



Amazon OpenSearch Service への移行

# AWS 規範ガイドンス



# AWS 規範ガイド: Amazon OpenSearch Service への移行

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
概要 .....	1
OpenSearch Service のメリット .....	3
デプロイと管理が簡単 .....	3
コスト効率が良い .....	3
よりスケーラブルで信頼性が高い .....	4
安全性とコンプライアンス準拠 .....	4
移行ジャーニー .....	5
計画 .....	6
サイジング .....	6
Storage .....	6
ノードの数とインスタンスタイプ .....	8
インデックス作成戦略とシャード数の決定 .....	8
CPU 使用率 .....	9
インスタンスのタイプ .....	9
機能 .....	10
現在のソリューションの機能 .....	11
Amazon OpenSearch Service の機能 .....	11
パッケージ化されたプラグイン .....	11
カスタムプラグイン .....	12
バージョン依存性 .....	12
エンジンバージョンの選択 .....	12
最新の OpenSearch Service バージョンへのアップグレード .....	13
バージョンアップグレード戦略 .....	13
アップグレード前のチェック .....	14
KPI とビジネス継続性 .....	14
運用パフォーマンス .....	15
プロセスのパフォーマンス .....	16
新しいサービスへのスムーズな移行 .....	16
財務メトリクス .....	16
運用とセキュリティ .....	17
ランブックと新しいプロセス .....	17
サポートとチケット発行システム .....	18
セキュリティ .....	18

トレーニング .....	19
トレーニングオプション .....	19
データフロー .....	20
データインGEST .....	21
データ保持 .....	22
データ移行アプローチ .....	22
デプロイフレームワーク .....	24
概念実証 .....	26
開始条件と終了条件の定義 .....	26
資金の確保 .....	26
自動化 .....	27
入念なテスト .....	27
PoC ステージ .....	28
障害シミュレーション .....	29
デプロイ .....	30
データ移行 .....	31
スナップショットから構築する .....	31
スナップショットに関する考慮事項 .....	32
ソースから構築する .....	33
リモート再インデックス .....	34
Logstash を使用する .....	34
カットオーバー .....	36
データ同期 .....	36
スワップまたはカットオーバー .....	40
オペレーショナルエクセレンス .....	41
結論 .....	42
リソース .....	43
寄稿者 .....	44
ドキュメント履歴 .....	45
用語集 .....	46
# .....	46
A .....	47
B .....	49
C .....	51
D .....	54
E .....	58

---

F .....	61
G .....	62
H .....	63
I .....	65
L .....	67
M .....	68
O .....	72
P .....	75
Q .....	78
R .....	78
S .....	81
T .....	85
U .....	86
V .....	87
W .....	87
Z .....	88
.....	lxxxix

# Amazon OpenSearch Service への移行

Amazon Web Services ([寄稿者](#))

2023 年 8 月 ([ドキュメント履歴](#))

多くのお客様にとって、セルフマネージド型の Elasticsearch または OpenSearch デプロイを [Amazon OpenSearch Service](#) に移行するのは難しい作業です。よくある課題として、ワークロード評価、キャパシティプランニング、アーキテクチャの最適化があります。また、Amazon Web Services (AWS) クラウドでオンプレミスデータセンターの運用分析アプリケーションのすべての要件をどのように満たすのかという疑問もあります。このガイドでは、Amazon OpenSearch Service への移行の全体的なジャーニーについて説明し、AWS のエキスパートが長い時間をかけて蓄積したベストプラクティスを提供します。詳細な手順に従うことで、効果的かつ効率的なアプローチで移行を実行することができます。このガイドでは、Amazon OpenSearch Serverless コレクションではなく、Amazon OpenSearch Service でプロビジョニングされたドメインについて主に説明します。

## 概要

[OpenSearch](#) は、リアルタイムのアプリケーションモニタリング、ログ分析、データオブザーバビリティ、アプリケーションと製品カタログの検索など、さまざまな運用分析のユースケースに使用される、オープンソースの分散型検索および分析スイートです。OpenSearch は低レイテンシーの検索レスポンスを提供します。また、OpenSearch Dashboards と呼ばれる統合されたオープンソースのデータ可視化ツールを使用して、大量のデータにすばやくアクセスできます。

Amazon OpenSearch Service では、インタラクティブなログ分析、リアルタイムのアプリケーションモニタリング、ウェブサイト検索などの実行をサポートします。Amazon OpenSearch Service は、OpenSearch の最新バージョンを提供し、Elasticsearch の 19 個のバージョン (バージョン 1.5 ~ 7.10) をサポートしています。また、OpenSearch Dashboards と Kibana (バージョン 1.5 ~ 7.10) を利用した視覚化機能も提供します。Amazon OpenSearch Service は現在、数万ものアクティブ顧客が使用しており、数十万のクラスターが 1 か月あたり数十兆のリクエストを処理しています。

オンプレミスまたはクラウドインフラストラクチャで OpenSearch または Elasticsearch クラスターを管理することは、非常に複雑でコストが高く、面倒な作業です。これらのクラスターを実行するには、インフラストラクチャをプロビジョニングして維持する必要があります。この作業には以下が含まれます。

- ハードウェアの調達とセットアップ

- ソフトウェアのインストール
- 設定、パッチ適用、アップグレード
- 信頼性と可用性に関する考慮事項
- パフォーマンスとスケーラビリティに関する考慮事項
- セキュリティとコンプライアンスに関する考慮事項 (ネットワーク分離、きめ細かなアクセスコントロール、暗号化、次のようなコンプライアンスプログラムなど):
  - Federal Risk and Authorization Management Program (FedRAMP)
  - 一般データ保護規則 (GDPR)
  - Health Insurance Portability and Accountability Act (HIPAA)
  - 国際標準化機構 ( ISO )
  - Payment Card Industry Data Security Standard (PCI DSS)
  - System and Organization Controls (SOC)

これに対して、Amazon OpenSearch Service はこれらのタスクを自動的に管理します。このガイドでは、オンプレミスまたはセルフマネージド型の Elasticsearch または OpenSearch をフルマネージド型の Amazon OpenSearch Service に移行するためのアプローチとベストプラクティスについて説明します。

# Amazon OpenSearch Service への移行のメリット

Amazon OpenSearch Service は、デプロイと継続的な管理タスクに役立ちます。費用対効果が高く、スケーラビリティを提供するため、信頼性が向上します。また、セキュリティを提供し、コンプライアンスのニーズをサポートするのに役立ちます。

## デプロイと管理が簡単

Amazon OpenSearch Service を使用して OpenSearch クラスターをデプロイする方が、クラスターを自分でデプロイするよりも簡単です。Amazon OpenSearch Service は、ハードウェアのプロビジョニング、ソフトウェアのインストールとパッチ適用、障害復旧、バックアップ、モニタリングなどのタスクの管理を支援します。クラスターを管理するために OpenSearch 専任のエキスパートチームを組む必要はありません。

Amazon OpenSearch Service の OpenSearch クラスターは、ドメインとも呼ばれます。Amazon OpenSearch Service は、Amazon CloudWatch サービスを通じてドメインのヘルスマニタリングを提供します。ドメインの状態が変化したときに通知するアラートを設定できます。AWS サポートは、経験豊富なエンジニアによる一対一のテクニカルサポートを提供します。運用上の課題や技術的な質問があるお客様は、AWS サポートに連絡すると、信頼できる応答時間でパーソナライズされたサポートを受けることができます。

## コスト効率が良い

Amazon OpenSearch Service はコスト効率に優れています。追加のライセンス料金を請求することなく、高度な機能を取り揃えています。エンタープライズグレードのセキュリティ、リアルタイムアラート、クラスター間検索、自動インデックス管理、異常検出などの機能を追加料金なしで使用できます。アベイラビリティゾーン間のデータ転送には料金はかかりません。時間単位のスナップショットは追加料金なしで提供されます。

UltraWarm を使用すると、ホットストレージ層と比較して GB あたりのコストを最大 90% 削減しながら、最大 3 ペタバイトのログデータに対してインタラクティブな分析を実行できます。さらに、Amazon OpenSearch Service が提供するリザーブドインスタンス (RI) では、標準オンデマンドインスタンスと比べて大幅な割引を受けられます。詳細については、「[Cost-conscious](#)」を参照してください。

## よりスケーラブルで信頼性が高い

Amazon OpenSearch Service を使用すると、ペタバイトのデータを単一のドメインに保存できます。複数のドメインにまたがるデータをクエリし、単一の OpenSearch Dashboards インターフェイスですべてのデータを分析できます。Amazon OpenSearch Service は、マルチアベイラビリティゾーン (マルチ AZ) 配置を使用して信頼性を高めるように設計されているため、同じ AWS リージョン内の最大 3 つのアベイラビリティゾーン間でデータをレプリケートできます。ソフトウェアの更新やアップグレード、または環境のスケーリング時にダウンタイムは発生しません。

