと教訓を使用して、移行ファクトリー チーム、ツール、プロセスのうち、オートメーションとアジャイルデリバリーによってワークロードの移行を合理化します。これは、[AWS 移行戦略](#) の第 3 段階です。

移行ファクトリー

自動化された俊敏性のあるアプローチにより、ワークロードの移行を合理化する部門横断的なチーム。移行ファクトリーチームには、通常、運用、ビジネスアナリストおよび所有者、移行エンジニア、デベロッパー、およびスプリントで作業する DevOps プロフェッショナルが含まれます。エンタープライズアプリケーションポートフォリオの 20~50% は、ファクトリーのアプローチによって最適化できる反復パターンで構成されています。詳細については、このコンテンツセットの[移行ファクトリーに関する解説](#)と [Cloud Migration Factory ガイド](#)を参照してください。

移行メタデータ

移行を完了するために必要なアプリケーションおよびサーバーに関する情報。移行パターンごとに、異なる一連の移行メタデータが必要です。移行メタデータの例としては、ターゲットサブネット、セキュリティグループ、AWS アカウントなどがあります。

移行パターン

移行戦略、移行先、および使用する移行アプリケーションまたはサービスを詳述する、反復可能な移行タスク。例: AWS Application Migration Service を使用して Amazon EC2 への移行をリホストします。

Migration Portfolio Assessment (MPA)

オンラインツール。これによって、AWS クラウドに移行するビジネスケースの検証に必要な情報を得られます。MPA は、詳細なポートフォリオ評価 (サーバーの適切なサイジング、価格設定、TCO 比較、移行コスト分析) および移行プラン (アプリケーションデータの分析とデータ収集、アプリケーションのグループ化、移行の優先順位付け、およびウェーブプランニング) を提供します。[MPA ツール](#) (ログインが必要) は、すべての AWS コンサルタントと APN パートナー コンサルタントが無料で利用できます。

移行準備状況評価 (MRA)

AWS CAF を使用して、組織のクラウド準備状況に関するインサイトを取得し、長所と短所を特定し、特定されたギャップを埋めるためのアクションプランを構築するプロセス。詳細については、[移行準備状況ガイド](#)を参照してください。MRA は、[AWS 移行戦略](#)の第一段階です。

移行戦略

ワークロードを AWS クラウドに移行するために使用するアプローチ。詳細については、この用語集の [7 Rs](#) エントリと、「[組織を動員して大規模な移行を加速する](#)」を参照してください。

ML

「[機械学習](#)」を参照してください。

モダナイゼーション

古い (レガシーまたはモノリシック) アプリケーションとそのインフラストラクチャをクラウド内の俊敏で弾力性のある高可用性システムに変換して、コストを削減し、効率を高め、イノベーションを活用します。詳細については、「[AWS クラウドでのアプリケーションのモダナイズ戦略](#)」を参照してください。

モダナイゼーション準備状況評価

組織のアプリケーションのモダナイゼーションの準備状況を判断し、利点、リスク、依存関係を特定し、組織がこれらのアプリケーションの将来の状態をどの程度適切にサポートできるかを決定するのに役立つ評価。評価の結果として、ターゲットアーキテクチャのブループリント、モダナイゼーションプロセスの開発段階とマイルストーンを詳述したロードマップ、特定されたギャップに対処するためのアクションプランが得られます。詳細については、「[AWS クラウドでのアプリケーションのモダナイゼーションの準備状況を評価する](#)」を参照してください。

モノリシックアプリケーション (モノリス)

緊密に結合されたプロセスを持つ単一のサービスとして実行されるアプリケーション。モノリシックアプリケーションにはいくつかの欠点があります。1つのアプリケーション機能エクスペリエンスの需要が急増する場合は、アーキテクチャ全体をスケーリングする必要があります。モノリシックアプリケーションの特徴を追加または改善することは、コードベースが大きくなると複雑になります。これらの問題に対処するには、マイクロサービスアーキテクチャを使用できます。詳細については、「[モノリスをマイクロサービスに分解する](#)」を参照してください。