スタンバイ付きマルチ AZ 機能により、OpenSearch Service ドメインは、ノードやアベイラビリティゾーンの障害など、潜在的なインフラストラクチャ障害に対する回復力を備えています。これにより、ビジネスクリティカルなワークロードに対して 99.99% の可用性と一貫したパフォーマンスを実現できます。スタンバイ付きマルチ AZ では、クラスターはハードウェアやネットワーク障害などのインフラストラクチャ障害に対して回復力があります。このオプションにより信頼性が向上し、ベストプラクティスを適用して複雑さを軽減すると、クラスターの設定と管理が簡素化するという利点が追加されます。

## 安全性とコンプライアンス準拠

Amazon OpenSearch Service は、すべてのセキュリティパッチを処理します。また、仮想プライベートクラウド (VPC) によるネットワーク分離、きめ細かなアクセスコントロール、マルチテナント OpenSearch Dashboards サポートも提供します。保管中のデータと転送中のデータを暗号化できます。業界固有の要件と規制要件を満たすために、Amazon OpenSearch Service は HIPAA に適格で、次の規格に準拠しています。

- FedRAMP
- GDPR
- PCI DSS
- ISO
- SOC

詳細については、[Amazon OpenSearch Service ドキュメント](#)を参照してください。

# 移行ジャーニー

現在のデプロイに応じて、Amazon OpenSearch Service への移行は基本的な手順となることも、複数のステップを含む複雑な手順となることもあります。以下のセクションでは、プロセスの各ステップにおける移行アプローチと重要な考慮事項について説明します。これには、多くのお客様が既存のツールから Amazon OpenSearch Service に移行できるよう支援してきた AWS の経験に基づくベストプラクティスが含まれます。このセクションでは、効果的な移行戦略の構成要素についても説明します。

一般的な移行ジャーニーには、次の 5 つの段階があります。

1. 計画
2. 概念実証 (PoC)
3. デプロイ
4. データ移行
5. カットオーバー

セルフマネージド型の Elasticsearch または OpenSearch クラスターから移行する場合も、別のテクノロジーから Amazon OpenSearch Service に移行する場合もあるでしょう。ほとんどのケースで、手順は変わりません。各ステップにかかる時間は、環境の複雑さによって異なります。

移行ジャーニーは、慎重な計画アクティビティから始まり、その後に PoC 演習を行って、ターゲット環境がコスト、セキュリティ、パフォーマンス、移行目標を確実に満たすようにします。PoC アクティビティの後、ターゲット環境をデプロイし、その環境にデータを移行します。データが現在の環境と新しい環境の間で同期されていることを確認したら、新しい環境にカットオーバーできます。カットオーバー後、運用上のベストプラクティスに従って環境を運用します。以下のセクションでは、各ステージについて詳しく説明します。

# ステージ 1 – 計画

移行は、要件を満たすようにターゲット環境をどのように構築するかを計画するところから始まります。計画には、一連の重点領域を調査するステップが含まれます。各領域は慎重に検討する必要があります。

- [サイジング](#)
- [機能](#)
- [バージョン依存性](#)
- [主要業績評価指標 \(KPI\) とビジネス継続性](#)
- [運用とセキュリティ](#)
- [トレーニング](#)
- データフロー
- [デプロイフレームワーク](#)

これらの重点領域は、判断を下して移行戦略を策定するのに役立ちます。また、移行の複雑さとコストを削減することで、移行目標の達成にも役立ちます。

計画段階では、現在の環境を評価し、この移行の一環として対処したい問題点を特定することも重要です。こうした問題点は、パフォーマンス、セキュリティ、信頼性、提供速度、コスト、運用のしやすさに関するものが考えられます。重点領域を確認するときは、移行の一環としてどのような改善を実現できるかを検討してください。

## サイジング

サイジングは、ターゲット環境に適したインスタンスタイプ、データノードの数、ストレージ要件を決定するのに役立ちます。最初にストレージのサイズを指定し、次に CPU のサイズを指定することをお勧めします。Elasticsearch または OpenSearch を既に使用している場合、通常はサイズを変更しません。ただし、現在の環境と同等のインスタンスタイプを特定する必要があります。適切なサイズを決定するには、次のガイドラインを使用することをお勧めします。

### Storage

クラスターのサイズ指定は、ストレージ要件の定義から始まります。クラスターに必要な raw ストレージを特定します。これは、ソースシステムによって生成されたデータ (ログを生成するサーバー、製品カタログの raw サイズなど) を評価することによって判断します。raw データの量を特定

したら、次の式を使用してストレージ要件を計算します。その後、結果を PoC の開始点として使用できます。

$$\text{storage needed} = (\text{daily source data in bytes} \times 1.45) (\text{number\_of\_replicas} + 1) \times \text{number of days retained}$$

この式では、以下が考慮されます。

- ディスク上のインデックスのサイズは一定ではありませんが、多くの場合でソースデータよりも 10% 大きくなります。
- オペレーティングシステムのオーバーヘッド 5% は、システム復旧とディスクのデフラグの問題から保護するために Linux によって予約されています。
- OpenSearch は、セグメントマージ、ログ、およびその他の内部オペレーション用として、インスタンスごとにストレージ容量の 20% を予約します。
- ノード障害やアベイラビリティゾーンの停止による影響を最小限に抑えるために、10% の追加ストレージを維持することをお勧めします。

これらのオーバーヘッドと予約を組み合わせると、ソース内の実際の raw データをベースとして 45% の追加領域が必要になります。そのため、ソースデータに 1.45 を掛けます。次に、これをデータのコピー数 (1 つのプライマリと使用するレプリカの数など) で乗算します。レプリカ数は、耐障害性とスループットの要件によって異なります。平均的なユースケースでは、1 つのプライマリと 1 つのレプリカから始めます。最後に、ホットストレージ階層にデータを保持する日数を掛けます。

Amazon OpenSearch Service は、ホット、ウォーム、コールドストレージ階層を提供します。ウォームストレージ階層は UltraWarm ストレージを使用します。UltraWarm は、大量の読み取り専用データをコストパフォーマンスに優れた方法で Amazon OpenSearch Service に保存できます。標準データノードはホットストレージを使用します。このストレージは、各ノードにアタッチされたインスタンスストアまたは Amazon Elastic Block Store (Amazon EBS) ボリュームの形をとります。ホットストレージは、新しいデータのインデックス作成と検索において、可能な限り高速なパフォーマンスを提供します。UltraWarm ノードは、Amazon Simple Storage Service (Amazon S3) をストレージとして使用し、パフォーマンスを向上させるために高度なキャッシュソリューションを使用します。アクティブに書き込みを行っていないインデックス、クエリの頻度が低いインデックス、および同じパフォーマンス要件が設定されていないインデックスについては、UltraWarm により、データの GiB あたりのコストが大幅に削減されます。UltraWarm の詳細については、[AWS ドキュメント](#)を参照してください。

OpenSearch Service ドメインを作成してホットストレージを使用する場合は、EBS ボリュームサイズの定義が必要なことがあります。これはデータノードにどのインスタンスタイプを選択し

たかによって異なります。同じストレージ要件の式を使用して、Amazon EBS backed インスタンスのボリュームサイズを決定できます。最新世代の T3、R5、R6G、M5、M5g、C5、および C6g インスタンスファミリーには、gp3 ボリュームを使用することをお勧めします。Amazon EBS gp3 ボリュームを使用すると、ストレージ容量に関係なくパフォーマンスをプロビジョニングできます。Amazon EBS gp3 ボリュームではベースラインパフォーマンスも向上し、OpenSearch Service の既存の gp2 ボリュームよりも GB あたりのコストが 9.6% 低くなります。gp3 を使用すると、R5、R6g、M5、M6g インスタンスファミリーでより高密度なストレージも得られるため、コストをさらに最適化できます。EBS ボリュームはサポートされているクォータまで作成できます。クォータの詳細については、「[Amazon OpenSearch Service クォータ](#)」を参照してください。

i3 インスタンスや r6gd インスタンスなどの NVM Express (NVMe) ドライブを持つデータノードの場合、ボリュームサイズは固定されているため、EBS ボリュームは選択できません。

## ノードの数とインスタンスタイプ

ノードの数は、ワークロードの運用に必要な CPU の数に基づきます。CPU の数はシャード数に基づきます。OpenSearch のインデックスは複数のシャードで構成されます。インデックスを作成するときに、インデックスのシャード数を指定します。そのため、以下のことを行う必要があります。

1. ドメインに保存するシャードの合計数を計算します。
2. CPU を決定します。
3. 最もコスト効率の高いノードのタイプと数を見つけたら、必要な CPU の数とストレージがわかります。