MPA

「[Migration Portfolio Assessment](#)」を参照してください。

MQTT

「[Message Queuing Telemetry Transport](#)」を参照してください。

多クラス分類

複数のクラスの予測を生成するプロセス (2 つ以上の結果の 1 つを予測します)。例えば、機械学習モデルが、「この製品は書籍、自動車、電話のいずれですか?」または、「このお客様にとって最も関心のある商品のカテゴリはどれですか?」と聞くかもしれません。

ミュータブルなインフラストラクチャ

本番ワークロードに使用する既存のインフラストラクチャを更新および変更するためのモデル。Well-Architected AWS フレームワークでは、一貫性、信頼性、予測可能性を向上させるために、[イミュータブルインフラストラクチャ](#)の使用をベストプラクティスとして推奨しています。

O

OAC

「[オリジンアクセス制御](#)」を参照してください。

OAI

「[オリジンアクセスアイデンティティ](#)」を参照してください。

OCM

「[組織変更管理](#)」を参照してください。

オフライン移行

移行プロセス中にソースワークロードを停止させる移行方法。この方法はダウンタイムが長くなるため、通常は重要ではない小規模なワークロードに使用されます。

OI

「[オペレーション統合](#)」を参照してください。

Ola

「[オペレーショナルレベルアグリーメント](#)」を参照してください。

オンライン移行

ソースワークロードをオフラインにせずにターゲットシステムにコピーする移行方法。ワークロードに接続されているアプリケーションは、移行中も動作し続けることができます。この方法はダウンタイムがゼロから最小限で済むため、通常は重要な本番稼働環境のワークロードに使用されます。

OPC-UA

「[Open Process Communications - Unified Architecture](#)」を参照してください。

Open Process Communications - Unified Architecture (OPC-UA)

産業オートメーション用のマシンツーマシン (M2M) 通信プロトコル。OPC-UA により、相互運用の際に、データ暗号化、認証、認可の各スキームを標準化できます。

オペレーショナルレベルアグリーメント (OLA)

サービスレベルアグリーメント (SLA) をサポートするために、どの機能的 IT グループが互いに提供することを約束するかを明確にする契約。

運用準備状況レビュー (ORR)

質問と関連するベストプラクティスのチェックリスト。インシデントや起こり得る障害を理解、評価、防止したり、その範囲を縮小したりする際に役立ちます。詳細については、AWS Well-Architected フレームワークの「[Operational Readiness Reviews \(ORR\)](#)」を参照してください。

運用テクノロジー (OT)

産業オペレーション、機器、インフラストラクチャを制御するために物理環境と連携させるハードウェアおよびソフトウェアシステム。製造分野では、[Industry 4.0](#) への変革を進める上で、OT と情報技術 (IT) システムの統合に焦点が当てられています。

オペレーション統合 (OI)

クラウドでオペレーションをモダナイズするプロセスには、準備計画、オートメーション、統合が含まれます。詳細については、[オペレーション統合ガイド](#)を参照してください。

組織の証跡

組織 AWS アカウント 内のすべてのイベント AWS CloudTrail をログに記録することによって作成された証跡 AWS Organizations。証跡は、組織に含まれている各 AWS アカウントに作成され、各アカウントのアクティビティを追跡します。詳細については、CloudTrail ドキュメントの「[組織の証跡の作成](#)」を参照してください。

組織変更管理 (OCM)

人材、文化、リーダーシップの観点から、主要な破壊的なビジネス変革を管理するためのフレームワーク。OCM は、変化の導入を加速し、移行問題に対処し、文化や組織の変化を推進することで、組織が新しいシステムと戦略の準備と移行するのを支援します。AWS 移行戦略では、クラウド導入プロジェクトに必要な変化のスピードにより、このフレームワークは人材アクセラレーションと呼ばれます。詳細については、[OCM ガイド](#)を参照してください。

オリジンアクセス制御 (OAC)

Amazon Simple Storage Service (Amazon S3) コンテンツを保護するための、CloudFront のアクセス制限の強化オプション。OAC は AWS リージョン、すべての S3 バケット、AWS KMS (SSE-KMS) によるサーバー側の暗号化、S3 バケットへの動的 PUT および DELETE リクエストをサポートします。

オリジンアクセスアイデンティティ (OAI)

CloudFront の、Amazon S3 コンテンツを保護するためのアクセス制限オプション。OAI を使用すると、CloudFront が、Amazon S3 に認証可能なプリンシパルを作成します。認証されたプリンシパルは、S3 バケット内のコンテンツに、特定の CloudFront ディストリビューションを介してのみアクセスできます。[OAC](#) も併せて参照してください。OAC では、より詳細な、強化されたアクセス制御が可能です。

ORR

「[運用準備状況レビュー](#)」を参照してください。

O

「[運用テクノロジー](#)」を参照してください。

アウトバウンド (送信) VPC

AWS マルチアカウントアーキテクチャでは、アプリケーション内から開始されたネットワーク接続を処理する VPC。[AWS Security Reference Architecture](#) では、アプリケーションとより広範なインターネット間の双方向のインターフェイスを保護するために、インバウンド、アウトバウンド、インスペクションの各 VPC を使用してネットワークアカウントを設定することを推奨しています。

P

アクセス許可の境界

ユーザーまたはロールが使用できるアクセス許可の上限を設定する、IAM プリンシパルにアタッチされる IAM 管理ポリシー。詳細については、IAM ドキュメントの[アクセス許可の境界](#)を参照してください。

個人を特定できる情報 (PII)

直接閲覧した場合、または他の関連データと組み合わせた場合に、個人の身元を合理的に推測するために使用できる情報。PII の例には、氏名、住所、連絡先情報などがあります。

PII

「[個人を特定できる情報](#)」を参照してください。

プレイブック

クラウドでのコアオペレーション機能の提供など、移行に関連する作業を取り込む、事前定義された一連のステップ。プレイブックは、スクリプト、自動ランブック、またはお客様のモダナイズされた環境を運用するために必要なプロセスや手順の要約などの形式をとることができます。

PLC

「[プログラマブルロジックコントローラー](#)」を参照してください。

PLM

「[製品ライフサイクル管理](#)」を参照してください。

ポリシー

次の操作を可能にするオブジェクト: アクセス許可を定義する ([ID ベースのポリシー](#)を参照)。アクセス条件を指定する ([リソースベースのポリシー](#)を参照)。AWS Organizations の組織における全アカウントにアクセス許可の上限を定義する ([サービスコントロールポリシー](#)を参照)。

多言語の永続性

データアクセスパターンやその他の要件に基づいて、マイクロサービスのデータストレージテクノロジーを個別に選択します。マイクロサービスが同じデータストレージテクノロジーを使用している場合、実装上の問題が発生したり、パフォーマンスが低下する可能性があります。マイクロサービスは、要件に最も適合したデータストアを使用すると、より簡単に実装でき、パフォーマンスとスケーラビリティが向上します。

ポートフォリオ評価

移行を計画するために、アプリケーションポートフォリオの検出、分析、優先順位付けを行うプロセス。詳細については、「[移行の準備状況の評価](#)」を参照してください。

述語

true または false を返すためのクエリ条件。一般的に、WHERE 句に記述されます。

述語プッシュダウン

データベースクエリを最適化する手法。これによって、転送前にクエリ内のデータをフィルタリングします。この手法を取ると、リレーショナルデータベースから取得し処理する必要のあるデータの量が減少するため、クエリのパフォーマンスが向上します。

予防的コントロール

イベントの発生を防ぐように設計されたセキュリティコントロール。このコントロールは、ネットワークへの不正アクセスや好ましくない変更を防ぐ最前線の防御です。詳細については、「AWSでのセキュリティコントロールの実装」の「[予防的コントロール](#)」を参照してください。

プリンシパル

アクションを実行し AWS、リソースにアクセスできるエンティティ。このエンティティは通常、IAM AWS アカウントロール、またはユーザーのルートユーザーです。詳細については、IAM ドキュメントの「[ロールに関する用語と概念](#)」にあるプリンシパルを参照してください。