通常はこれが出発点となります。テストを実行して、見積もったサイズが機能要件と非機能要件を満たしていることを確認します。

## インデックス作成戦略とシャード数の決定

ストレージ要件がわかったら、必要なインデックスの数を決定し、それぞれのシャード数を特定できます。一般的に、検索ユースケースには 1 つまたはいくつかのインデックスがあり、それぞれが検索可能なエンティティまたはカタログを表します。ログ分析のユースケースでは、1 つのインデックスが毎日または毎週のログファイルを表す場合があります。インデックスの数を決定したら、次のスケールガイドを参考に、適切なシャード数を決定します。

- 検索ユースケース – 10 ~ 30 GB/シャード
- ログ分析ユースケース – 50 GB/シャード

1つのインデックスのデータの総量を、ユースケースで目指すシャードサイズで割ることができず。これにより、インデックスのシャード数がわかります。シャードの合計数を特定すると、ワークロードに適したインスタンスタイプを見つけるのに役立ちます。シャードが大きすぎたり多すぎたりしないようにしてください。シャードが大きいと、OpenSearch の障害復旧が困難になる可能性があります。一方、各シャードはある程度の CPU とメモリを使用するため、小規模なシャードが多数ある場合には、パフォーマンスの問題やメモリ不足のエラーが発生する可能性があります。さらに、データノードへのシャード割り当てが不均衡だと、歪みが生じる可能性があります。複数のシャードを持つインデックスがある場合は、シャードがデータノード数の偶数倍になるように試みます。これは、シャードがデータノード全体に均等に分散されるようにするのに役立ち、ノードがホットになるのを防ぎます。例えば、プライマリシャードが 12 個ある場合、データノード数は 2、3、4、6、または 12 になります。ただし、シャード数よりもシャードサイズの方が重要です。5 GiB のデータがある場合、1 個のシャードを使用するべきです。レプリカシャードの数をアベイラビリティゾーン間で均等に分散させると、耐障害性も向上します。

## CPU 使用率

次のステップでは、ワークロードに必要な CPU の数を特定します。CPU の数はアクティブなシャードの 1.5 倍から開始することをお勧めします。アクティブなシャードとは、大量の書き込みを受けているインデックスの任意のシャードです。プライマリシャード数を使用して、大量の読み取りまたは書き込みリクエストを受信しているインデックスのアクティブなシャードを決定します。ログ分析では、通常は現在のインデックスのみがアクティブです。検索ユースケースでは、すべてのプライマリシャードがアクティブなシャードと見なされます。アクティブなシャードあたり 1.5 倍の CPU をお勧めしますが、これはワークロードに大きく依存します。CPU 使用率をテストおよびモニタリングし、適宜スケールアップしてください。

CPU 使用率を維持するためのベストプラクティスは、OpenSearch サービスドメインにタスクを実行するのに十分なリソースがあることを確認することです。CPU 使用率が一貫して高いクラスターは、クラスターの安定性を低下させる可能性があります。クラスターが過負荷になると、OpenSearch Service は受信リクエストをブロックし、リクエストが拒否されます。これは、ドメインが失敗しないように保護するものです。CPU 使用率に関する一般的なガイドラインは平均で約 60%、最大 CPU 使用率は 80% です。一時的な 100% のスパイクも許容され、スケールアップや再設定は必要でない場合があります。

## インスタンスのタイプ

Amazon OpenSearch Service では、複数のインスタンスタイプを選択できます。ユースケースに合わせて最適なインスタンスタイプを選択できます。Amazon OpenSearch Service は、R、C、M、T、I インスタンスファミリーをサポートしています。インスタンスファミリーは、

メモリ最適化、コンピューティング最適化、またはそれらを組み合わせたワークロードに基づいて選択します。インスタンスファミリーを特定したら、最新世代のインスタンスタイプを選択します。一般的には、Graviton 以降の世代をお勧めします。前世代のインスタンスと比較して、パフォーマンスが向上し、コストが低くなるように構築されているためです。

ログ分析および検索ユースケースに関しては、それぞれに実行された各種テストに基づいて、以下をお勧めします。

- ログ分析ユースケースの場合、一般的なガイドラインとして、データノードには R ファミリーの [Graviton](#) インスタンスから始めます。テストを実行し、要件のベンチマークを確立し、ワークロードに適したインスタンスサイズを特定することをお勧めします。
- 検索ユースケースでは、ログ分析ユースケースよりも多くの CPU が必要になるため、データノードに R および C ファミリーの Graviton インスタンスを使用することをお勧めします。小規模なワークロードでは、検索とログの両方に M ファミリーの Graviton インスタンスを使用できます。I ファミリーインスタンスは NVMe ドライブを提供し、高速インデックス作成と低レイテンシーの検索を要件とするお客様が使用します。

クラスターは、データノードとクラスターマネージャーノードで構成されます。専用マスターノードでは検索およびクエリリクエストを処理しませんが、そのサイズは、管理可能なインスタンスサイズや、インスタンス数、インデックス数、シャード数と大きな相関があります。[AWS ドキュメントに記載されているマトリックス](#)で、推奨される最小限の専用クラスターマネージャーインスタンスのタイプがわかります。

AWS は、[AWS Graviton2](#) プロセッサを搭載した Amazon OpenSearch Service バージョン 7.9 以降向けに、汎用 (M6g)、コンピューティング最適化 (C6g)、メモリ最適化 (R6g および R6gd) を提供しています。これらのインスタンスは、Amazon によって設計されたカスタムシリコンを使用して構築されます。これらは、Amazon が設計したハードウェアおよびソフトウェアのイノベーションであり、分離されたマルチテナンシー、プライベートネットワーク、高速ローカルストレージを使用して、効率的で柔軟な、安全性の高いクラウドサービスを提供できます。

Graviton2 インスタンスファミリーは、OpenSearch Service (M5、C5、R5) で利用可能な前世代の Intel ベースのインスタンスと比較して、インデックス作成のレイテンシーを最大 50 % 削減し、クエリパフォーマンスを最大 30% 向上させます。

## 機能

機能の重点領域は、ターゲットの Amazon OpenSearch Service 環境に移行するときに機能が失われないようにするのに役立ちます。以下の点に細心の注意を払ってください。

- 現在のソリューションの機能
- Amazon OpenSearch Service の機能
- パッケージ化されたプラグイン

## 現在のソリューションの機能

現在のソリューションを分析し、現在のテクノロジースタックで使用する機能、プラグイン、API (Elasticsearch、OpenSearch、その他のソリューションなど)を確認することをお勧めします。事業にとって重要な機能、変更可能な機能、移行中に削除しても構わない機能を決定します。

## Amazon OpenSearch Service の機能

移行が終わった後に必要な機能を実際に利用できるようにするには、Amazon OpenSearch Service が提供する機能や Amazon OpenSearch Service で使用できるプラグインなど、Amazon OpenSearch Service でサポートされている最新の OpenSearch バージョンを分析することをお勧めします。ターゲットプラットフォームが必要な機能 (インデックスのロールオーバーを自動化するインデックス状態管理、異常検出などの機械学習機能など) をサポートしていることを確認します。ワークロードを引き続きサポートできるように、現在のソリューションの既存の機能を、同等の機能を提供する Amazon OpenSearch Service の機能にマッピングします。

サポートされている各バージョンの Elasticsearch または OpenSearch ソフトウェアで利用できる機能の詳細については、[Amazon OpenSearch Service のドキュメント](#)を参照してください。

## パッケージ化されたプラグイン

Amazon OpenSearch Service は、オープンソースの OpenSearch プロジェクトの一部である多数のプラグインをサポートしています。X-Pack またはその他の一部である Elasticsearch スイートからライセンスされたプラグインを使用している場合は、OpenSearch サービス内で同等のプラグインまたはネイティブ機能を確認できます。これは、PoC ステージでの実証ポイントとしてキャプチャすることもできます。

OpenSearch には、ライセンスされたプラグインと同等のエンタープライズグレードの機能を提供するプラグインがいくつかあります。ターゲット環境に適したプラグインとバージョンを確認するには、OpenSearch Service ドキュメントの[バージョン別のプラグイン](#)のリストを参照してください。Amazon OpenSearch Service は、すぐに使用できる多数の OpenSearch プラグインをサポートしていますが、使用しているオープンソースの OpenSearch プラグインの中には、Amazon OpenSearch Service で現在利用できないものがあるかもしれません。Amazon OpenSearch Service

の今後のロードマップにプラグインを追加するようリクエストするには、[AWS にお問い合わせください](#)。

## カスタムプラグイン

このガイドの作成時点で、カスタムプラグインはサポートされていません。したがって、カスタムプラグインの機能とエクスペリエンスを提供する代替的な方法を検討する必要があります。お使いのソリューションがカスタムプラグインを使用している場合は、その機能を分析して、Amazon OpenSearch Service がサポートするプラグインまたは OpenSearch 内のネイティブ機能を使用してカスタムプラグインをターゲット環境に移植できるかどうかを判断します。PoC ステージでは、すべてのプラグインの選択をテストして検証することをお勧めします。移行は、現在のソリューションの機能を評価して、それがビジネスにとって重要かどうかを判断する良い機会になります。