プライバシーバイデザイン

開発プロセス全体を通してプライバシーが考慮されているシステムエンジニアリングのアプローチ。

プライベートホストゾーン

1 つ以上の VPC 内のドメインとそのサブドメインへの DNS クエリに対し、Amazon Route 53 がどのように応答するかに関する情報を保持するコンテナ。詳細については、Route 53 ドキュメントの「[プライベートホストゾーンの使用](#)」を参照してください。

プロアクティブコントロール

非準拠リソースのデプロイ防止を目的とした[セキュリティコントロール](#)。このコントロールにより、プロビジョニング前にリソースをスキャンします。コントロールに準拠していないリソースは、プロビジョニングされません。詳細については、AWS Control Tower ドキュメントの「[コントロールリファレンスガイド](#)」および「[セキュリティコントロールの実装](#)」の「[プロアクティブコントロール](#)」を参照してください。 AWS

製品ライフサイクル管理 (PLM)

製品の設計、開発、発売から、成長、成熟、衰退、廃棄に至る、製品のライフサイクル全体を通してデータとプロセスを管理すること。

本番環境

「[環境](#)」を参照してください。

プログラマブルロジックコントローラー (PLC)

製造分野で使用される、信頼性と適応性に優れたコンピュータであり、これによって、マシンをモニタリングするとともに、製造プロセスを自動化します。

プロンプトチェイニング

1 つの [LLM](#) プロンプトによる出力を次のプロンプトの入力に使用して、より良いレスポンスを生成します。この手法を使用すると、複雑なタスクをサブタスクに分割したり、事前レスポンスを繰り返し改良または拡張したりできます。これによって、モデルのレスポンスの精度と関連性が向上し、粒度の高いパーソナライズされた結果を得られます。

仮名化

データセット内の個人識別子をプレースホルダー値に置き換えるプロセス。仮名化は個人のプライバシー保護に役立ちます。仮名化されたデータは、依然として個人データとみなされます。

発行/サブスクライブ (pub/sub)

マイクロサービス間の非同期通信を可能にするパターン。これにより、スケーラビリティと応答性を向上させます。例えば、マイクロサービスベースの [MES](#) の場合、マイクロサービスは、他のマイクロサービスがサブスクライブ可能なチャンネルにイベントメッセージを発行できます。このシステムでは、発行サービスの変更なしに、新規マイクロサービスを追加できます。

Q

クエリプラン

手順などの一連のステップであり、SQL リレーショナルデータベースシステムのデータにアクセスするために使用されます。

クエリプランのリグレッション

データベースサービスのオプティマイザーが、データベース環境に特定の変更が加えられる前に選択されたプランよりも最適性の低いプランを選択すること。これは、統計、制限事項、環境設定、クエリパラメータのバインディングの変更、およびデータベースエンジンの更新などが原因である可能性があります。

R

RACI マトリックス

「[実行責任者、説明責任者、協業先、報告先 \(RACI\)](#)」を参照してください。

RAG

「[検索拡張生成](#)」を参照してください。

ランサムウェア

決済が完了するまでコンピュータシステムまたはデータへのアクセスをブロックするように設計された、悪意のあるソフトウェア。

RASCI マトリックス

「[実行責任者、説明責任者、協業先、報告先 \(RACI\)](#)」を参照してください。

RCAC

「[行と列のアクセス制御](#)」を参照してください。

リードレプリカ

読み取り専用で使用されるデータベースのコピー。クエリをリードレプリカにルーティングして、プライマリデータベースへの負荷を軽減できます。

リアーキテクト

「[7 Rs](#)」を参照してください。

目標復旧時点 (RPO)

最後のデータリカバリポイントからの最大許容時間です。これにより、最後の回復時点からサービスが中断されるまでの間に許容できるデータ損失の程度が決まります。

目標復旧時間 (RTO)

サービスが中断から復旧までの最大許容遅延時間。

リファクタリング

「[7 Rs](#)」を参照してください。

リージョン

地理的エリア内の AWS リソースのコレクション。各 AWS リージョンは、耐障害性、安定性、耐障害性を提供するために、他のから分離され、独立しています。詳細については、「[アカウントが使用できる AWS リージョンを指定する](#)」を参照してください。