## バージョン依存性

バージョン依存関係の重点領域は、さまざまなバージョンを通して Amazon OpenSearch Service の最新バージョンに到達する、移行ジャーニーのロードマップを構築するのに役立ちます。次の重要な点を考慮してください。

- エンジンバージョンの選択
- 最新バージョンへのアップグレード
- バージョンアップグレード戦略
- アップグレード前のチェック

## エンジンバージョンの選択

バージョンの依存関係を慎重に検討することはとても重要です。Amazon OpenSearch Service は、多数の Elasticsearch バージョンとすべての主要な OpenSearch エンジンバージョンをサポートしています。(ただし、最新バージョンの OpenSearch が Amazon OpenSearch Service でサポートされるまでには、リリース日から数週間ほどかかる場合があります)。Amazon OpenSearch Service ドキュメントの[エンジンバージョンでサポートされている機能](#)を確認して、要件に適したバージョンを特定することをお勧めします。同じメジャー (および最も近いマイナー) バージョンを選択することで、[スナップショット復元アプローチ](#)を使用して移行できます。これは多くの場合、最も直接的なアプローチです。

## 最新の OpenSearch Service バージョンへのアップグレード

以前のバージョンの Amazon OpenSearch Service を運用できる可能性もありますが、利用可能な最新バージョンにアップグレードすることを強くお勧めします。これにより、最新バージョンのエンジンで導入されたパフォーマンスの向上、信頼性、コスト削減、および多くの新機能を活用できます。移行は、古いバージョンのソフトウェアを実行することで生じる技術的負債を軽減する良い機会です。

### バージョンアップグレード戦略

移行中に最新バージョンのソフトウェアにアップグレードすると決定した場合は、手順およびアップグレード戦略を策定します。Amazon OpenSearch Service ドキュメントには、[アップグレードパス](#)に関する情報が記載されています。バージョン間の重大な変更点を理解することが重要です。場合によっては、重大な変更により、インデックスのモデリングと設計を調整するための計画が必要になる場合があります。

#### Note

注: 複数のマッピングタイプの機能は、Elasticsearch バージョン 5.x 以前でのみ使用できます。バージョン 6.x 以降で作成されたインデックスは、インデックスごとに 1 つのマッピングタイプのみをサポートします。複数のマッピングタイプを使用している場合は、そのデータを複数のインデックスにリモデリングすることをお勧めします。

移行に時間的制約がある場合は、同等バージョンの移行 (5.x から 5.x など) を実行し、後から OpenSearch Service バージョンをアップグレードする基本オプションを検討してください。OpenSearch Service では、Elasticsearch バージョン 5.1 (互換性のある場合) 以降、および OpenSearch 1.0 以降を実行するドメインに対して、インプレースアップグレードを提供しています。Elasticsearch バージョン 5.x を実行している場合、テストを実行してインデックスがインプレースアップグレードに対応しているかどうかを確認してください。これは、同等バージョンに移行して、インデックスやその他の機能を最新バージョンと互換性を持たせるために必要な変更を加えた後に、インプレースアップグレードを実行できることを意味します [ドメインアップグレードドキュメント](#)をよくお読みください。

## アップグレード前のチェック

Amazon OpenSearch Service のアップグレード機能は、環境をスキャンして[アップグレード前のチェック](#)を実行し、アップグレードをブロックする可能性のある問題を特定することができます。これらのチェックが成功しない限り、アップグレードは次のステップに進みません。

## KPI とビジネス継続性

移行の間に、成功を測定するためのビジネス目標と主要業績評価指標 (KPIs) を確立することは不可欠です。移行プロセスの開始時に目標を決定し、現在のシステムのベースラインを確立することで、測定可能な改善内容を決定できるようにすることが重要です。カスタマージャーニーの一般的な目標は次のとおりです。

- 運用の俊敏性を向上させます。

この目標に基づき、次のメトリクスを使用して既存のデプロイを測定し、ターゲット環境と比較できます。

- クラスタをプロビジョニングするための平均所要時間
- 新しい地域にデプロイをロールアウトするための所要時間
- クラスタセキュリティを設定するための平均所要時間
- 環境をスケーリングするための平均所要時間 (ノードの追加やストレージの追加など)
- パフォーマンスの低いクエリを検出するための平均所要時間と、クエリを修復するための平均所要時間
- ソフトウェアバージョンをアップグレードするための平均所要時間
- 総保有コスト (TCO) を削減します。

現在の TCO を計算するには、次のメトリクスを使用できます。

- ソリューションの構築と運用にかかる工数 (開発、DevOps、モニタリング、スケーリング、バックアップ、復元)
- 既存のソフトウェアに関連するライセンスコスト
- データセンターのコスト (ハードウェアの調達と更新、電力、冷却、スペース、ラック、ネットワーク機器)
- ソリューションの設定にかかる工数 (ソフトウェアのインストール、ネットワーク設定)
- コンプライアンス監査コスト (HIPAA、PCI DSS、SOC、ISO、GDPR、FedRAMP)

- セキュリティの設定コスト (保管時および転送時の暗号化、認証と認可の設定、きめ細かなアクセスコントロール)
- 大量のウォームデータとコールドデータを保持するコスト
- アベイラビリティゾーン間で高可用性を設定するコスト
- 頻繁なハードウェア調達やピーク負荷の処理を避けるためのオーバープロビジョニングのコスト

これはすべてを網羅したリストではありません。

- 稼働時間およびその他のサービスレベルアグリーメント (SLA) を監視します。新しい環境に移行することで測定および改善できる SLA は次のとおりです。
  - 合計稼働時間 (Amazon OpenSearch Service が提供する 99.9% の SLA と比較した既存のデプロイの過去の稼働時間データ)
  - 障害復旧 (目標復旧時点と目標復旧時間)
  - さまざまな機能 (検索やインデックス作成など) に関連する応答時間
  - 同時ユーザーの数
  - 異なる地域とクラスター間のレプリケーション時間。

Amazon OpenSearch Service に移行するときは、反復プロセスを使用して、これらの KPI を達成しているかどうか、および望ましい成果を達成しているかどうかを検証します。

## 運用パフォーマンス

現在のソリューションで確認すべき重要な領域は、パフォーマンスメトリクスです。ベンチマークを確立し、ターゲット環境内で達成することが期待される改善点を決定します。これには、稼働時間、SLA およびレイテンシーの要件が含まれます。これは、ほとんどの場合、現在のサービスレベルを確立し、改善するのに役立ちます。通常、お客様は以下のサービスレベルインジケータを確認します。

- 1 秒あたりの読み取りと書き込み
- 読み取りおよび書き込みのレイテンシー
- 稼働時間の割合

独自の SLA を作成するときは、[Amazon OpenSearch Service - サービスレベルアグリーメント](#) を完全に理解することが重要です。

## プロセスのパフォーマンス

ビジネス継続性の目標を確立するには、現在のプロセスパフォーマンスを評価することが重要です。現在のプラットフォームの既存のランブックまたは標準運用手順 (SOP) を確認し、チームが大半の時間を費やしている領域を判断します。移行は、チームがイノベーション、ビジネス機能の構築、カスタマーエクスペリエンスの向上に集中できるように、これらの領域の改善に取り組む良い機会です。サポート履歴やトラブルチケットデータを確認することで、既存の環境の問題点を特定し、サポートや開発スタッフがこれらの問題を解決するのに費やした時間を判断できます。次のメトリクスを把握することは、ターゲット環境によってもたらされる改善を測定するのに役立ちます。

- 平均故障時間 (MTTF) (稼働時間)
- 平均故障間隔 (MTBF)
- 故障の平均検出時間 (MTTD)
- 平均修復時間 (MTTR)
- 受信したサポートチケットの数

## 新しいサービスへのスムーズな移行

サービスのビジネス継続性を確保するには、シームレスな移行を入念に計画することが重要です。移行は、アプリケーションと、検索またはログ分析プラットフォームに関連付けられたサービスをモダナイズする良い機会です。ただし、既存のサービスに影響を与えないように、カットオーバー戦略を慎重に計画する必要があります。このドキュメントの[カットオーバー戦略](#)セクションでは、ターゲット環境へのシームレスなカットオーバーを計画する方法について説明します。

## 財務メトリクス

Amazon OpenSearch Service に移行する理由は多数ありますが、一般的に重要な要因はコストです。既存の環境の総保有コスト (TCO) を把握して、マネージドサービスへの移行により達成したコスト削減効果を測定できるようにします。総保有コストの削減目標に記載されているメトリクスのリストを出発点とすることができます。AWS は、チームが AWS クラウドへの移行のビジネスケースを作成するのに役立つ、[クラウドバリューベンチマーク調査](#)を発表しました。この調査は Amazon OpenSearch Service に固有のものではありませんが、Amazon OpenSearch Service への移行など、ほとんどのクラウド移行で共通する主要な価値領域を対象としています。