リグレッション

数値を予測する機械学習手法。例えば、「この家はどれくらいの値段で売れるでしょうか?」という問題を解決するために、機械学習モデルは、線形回帰モデルを使用して、この家に関する既知の事実 (平方フィートなど) に基づいて家の販売価格を予測できます。

リホスト

「[7 Rs](#)」を参照してください。

リリース

デプロイプロセスで、変更を本番環境に昇格させること。

再配置

「[7 Rs](#)」を参照してください。

リプラットフォーム

「[7 Rs](#)」を参照してください。

再購入

「[7 Rs](#)」を参照してください。

回復性

中断に抵抗または中断から回復するアプリケーションの機能。AWS クラウドでの回復力を計画する際には、一般的に、[高可用性](#)と[ディザスタリカバリ](#)が考慮されます。詳細については、「[AWS クラウドの耐障害性](#)」を参照してください。

リソースベースのポリシー

Amazon S3 バケット、エンドポイント、暗号化キーなどのリソースにアタッチされたポリシー。このタイプのポリシーは、アクセスが許可されているプリンシパル、サポートされているアクション、その他の満たすべき条件を指定します。

実行責任者、説明責任者、協業先、報告先 (RACI) に基づくマトリックス

移行活動とクラウド運用に関わるすべての関係者の役割と責任を定義したマトリックス。マトリックスの名前は、マトリックスで定義されている責任の種類、すなわち責任 (R)、説明責任 (A)、協議 (C)、情報提供 (I) に由来します。サポート (S) タイプはオプションです。サポートが含まれる場合は RASCI マトリックスと呼ばれ、含まれない場合は RACI マトリックスと呼ばれます。

レスポンスコントロール

有害事象やセキュリティベースラインからの逸脱について、修復を促すように設計されたセキュリティコントロール。詳細については、「AWSでのセキュリティコントロールの実装」の「[レスポンスコントロール](#)」を参照してください。

保持

「[7 Rs](#)」を参照してください。

廃止

「[7 Rs](#)」を参照してください。

検索拡張生成 (RAG)

[生成 AI](#) の技術。これにより、[LLM](#) では、レスポンスの生成前に、トレーニングデータソースの外部にある信頼できるデータソースが参照されます。例えば、RAG モデルによって、組織のナレッジベースまたはカスタムデータのセマンティック検索を実行できる場合があります。細については、「[RAG \(検索拡張生成\) とは何ですか?](#)」を参照してください。

ローテーション

定期的に[シークレット情報](#)を更新して、攻撃者が認証情報にアクセスするのをより困難にするプロセス。

行と列のアクセス制御 (RCAC)

アクセスルールが定義された、基本的で柔軟な SQL 表現の使用。RCAC は行権限と列マスクで構成されています。

RPO

「[目標復旧時点](#)」を参照してください。

RTO

「[目標復旧時間](#)」を参照してください。

ランブック

特定のタスクを実行するために必要な手動または自動化された一連の手順。これらは通常、エラー率の高い反復操作や手順を合理化するために構築されています。

S

SAML 2.0

多くの ID プロバイダー (IdP) が使用しているオープンスタンダード。この機能を使用すると、フェデレーテッドシングルサインオン (SSO) が有効になるため、ユーザーは組織内のすべてのユーザーを IAM で作成しなくても、AWS マネジメントコンソールにログインしたり AWS、API オペレーションを呼び出すことができます。SAML 2.0 ベースのフェデレーションの詳細については、IAM ドキュメントの「[SAML 2.0 ベースのフェデレーションについて](#)」を参照してください。

SCADA

「[監視制御とデータ取得](#)」を参照してください。

SCP

「[サービスコントロールポリシー](#)」を参照してください。

シークレット

暗号化された形式で保存するパスワードやユーザー認証情報などの AWS Secrets Manager 機密情報または制限付き情報。シークレット値とそのメタデータで構成されます。シークレット値には、バイナリ、1 つの文字列、複数の文字列を指定できます。詳細については、Secrets Manager ドキュメントの「[Secrets Manager シークレットの概要](#)」を参照してください。

セキュリティバイデザイン

開発プロセス全体を通してセキュリティが考慮されているシステムエンジニアリングのアプローチ。

セキュリティコントロール

脅威アクターによるセキュリティ脆弱性の悪用を防止、検出、軽減するための、技術上または管理上のガードレール。セキュリティコントロールには、主に 4 つの種類があります。4 つとは、[予防](#)、[検出](#)、[レスポンス](#)、[プロアクティブ](#)です。

セキュリティ強化

アタックサーフェスを狭めて攻撃への耐性を高めるプロセス。このプロセスには、不要になったリソースの削除、最小特権を付与するセキュリティのベストプラクティスの実装、設定ファイル内の不要な機能の無効化、といったアクションが含まれています。

Security Information and Event Management (SIEM) システム

セキュリティ情報管理 (SIM) とセキュリティイベント管理 (SEM) のシステムを組み合わせたツールとサービス。SIEM システムは、サーバー、ネットワーク、デバイス、その他ソースからデータを収集、モニタリング、分析して、脅威やセキュリティ違反を検出し、アラートを発信します。

セキュリティレスポンスの自動化

セキュリティイベントへの自動レスポンスまたは自動修復を目的として、事前定義およびプログラムされたアクション。これらの自動化は、セキュリティのベストプラクティスを実装するのに役立つ[検出的](#)または[応答的](#)な AWS セキュリティコントロールとして機能します。自動レスポンスアクションの例には、VPC セキュリティグループの変更、Amazon EC2 インスタンスへのパッチ適用、認証情報の更新などがあります。

サーバー側の暗号化

送信先にあるデータの、それ AWS のサービスを受け取る による暗号化。

サービスコントロールポリシー (SCP)

AWS Organizations の組織内の、すべてのアカウントのアクセス許可を一元的に管理するポリシー。SCP は、管理者がユーザーまたはロールに委任するアクションに、ガードレールを定義したり、アクションの制限を設定したりします。SCP は、許可リストまたは拒否リストとして、許可または禁止するサービスやアクションを指定する際に使用できます。詳細については、AWS Organizations ドキュメントの「[サービスコントロールポリシー](#)」を参照してください。

サービスエンドポイント

のエンドポイントの URL AWS のサービス。ターゲットサービスにプログラムで接続するには、エンドポイントを使用します。詳細については、「AWS 全般のリファレンス」の「[AWS のサービス エンドポイント](#)」を参照してください。

サービスレベルアグリーメント (SLA)

サービスのアップタイムやパフォーマンスなど、IT チームがお客様に提供すると約束したものを明示した合意書。

サービスレベルインジケータ (SLI)

エラー率、可用性、スループットといった、サービスパフォーマンス面の指標。

サービスレベル目標 (SLO)

[サービスレベルインジケータ](#)によって測定され、サービスの状態を表すターゲットメトリクス。

責任共有モデル

クラウドのセキュリティとコンプライアンス AWS についてと共有する責任を説明するモデル。AWS はクラウドのセキュリティを担当しますが、お客様はクラウドのセキュリティを担当します。詳細については、「[責任共有モデル](#)」を参照してください。

SIEM

「[Security Information and Event Management システム](#)」を参照してください。

単一障害点 (SPOF)

特定のアプリケーションを構成する単一の重要なコンポーネントで発生し、システム稼働に支障をきたす可能性のある障害。

SLA

「[サービスレベルアグリーメント](#)」を参照してください。

SLI

「[サービスレベルインジケータ](#)」を参照してください。

SLO

「[サービスレベルの目標](#)」を参照してください。

スプリットアンドシードモデル

モダナイゼーションプロジェクトのスケーリングと加速のためのパターン。新機能と製品リリースが定義されると、コアチームは解放されて新しい製品チームを作成します。これにより、お客様の組織の能力とサービスの拡張、デベロッパーの生産性の向上、迅速なイノベーションのサポートに役立ちます。詳細については、「[AWS クラウドでのアプリケーションをモダナイズするための段階的アプローチ](#)」を参照してください。