ほとんどのケースで、Amazon OpenSearch Service は TCO を削減します。TCO を計算するときは、人員配置コストを組み込むことが重要です。エンジニアが現在の環境を維持するために費やす時間とコストを理解することは、重要な要素です。多くのお客様は、ストレージ、コンピューティン

グ、ネットワークインフラストラクチャのコストとマネージドサービスのコストのみを比較します。しかしながら、これでは正確な総保有コストが得られません。Amazon OpenSearch Service は、エンジニアが実行するしかなかったタスクを管理することで、チームの運用効率を向上させます。これには以下のタスクが含まれます。

- ノードを追加または削除してクラスターをスケーリングする
- パッチ適用
- インプレースのアップグレード
- バックアップの取得
- ログとメトリクスをキャプチャするためのモニタリングツールの設定

これらのアクティビティは、サービスによって自動化されます。また、AWS は本番稼働レベルのサポートチームを提供します。つまり、スタッフはビジネスに直接的な価値をもたらす活動に集中できます。

## 運用とセキュリティ

Amazon OpenSearch Service に移行すると、運用アクティビティは変化します。ノードのプロビジョニング、ストレージの追加、オペレーティングシステムのインストールとパッチ適用、高可用性、スケーリング、その他の低レベルのアクティビティの設定と維持について責任を負うことがなくなります。代わりに、ユースケースと新しいユーザーエクスペリエンスの構築に集中できます。

運用プロセスを最適化するには、Amazon OpenSearch Service が提供するログ記録、モニタリング、トラブルシューティング機能について深く理解する必要があります。

## ランブックと新しいプロセス

計画段階の間に、変更または削除が必要な既存のプロセスを特定します。その後、以前には対応できなかった新しい運用プロセスを追加できます。

Amazon OpenSearch Service は差別化につながらない重労働を排除しますが、アプリケーションが最高のパフォーマンスを実現するように設計およびモニタリングされていることを確認する必要があります。内部または外部要因による健全性の問題を完全に認識できるように、ドメインのモニタリングとアラートを設定する必要があります。最新バージョンへのアップグレードをスケジュールし、開始する必要があります。

このようなすべての運用アクティビティで、ランブックの作成と、既存のランブックの変更が必要になります。インフラストラクチャをモニタリングし、Amazon OpenSearch Service の運用メトリク

ス进行分析するには、ランブックを維持することが重要です。ランブックを使用することで、コンプライアンスと規制要件に従った一貫した運用が実現します。ランブックを使用していない場合は、使用を検討することをお勧めします。事前に計画されたステップを定期的に行うプロセスを作成し、アプリケーションのクラッシュや予期しない障害からの復旧などの修復プロセスが完全に自動化されるようにします。

## サポートとチケット発行システム

デプロイに関連するインシデントをキャプチャするには、チケット発行システムの計画と運用をお勧めします (既に実行しているかもしれません)。 [AWS サポート](#) でサポートチケットを作成する方法について、サポートスタッフにトレーニングを実施する必要がある場合があります。チケットのトラージ中のエスカレーションプロセスを合理化することをお勧めします。

このガイドの後半の「[運用上の優秀性](#)」セクションでは、ランブックや構築プロセスで考慮すべき領域といくつかのベストプラクティスへのリンクを提供します。

## セキュリティ

AWS では、セキュリティを最優先事項としています。Amazon OpenSearch Service は、多層的なセキュリティを提供します。このサービスは、すべてのセキュリティパッチを処理し、VPC によるネットワーク分離、きめ細かなアクセスコントロール、マルチテナントサポートを提供します。保管時のデータは、AWS Key Management Service (AWS KMS) を通じて作成および管理するキーを使用して暗号化されます。ノード間の暗号化機能は、ドメイン内のインスタンス間のすべての通信に Transport Layer Security (TLS) を提供します。Amazon OpenSearch Service は HIPAA にも準拠しており、PCI DSS、SOC、ISO、および FedRAMP 標準に準拠しているため、業界固有の要件や規制要件を満たすことができます。

計画段階の間に、ドメインとやり取りするユーザーとプロセスを特定し、ネットワークトポロジを選択し、各プリンシパルの認証と認可を計画します。組織のセキュリティ要件とコンプライアンス要件に応じて、複数のセキュリティ機能を使用して、ビジネスニーズを満たす環境を作成できます。また、次の要素についても考慮してください。

- VPC – AWS の仮想プライベートクラウド (VPC) 内で Amazon OpenSearch Service を設定できます。これは [推奨設定](#) です。パブリックエンドポイントを使用してドメインを作成することはお勧めしません。クライアントアプリケーションとユーザーがターゲット環境にアクセスできるように、必要なネットワークアーキテクチャを作成するよう計画します。
- 認証 – Amazon OpenSearch Service は、ユーザーまたはソフトウェアクライアントを認証する複数の方法をサポートしています。 [OpenSearch Dashboards](#) にアクセスするために、既存の

ID プロバイダーによる [Amazon Cognito](#) または [SAML 認証](#) をサポートしています。また、IAM ID との統合と、[内部ユーザーデータベースを使用した基本的な HTTP 認証](#) も提供します。認証に適したオプションを設定してテストするように計画を立てる必要があります。詳細については、[OpenSearch Service セキュリティドキュメント](#) を参照してください。

- 認可 – サービスへのアクセス権を設定する際は、最小特権の原則に従うことをお勧めします。Amazon OpenSearch Service は、ドキュメント、行、列レベルでアクセス権を設定できるきめ細かなアクセスコントロールを提供します。

セキュリティ機能について十分に理解し、PoC ステージの間にテストしてください。

## トレーニング

AWS への移行を開始するときは、ソフトウェア開発チーム、運用チーム、サポートチーム、セキュリティチームが Amazon OpenSearch Service に関する知識を備えている必要があります。ソリューションとやり取りするすべてのチームを考慮に入れてください。Elasticsearch 環境または OpenSearch 環境から移行する場合は、ほとんどの知識を引き継ぎます。以下のチームにトレーニングを提供します。

- ソフトウェア開発チーム – ソフトウェア開発チームには、データインジェストを設定するメカニズムなど、API と機能を学習してもらいます。
- 運用チーム – Amazon CloudWatch を使用して、Amazon OpenSearch Service ドメインの操作、運用メトリクスのモニタリング、ログへのアクセスを行う方法について運用チームをトレーニングします。チームメンバーは、OpenSearch Service ドメインに注意を向ける必要が生じたときに警告する、自動アラームの設定方法を学習する必要があります。Splunk など、オンプレミスで使用する既存のツールセットから移行する場合は、ワークロードに対して同様の可視性を提供できる Amazon OpenSearch Service のモニタリングオプションを特定します。
- サポートチーム – サポートチームには、OpenSearch Service リソースを含むランブックを実装する方法を学習してもらいます。AWS サポートサービスを使用するようにランブックとイベント管理手順を更新することもできます。
- セキュリティチーム – セキュリティチームには、きめ細かなアクセスコントロールを設定する方法と、既存の ID プロバイダー (IDP) と統合する方法を学習してもらいます。

## トレーニングオプション

AWS トレーニングと認定は、初心者からプロフェッショナルレベル向けに、AWS でソリューションを構築して運用するために必要なクラウドスキルに関するデジタルトレーニングとクラスルームト

トレーニングの両方を提供します。内容は AWS の専門家によって作成され、定期的に更新されます。複数のトレーニングオプションがあります。

AWS アカウントチームと協力して、適切なリソースを特定できます。以下は、Amazon OpenSearch Service に関するチームのスキルアップを図るために使用できるリソースの一部です。

- **イマージョンデー** – AWS ソリューションアーキテクトはイマージョンデーを開催できます。イマージョンデーは、ユースケース、一般的な実装パターン、およびユースケースに特に関連する可能性のあるロードマップ項目に対応するようにカスタマイズされた実践的なワークショップです。
- **ハンズオンワークショップ** – チームは AWS の専門家によって作成されたセルフサービス型ワークショップに取り組むことができます。
- **[ホワイトペーパーとガイド](#)** – AWS ホワイトペーパーは、クラウドに関する知識を広げる優れた方法です。AWS と AWS コミュニティによって作成されており、具体的なお客様の状況によく対応する詳細なコンテンツが提供されています。
- **[ブログ記事](#)** – AWS の専門家とお客様が執筆したこれらのブログ記事では、最新の発表、ベストプラクティス、ソリューション、サービス機能、お客様のユースケース、その他のトピックについて解説します。
- **ベストプラクティス** – Amazon OpenSearch Service のベストプラクティスを理解するのに役立つオンラインまたはカンファレンスでのトーク、または AWS の専門家によって実行されるセッションに参加します。
- **[AWS プロフェッショナルサービス](#)** – AWS プロフェッショナルサービスチームは、ベストプラクティスと規範的なアドバイスを提供できます。このチームは、IT プロフェッショナルが成功する移行に関する理解を深め、実現するための [トレーニングプログラム](#) を提供しています。