SPOF

「[単一障害点](#)」を参照してください。

スタースキーマ

データベースの編成構造を意味し、1つの大きいファクトテーブルにトランザクションデータまたは測定データが保存され、1つ以上の小さいディメンションテーブルにデータ属性が保存されます。この構造は、[データウェアハウス](#)やビジネスインテリジェンスを用途とするように設計されています。

strangler fig パターン

レガシーシステムが廃止されるまで、システム機能を段階的に書き換えて置き換えることにより、モノリシックシステムをモダナイズするアプローチ。このパターンは、宿主の樹木から根を成長させ、最終的にその宿主を包み込み、宿主に取って代わるイチジクのつるを例えています。そのパターンは、モノリシックシステムを書き換えるときのリスクを管理する方法として [Martin Fowler](#) により提唱されました。このパターンの適用方法の例については、「[コンテナと Amazon API Gateway を使用して、従来の Microsoft ASP.NET \(ASMX\) ウェブサービスを段階的にモダナイズ](#)」を参照してください。

サブネット

VPC 内の IP アドレスの範囲。サブネットは、1つのアベイラビリティゾーンに存在する必要があります。

監視制御とデータ取得 (SCADA)

製造分野において、ハードウェアとソフトウェアを使用して物理アセットと本番運用をモニタリングするシステム。

対称暗号化

データの暗号化と復号に同じキーを使用する暗号化のアルゴリズム。

合成テスト

ユーザーとのやり取りをシミュレートして、起こり得る問題を検出したり、パフォーマンスをモニタリングしたりすることで、システムをテストします。[Amazon CloudWatch Synthetics](#) を使用すると、こうしたテストを作成できます。

システムプロンプト

コンテキスト、指示、ガイドラインなどを提示して、[LLM](#) に動作を指示する手法。システムプロンプトは、コンテキストを設定して、ユーザーとやり取りするルールを確立するのに有用です。

T

タグ

AWS リソースを整理するためのメタデータとして機能するキーと値のペア。タグは、リソースの管理、識別、整理、検索、フィルタリングに役立ちます。詳細については、「[AWS リソースのタグ付け](#)」を参照してください。

ターゲット変数

監督された機械学習でお客様が予測しようとしている値。これは、結果変数のことも指します。例えば、製造設定では、ターゲット変数が製品の欠陥である可能性があります。

タスクリスト

ランブックの進行状況を追跡するために使用されるツール。タスクリストには、ランブックの概要と完了する必要がある一般的なタスクのリストが含まれています。各一般的なタスクには、推定所要時間、所有者、進捗状況が含まれています。

テスト環境

「[環境](#)」を参照してください。

トレーニング

お客様の機械学習モデルに学習するデータを提供すること。トレーニングデータには正しい答えが含まれている必要があります。学習アルゴリズムは入力データ属性をターゲット (お客様が予測したい答え) にマッピングするトレーニングデータのパターンを検出します。これらのパターンをキャプチャする機械学習モデルを出力します。そして、お客様が機械学習モデルを使用して、ターゲットがわからない新しいデータでターゲットを予測できます。

トランジットゲートウェイ

VPC とオンプレミスネットワークを相互接続するために使用できる、ネットワークの中継ハブ。詳細については、AWS Transit Gateway ドキュメントの「[トランジットゲートウェイとは](#)」を参照してください。

トランクベースのワークフロー

デベロッパーが機能ブランチで機能をローカルにビルドしてテストし、その変更をメインブランチにマージするアプローチ。メインブランチはその後、開発環境、本番前環境、本番環境に合わせて順次構築されます。

信頼されたアクセス

ユーザーに代わって AWS Organizations およびそのアカウントで組織内でタスクを実行するために指定したサービスにアクセス許可を付与します。信頼されたサービスは、サービスにリンクされたロールを必要なときに各アカウントに作成し、ユーザーに代わって管理タスクを実行します。詳細については、ドキュメントの「[を他の AWS のサービス AWS Organizations で使用する AWS Organizations](#)」を参照してください。

チューニング

機械学習モデルの精度を向上させるために、お客様のトレーニングプロセスの側面を変更する。例えば、お客様が機械学習モデルをトレーニングするには、ラベル付けセットを生成し、ラベルを追加します。これらのステップを、異なる設定で複数回繰り返して、モデルを最適化します。

ツーピザチーム

2 枚のピザを分け合えることができるくらい小さな DevOps チーム。ツーピザチームの規模では、ソフトウェア開発におけるコラボレーションに最適な機会が確保されます。