## データフロー

データフローの重点領域には、次の 3 つの領域が含まれます。

- データインGEST
- データ保持
- データ移行アプローチ

## データインジェスト

データインジェストは、Amazon OpenSearch Service ドメインにデータを取り込む方法に焦点を当てています。OpenSearch に適した取り込みフレームワークを選択するときは、データソースと形式を完全に理解することが最も重要です。

取り込み設計を作成またはモダナイズするには、さまざまな方法があります。セルフマネージド型の取り込みパイプラインを構築するための多数のオープンソースツールがあります。OpenSearch Service は、[Fluentd](#)、[Logstash](#)、[OpenSearch Data Prepper](#) との統合をサポートしています。これらのツールは、大半のログ分析ソリューション開発者に好まれています。これらのツールは、Amazon EC2 インスタンス、Amazon Elastic Kubernetes Service (Amazon EKS)、またはオンプレミスにデプロイできます。Logstash と Fluentd はどちらも、出力先として Amazon OpenSearch Service ドメインをサポートしています。ただし、そのためには、Fluentd または Logstash の維持管理、パッチ適用、テストを行い、ソフトウェアバージョンを最新の状態に保つ必要があります。

運用オーバーヘッドを減らすには、Amazon OpenSearch Service との統合をサポートする AWS マネージドサービスのいずれかを使用できます。例えば、[Amazon OpenSearch Ingestion](#) は、フルマネージド型のサーバーレスデータコレクターであり、Amazon OpenSearch Service ドメインにリアルタイムログ、メトリクス、トレースデータを配信します。OpenSearch Ingestion を使用すると、OpenSearch Service ドメインにデータを取り込むときに、Logstash や [Jaeger](#) などのサードパーティーのソリューションを使用する必要がなくなります。OpenSearch Ingestion にデータを送信するようにデータプロデューサーを設定します。その後、指定したドメインまたはコレクションにデータが自動的に配信されます。また、OpenSearch Ingestion で、送信前にデータを変換するように設定することもできます。

もう 1 つのオプションは、サーバーレス取り込みパイプラインの構築に役立つフルマネージドサービスである [Amazon Data Firehose](#) です。Firehose は、[ストリーミングデータを取り込んで変換し、Amazon OpenSearch Service ドメインに配信](#)する安全な方法を提供します。データのスループットに合わせて自動的にスケールリングし、継続的な管理は不要です。Firehose は AWS Lambda、OpenSearch Service ドメインにロードする前に、データを使用して受信レコードを変換、圧縮、バッチ処理することもできます。

マネージドサービスを使用すると、既存のデータインジェストパイプラインを廃止したり、現在のセットアップを拡張して運用上のオーバーヘッドを削減したりできます。

移行計画の立案は、現在の取り込みパイプラインが現在および将来のユースケースのニーズを満たすかどうかを評価するための良い機会となります。セルフマネージド型の Elasticsearch または OpenSearch クラスターから移行する場合、取り込みパイプラインは、クライアントライブラリの更

新を最小限に抑えながら、現在のクラスターから Amazon OpenSearch Service ドメインへのエンドポイントのスイッチングをサポートする必要があります。

## データ保持

データインジェストとストレージを計画するときは、必ずデータ保持の計画を立て、合意を得てください。ログ分析のユースケースでは、履歴データを廃止するための適切なポリシーをドメイン内に作成することが重要です。既存のオンプレミスおよびクラウド VM ベースのアーキテクチャから移行する場合、すべてのデータノードに特定のタイプのインスタンスを使用している可能性があります。データノードに同じ CPU、メモリ、ストレージプロファイルを設定しています。ほとんどのお客様は、高速インデックス作成要件を満たすように高スループットストレージを設定します。この単一のストレージプロファイルのアーキテクチャは、ホットノードのみのアーキテクチャ、または「ホットオンリー」と呼ばれます。ホットオンリーアーキテクチャは、ストレージとコンピューティングを結合します。これは、ストレージ要件が増大した場合にコンピューティングノードを追加する必要があることを意味します。

ストレージをコンピューティングから切り離すために、Amazon OpenSearch Service は UltraWarm ストレージ階層を提供します。UltraWarm は、従来のデータノードよりも大量のデータに対応できるノードを提供することで、Amazon OpenSearch Service に読み取り専用データをコスト効率の高い方法で保存することができます。

計画立案の間に、データ保持と処理の要件を決定します。既存のソリューションのコストを削減するには、UltraWarm 階層を活用します。データ保持要件を特定します。次に、データをホットからウォームに移動するか、不要になったデータをドメインから自動的に削除するインデックス状態管理ポリシーを作成します。これにより、ドメインのストレージ不足を避けることもできます。

## データ移行アプローチ

計画段階では、特定のデータ移行アプローチを決定することが重要です。データ移行アプローチは、現在のデータストアにあるデータを、ギャップなくターゲットストアに移動する方法を定めます。これらのアプローチの手順の詳細については、[ステージ 4 – データ移行](#) セクションで説明します。アプローチの実装はこのステージで行います。

このセクションでは、Elasticsearch または OpenSearch クラスターを Amazon OpenSearch Service に移行するために使用できるさまざまな方法とパターンについて説明します。パターンを選択するときは、次の要素のリストを考慮してください (すべて網羅しているわけではありません)。

- 既存のセルフマネージ型のドクラスターからデータをコピーするか、元のデータソース (ログファイル、製品カタログデータベース) から再構築するか

- ソース Elasticsearch または OpenSearch クラスターとターゲット Amazon OpenSearch Service ドメインのバージョン互換性
- Elasticsearch または OpenSearch クラスターに依存するアプリケーションとサービス
- 移行のために用意できる時間枠
- 既存の環境内のインデックス付きデータの量

### スナップショットから構築する

スナップショットは、セルフマネージド型 Elasticsearch クラスターから Amazon OpenSearch Service に移行する最も一般的な方法です。スナップショットは、Amazon S3 などの耐久性の高いストレージサービスを使用して、OpenSearch または Elasticsearch データをバックアップする方法を提供します。このアプローチでは、現在の Elasticsearch または OpenSearch 環境のスナップショットを作成し、ターゲットの Amazon OpenSearch Service 環境で復元します。スナップショットを復元した後、アプリケーションを新しい環境にポイントすることができます。これは以下の状況においてより迅速なソリューションとなります。

- ソースとターゲットに互換性がある。
- 既存のクラスターに大量のインデックス付きデータが含まれているため、インデックスの再作成に時間がかかる。
- ソースデータがインデックスの再作成に対応していない。

その他の考慮事項については、「[ステージ 4 – データ移行](#)」セクションの「スナップショットに関する考慮事項」を参照してください。

### ソースから構築する

このアプローチは、現在の Elasticsearch または OpenSearch クラスターからデータを移動しないことを意味します。代わりに、ログまたは製品カタログソースからターゲットの Amazon OpenSearch Service ドメインにデータを直接再ロードします。これは通常、既存のデータインGESTパイプラインに軽微な変更を加えて行われます。ログ分析のユースケースでは、ソースからの構築時に、ソースから新しい OpenSearch Service 環境に履歴ログを再ロードする必要がある場合もあります。検索のユースケースでは、完全な製品カタログとコンテンツを新しい Amazon OpenSearch Service ドメインに再ロードする必要がある場合があります。このアプローチは、以下のシナリオに適しています。

- ソース環境とターゲット環境のバージョンが、スナップショットの復元と互換性がない場合。
- 移行の一環として、ターゲット環境でデータモデルを変更する場合。

- ローリングアップグレードを回避するために Amazon OpenSearch Service の最新バージョンにジャンプし、重大な変更に対応したいと考えている場合。これは、比較的古いバージョンの Elasticsearch (5.x 以前) を自己管理している場合にお勧めします。
- インデックス作成戦略を変更することもできます。例えば、毎日ロールオーバーする代わりに、新しい環境で毎月ロールオーバーできます。

ソースから構築するためのオプションについては、「2. ソースから構築する」([「ステージ 4 – データ移行」](#)セクション)を参照してください。

### 既存の Elasticsearch または OpenSearch 環境からリモートでインデックスを再作成する

このアプローチでは、Amazon OpenSearch Service の[リモート再インデックス API](#)を使用します。リモート再インデックスを使用すると、既存のオンプレミスまたはクラウドベースの Elasticsearch クラスターまたは OpenSearch クラスターから Amazon OpenSearch Service ドメインにデータを直接コピーできます。ターゲット環境にカットオーバーするまで、2つの環境のロケーション間でデータを同期できるオートメーションを構築できます。

### オープンソースのデータ移行ツールを使用する

既存の Elasticsearch 環境からターゲット Amazon OpenSearch 環境にデータを移行するには、複数のオープンソースツールを使用できます。その一例が Logstash ユーティリティです。Logstash ユーティリティを使用して、Elasticsearch または OpenSearch クラスターからデータを抽出し、Amazon OpenSearch Service ドメインにコピーできます。

すべてのオプションを評価し、最も使いやすいオプションを選択することをお勧めします。選択したアプローチをフルプルーフとするため、PoC ステージですべてのツールとオートメーションをテストします。これらのアプローチを実装する方法の詳細と、詳しいガイドンスについては、「[ステージ 4 – データ移行](#)」セクションを参照してください。

## デプロイフレームワーク

現代のチームの多くは、継続的インテグレーションと継続的デリバリー (CI/CD) のプラクティスとパイプラインを使用して、ソリューションとインフラストラクチャのデプロイを自動化しています。チームが既に CI/CD パイプラインを使用している場合は、Amazon OpenSearch Service を環境に組み込むことができます。現在のセットアップで手動でデプロイする場合は、パイプラインを構築して反復可能な作業を自動化し、運用オーバーヘッドを減らし、人為的ミスを減らすことを検討してください。

Terraform (HashiCorp)、Chef、Puppet など、さまざまな Infrastructure as Code (IaC) フレームワークを使用して Amazon OpenSearch Service をデプロイできます。Terraform には、Amazon OpenSearch Service ドメインの作成に使用できる [OpenSearch モジュール](#) が用意されています。多くの場合、既存のインフラストラクチャのデプロイパイプラインを使用し、検索エンジンモジュールを Amazon OpenSearch Service モジュールにポイントできます。

パイプラインをゼロから構築することを検討している場合、または AWS ネイティブサービスを使用する場合、AWS にも複数の CI/CD ツールとサービスオプションが用意されています。これには以下が含まれます。

- [AWS CodePipeline](#)
- [AWS CodeBuild](#)
- [AWS Cloud Development Kit \(AWS CDK\)](#)
- [AWS CloudFormation](#)
- [AWS CodeDeploy](#)

これらのサービスを使用して、インフラストラクチャの構築、テスト、デプロイを自動化できます。これらのクラウドネイティブサービスのいずれかを使用してパイプラインをデプロイする場合、次のような多くの利点があります。

- 完全に自動化されたエンドツーエンド (ビルド、テスト、デプロイ) の製品リリース
- 複数の環境へのデプロイ (開発環境、テスト環境、本番稼働前環境、本番稼働環境)
- 他の AWS のサービスとの統合
- デプロイパイプラインをモダナイズして、複数の環境にわたる Amazon OpenSearch Service のデプロイを自動化する機能

## ステージ 2 – 概念実証

移行を実行するときは、ターゲット状態のソリューションが要求通りに機能するかどうかを検証することが重要です。概念実証 (PoC) 演習を実行することを強くお勧めします。このセクションでは、PoC の実行時に考慮すべきさまざまな側面に焦点を当てます。

- 開始条件と終了条件の定義
- 資金の確保
- 自動化
- 入念なテスト
- PoC ステージ
- 障害シミュレーション

### 開始条件と終了条件の定義

明確な開始条件と終了条件を設定することは、PoC 演習を成功させる上で重要です。開始条件を定義する際は、次の点を考慮してください。

- ユースケースの定義
- 環境へのアクセス権
- さまざまなサービスの知識
- 関連するトレーニング要件

同様に、PoC の結果を評価するために使用できる終了条件を定義します。これには以下が含まれます。

- 機能
- パフォーマンス要件
- セキュリティ実装 PoC

### 資金の確保

PoC 条件の定義に基づいて、PoC の資金を確保します。適切なサイズ設定を実行し、関連するすべてのコストを検討していることを確認します。オンプレミスから AWS に移行する場合は、オンプレ

ミスから AWS クラウドへのフレームワークの移行に関連するコストを含めます。AWS の既存のお客様の場合は、AWS アカウントマネージャーに相談して、Amazon OpenSearch Service への移行に使用できるクレジットの対象かどうかを確認してください。

## 自動化

自動化を実行できる場面を特定し、テストを自動化してタイムボックス化するための専用トラックを計画します。デプロイとテストが自動化されると、人為的なエラーなしで繰り返しテスト、検証を迅速に行うことができます。

テストをタイムボックス化することで、時間どおりに実行し、課題が発生した場合は他のアクティビティに切り替えることができます。例えば、パフォーマンステストに想定よりも時間がかかる場合は、そのアクティビティを一時停止できます。その後、開発者が問題を解決している間、他のテストや検証アクティビティを進めることができます。問題が解決したら、パフォーマンステストに戻ることができます。既存のソリューションのパフォーマンスをベンチマークし、PoC 中に設定変更の影響を検証できる自動パフォーマンステストを作成します。

## 入念なテスト

Amazon OpenSearch Service ドメインと統合されている取り込みパイプラインやクエリメカニズムなど、さまざまなレイヤーに必要な検証を実行し、スタックのすべての部分をテストします。これにより、エンドツーエンドのソリューション実装を検証できます。

### プレゼンテーションレイヤー

プレゼンテーションレイヤーでは、次のアクティビティを含む PoC 演習を必ず実行してください。

- 認証 — ユーザー認証用に計画したメカニズムを検証します。
- 認可 — 準拠する認可メカニズムを特定し、期待どおりに動作することを確認します。
- 質問 – 本番環境で発生する最も一般的なユースケースは何ですか？ ビジネスにとって重要なエッジケースシナリオにはどのようなものがありますか？ これらのパターンを特定し、PoC 中に検証します。
- レンダリング – データは、ユースケース全体でさまざまなユーザーに対して正確かつ適切にレンダリングされていますか？ ログ分析のユースケースでは、ターゲットバージョンに応じて OpenSearch Dashboards または Kibana でダッシュボードを構築してテストし、要件を満たしていることを確認できます。

## 取り込みレイヤー

取り込みレイヤーでは、収集、バッファリング、集約、ストレージなどのさまざまなコンポーネントを必ず評価してください。

- 収集 – ログ分析のユースケースでは、ログに記録しているすべてのデータが収集されているかどうかを検証します。検索ユースケースでは、データを供給するソースを特定し、データの完全性と正確性の検証を実行して、収集フェーズが正常に実行されたことを確認します。
- バッファ – トラフィックの急増が生じる場合は、取り込み対象のデータをバッファリングしていることを確認します。バッファリング設計を作成するには、さまざまな方法があります。例えば、Amazon Data Firehose でデータを収集したり、Amazon S3 ストレージをバッファとして使用したりできます。
- 集約 – バルク API の使用など、取り込み中に実行するデータの集約を検証します。
- ストレージ – 実行中の取り込みをストレージが最適に処理できるかどうかを検証します。

## PoC ステージ

PoC を実装し、結果を検証するには、次のような段階を踏むことをお勧めします。事前計画に時間をかけていたとしても、これらの PoC フェーズを繰り返し、計画した PoC を調整することを躊躇しないでください。

- 機能テストと負荷テスト – すべてのレベルについて入念にテストします。スタックのすべての部分で障害をシミュレートします。例えば、2つの大きなノードを持つクラスターがあり、そのうちの1つがダウンした場合、もう1つのノードはクラスター上のすべてのトラフィックを占有する必要があります。このようなシナリオでは、小さいノードを多数用意するほど、ノード障害からの復旧がスムーズになる可能性があります。このようなシナリオでは、負荷のピーク時にワークロードをテストして、パフォーマンスに影響がないことを確認します。テスト中に問題を早い段階で提起し、潜在的な問題が適切なタイミングでさまざまな利害関係者によって評価されるようにします。
- KPI の検証と調整 – PoC の間に、PoC の終了条件で定義した KPI とビジネス成果を満たしていることを確認します。KPI を満たすように設定を調整します。
- 自動化とデプロイ – 自動化とモニタリングは、PoC テスト中に重点を置くべきもう1つの重要な側面です。自動化ステップを絞り込み、詳細なモニタリングとともに検証して、すべての利害関係者が PoC の結果を確信をもって評価できるように十分な情報を提供します。すべてのステップを文書化し、本番稼働への移行時に再利用できるランブックを作成します。

## 障害シミュレーション

障害シナリオをシミュレートし、設計がユーザー要件を満たすために必要な回復力や耐障害性を備えているかどうかを検証することを強くお勧めします。データノードの障害をシミュレートして、クラスターにリカバリを適切に処理するのに十分なリソースがあるかどうかを確認できます。ドメインが大量の取り込みを処理しきれなくなる可能性を確認するには、一部のソースからのログの突然のバーストをシミュレートして、バッファリング設定をテストできます。本番稼働のデプロイにスケールするときに、設計がクォータを超えないことを確認します。詳細については、[サービスクォータ](#)に関する Amazon OpenSearch Service ドキュメントを参照してください。