U

不確実性

予測機械学習モデルの信頼性を損なう可能性がある、不正確、不完全、または未知の情報を指す概念。不確実性には、次の 2 つのタイプがあります。認識論的不確実性は、限られた、不完全なデータによって引き起こされ、弁論的不確実性は、データに固有のノイズとランダム性によって引き起こされます。詳細については、[深層学習システムにおける不確実性の定量化ガイド](#)を参照してください。

未分化なタスク

ヘビーリフティングとも呼ばれ、アプリケーションの作成と運用には必要だが、エンドユーザーに直接的な価値をもたらさなかったり、競争上の優位性をもたらしたりしない作業です。未分化なタスクの例としては、調達、メンテナンス、キャパシティプランニングなどがあります。

上位環境

「[環境](#)」を参照してください。

V

バキューミング

ストレージを再利用してパフォーマンスを向上させるために、増分更新後にクリーンアップを行うデータベースのメンテナンス操作。

バージョンコントロール

リポジトリ内のソースコードへの変更など、変更を追跡するプロセスとツール。

VPC ピアリング

プライベート IP アドレスを使用してトラフィックをルーティングできる、2 つの VPC 間の接続。詳細については、Amazon VPC ドキュメントの「[VPC ピア機能とは](#)」を参照してください。

脆弱性

システムのセキュリティを脅かすソフトウェアまたはハードウェアの欠陥。

W

ウォームキャッシュ

頻繁にアクセスされる最新の関連データを含むバッファキャッシュ。データベースインスタンスはバッファキャッシュから、メインメモリまたはディスクからよりも短い時間で読み取りを行うことができます。

ウォームデータ

アクセス頻度の低いデータ。この種類のデータをクエリする場合、通常は適度に遅いクエリでも問題ありません。

ウィンドウ関数

現在のレコードに何らかの形で関連している行のグループに計算を実行する SQL 関数。ウィンドウ関数は、移動平均を計算したり、現在の行の相対位置に基づいて他の行の値にアクセスするといったタスクの処理に役立ちます。

ワークロード

ビジネス価値をもたらすリソースとコード (顧客向けアプリケーションやバックエンドプロセスなど) の総称。

ワークストリーム

特定のタスクセットを担当する移行プロジェクト内の機能グループ。各ワークストリームは独立していますが、プロジェクト内の他のワークストリームをサポートしています。たとえば、ポートフォリオワークストリームは、アプリケーションの優先順位付け、ウェーブ計画、および移行メタデータの収集を担当します。ポートフォリオワークストリームは、これらの設備を移行ワークストリームで実現し、サーバーとアプリケーションを移行します。

WORM

「[Write-Once-Read-Many](#)」を参照してください。

WQF

「[AWS ワークロード資格フレームワーク](#)」を参照してください

Write-Once-Read-Many (WORM)

データを 1 回のみ書き込むことで、データの削除や変更を防ぐストレージモデル。承認済みユーザーは、必要な回数だけデータを読み取ることができますが、変更することはできません。このデータストレージインフラストラクチャは、[イミュータブル](#)と見なされます。

Z

ゼロデイエクスプロイト

[ゼロデイ脆弱性](#)を悪用した攻撃（一般的にマルウェアによる）。

ゼロデイ脆弱性

実稼働システムにおける未解決の欠陥または脆弱性。脅威アクターは、このような脆弱性を利用してシステムを攻撃する可能性があります。開発者は、よく攻撃の結果で脆弱性に気付きます。

ゼロショットプロンプト

[LLM](#) にタスク実行の手順は提示するが、実行のガイドとして役立つ例（ショット）は提示しない方法。LLM は、事前トレーニング済みの知識を使用してタスクを処理する必要があります。ゼロショットプロンプトの有効性は、タスクの複雑さとプロンプトの品質によって異なります。「[数ショットプロンプト](#)」も参照してください。

ゾンビアプリケーション

平均 CPU およびメモリ使用率が 5% 未満のアプリケーション。移行プロジェクトでは、これらのアプリケーションを廃止するのが一般的です。

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。