## ステージ 3 – デプロイ

デプロイステージに到達するまでに、PoC を完了し、ターゲット環境を本番環境にデプロイする方法をよく理解しておく必要があります。次の考慮事項に注意が必要です。

- 自動化の検証 – デプロイの間に、PoC 中に作成した自動化を実行し、期待どおりに動作していることを確認します。また、設定コードを変更するとき、CI/CD の自動化が期待どおりに動作していることを確認します。
- セキュリティの検証 – すべてのセキュリティ設定が期待どおりに動作し、データが保護されていることを確認することが重要です。ID プロバイダーの統合など、ソリューションが会社のセキュリティ標準に照らして検証されていること、およびキーユーザーがログインし、アクセスが許可されているデータにアクセスできることを確認します。
- モニタリング – モニタリング設定をテストし、推奨アラートを設定していることを確認します。CPU、メモリ、ディスク、JVM、シャード割り当てなどの主要なメトリクスを監視します。Amazon OpenSearch Service ドメインおよび関連する統合の状態に関するインサイトを取得するため、Amazon CloudWatch でダッシュボードを構築できます。運用サポートチームがダッシュボードにアクセスできることを確認できます。「[運用上の優秀性](#)」セクションには、ハイパフォーマンスで回復力のある OpenSearch Service ドメインを設定するのに役立つヒントへのリンクが記載されています。
- アラームの演習 – すべてのアラームをテストしてください。Amazon CloudWatch またはアラートプラグインを使用している場合は、Amazon Simple Notification Service (Amazon SNS) や Slack などのすべての統合が期待どおりに動作することを確認します。アラートをシミュレートして、アラートが送信先チャネルに正しく配信されていることを確認します。アラートテキストが有益な情報を提供することを確認します。例えば、アラートは、サポートチームが関連する修復プロセスを実装するための、関連するランブックへのリンクを提供することができます。

## ステージ 4 – データ移行

ターゲット環境の準備が完了し、計画段階で選択したデータ移行戦略を実装できるようになりました。

このセクションでは、4 つの異なるパターンの実装手順について説明します。

- [スナップショットからの構築](#)
- [ソースからの構築](#)
- [リモート再インデックス](#)
- [Logstash の使用](#)

### 1. スナップショットからの構築

スナップショット復元アプローチを使用する場合は、ソース Elasticsearch または OpenSearch クラスタから Amazon OpenSearch Service ドメインにデータをコピーします。

スナップショット復元プロセスは、大きく分けて以下のステップで構成されています。

1. 既存のクラスタから必要なデータ (インデックス) のスナップショットを作成し、そのスナップショットを S3 バケットにアップロードする。
2. Amazon OpenSearch Service ドメインを作成します。
3. バケットへのアクセス許可を Amazon OpenSearch Service に付与し、スナップショットを使用するためのアクセス許可をユーザーアカウントに付与する。スナップショットリポジトリを作成し、それをバケットにポイントする。
4. Amazon OpenSearch Service ドメインでスナップショットを復元する。
5. クライアントアプリケーションを Amazon OpenSearch Service ドメインにポイントする。
6. 保持設定のため、インデックス状態管理 (ISM) ポリシーを作成する (オプション)。

スナップショットは増分です。そのため、スナップショットは段階的に実行および復元できます。スナップショットを使用すると、データをストレージシステム (Amazon S3 など) 上のファイルとして一括で抽出できます。その後、`_restore` API オペレーションを使用して、これらのファイルをターゲット環境にロードできます。これにより、時間のかかるインデックス再作成が不要になり、ネットワークトラフィックも削減されます。

## スナップショットに関する考慮事項

スナップショット復元アプローチを使用する場合は、次の点を考慮してください。

- インデックスの復元中は、検索やインデックスの再生成はできません。ただし、スナップショットの作成中にインデックスを検索してインデックスを再生成することはできます。
- ソースバージョンとターゲットバージョンの Elasticsearch または OpenSearch に互換性がある必要があります。作成されたインデックスのスナップショットのバージョン:
  - 5.x は 6.x に復元できます
  - 2.x は 5.x に復元できます
  - 1.x は 2.x に復元できます
- これは Elasticsearch または OpenSearch スナップショットのポイントインタイム復元であるため、その後にソースクラスターに加えられた変更は、ターゲット Amazon OpenSearch Service ドメインにレプリケートされません。復元が完了するまでソース Elasticsearch または OpenSearch クラスターへのデータの取り込みを停止するか、スナップショット復元プロセスを数回繰り返すことができます。スナップショットは増分であるため、変更のみをコピーしてターゲット環境で復元するので、最初の復元と比べてかかる時間は短くなります。復元が正常に完了したら、取り込みアプリケーションを Amazon OpenSearch Service ドメインにポイントします。
- スナップショットの作成には、デフォルトでは、クラスターの状態とすべてのインデックスのスナップショットが含まれます。Elasticsearch から移行する場合、OpenSearch の ISM 機能を使用して、ターゲット環境に同等のインデックスライフサイクルポリシーを作成する必要があります。Elasticsearch のインデックスライフサイクル管理 (ILM) は、Amazon OpenSearch Service ではサポートされていません。
- スナップショットを以前のバージョンの Elasticsearch または OpenSearch に復元することはできません。例えば、バージョン 7.10 のスナップショットを 7.9 に復元することはできません。同様に、Elasticsearch 7.11 以降から Amazon OpenSearch Service ドメインにスナップショットを復元することはできません。セルフマネージド型 Elasticsearch 環境をバージョン 7.11 以降に移行した場合は、Logstash を使用して Elasticsearch クラスターからデータをロードし、OpenSearch ドメインに書き込むことができます。
- スナップショットは、リポジトリと呼ばれる指定された保存場所にエクスポートします。Elasticsearch または OpenSearch は、リポジトリに多数のファイルを作成します。これらのファイルを変更または削除することはできません。変更または削除を行うと、不整合が発生したり、復元プロセスが失敗したりする可能性があります。

## 2. ソースからの構築

前述のように、ソースからの構築は、現在の Elasticsearch 環境または OpenSearch 環境からデータを移行しないアプローチです。代わりに、ログ、製品カタログデータソース、またはコンテンツソースからターゲットドメインに直接インデックスを構築します。

ソースからの構築では、2つのオプションを使用できます。選択するオプションは、データのデータ型によって異なります。

- **AWS Database Migration Service を使用 – データのソースがリレーショナルデータベース管理システム (RDBMS) であり、ソースが AWS Database Migration Service (AWS DMS) でサポートされている場合は、AWS DMS を使用してデータソースからターゲット Amazon OpenSearch Service ドメインにデータをコピーできます。** AWS DMS は、全ロードおよび変更データキャプチャ (CDC) オプションをサポートしています。全ロードオプションでは、AWS DMS のタスクがソースデータベーステーブルのすべてのデータをターゲット OpenSearch インデックスにコピーします。デフォルトのマッピングを使用するか、カスタムマッピング設定を指定できます。CDC オプションでは、AWS DMS はまずソーステーブルレコードの完全なコピーをターゲット OpenSearch インデックスに作成します。次に、変更されたデータ (更新と挿入) をキャプチャし、OpenSearch インデックスにコピーします。詳細については、ブログ記事「[AWS Database Migration Service のターゲットとしての Amazon Elasticsearch Service の導入](#)」および「[AWS Database Migration Service の移行用に Amazon Elasticsearch Service を拡張する](#)」を参照してください。
- **ドキュメントソースから構築 – データソースが RDBMS でないか、AWS DMS でサポートされていない場合は、オープンソースツールを使用する、またはオープンソースツールと AWS サービスの組み合わせを使用してカスタムソリューションを作成する必要があります。** OpenSearch にロードする前に、ソースデータを JSON ドキュメントに変換する必要があります。ソースから現在の Elasticsearch または OpenSearch 環境へのパイプラインが既に設定されている場合は、クライアントライブラリに適切な変更を加え、(必要に応じて) Amazon OpenSearch Service ドメインのインデックスのデータモデルを変更することで、それらのデータパイプラインを OpenSearch にポイントできます。ソースからインデックスを構築するときは、次の考慮事項に注意してください。
- **ドキュメントの場所 – ドキュメントが既に AWS クラウド、Amazon S3 などのオブジェクトストレージ、またはファイルシステムなどのオンプレミスストレージに保存されている場合があります。**
- **ドキュメントの形式 – ドキュメントが既に JSON 形式である、Amazon OpenSearch Service ドメインに取り込む準備ができていて、または Amazon OpenSearch Service ドメインに取り込む前にクリーンアップ、処理、JSON へのフォーマットが必要である場合があります。**

ソースからの構築では、大まかに次のようなステップが必要です。

1. Amazon OpenSearch Service ドメインでインデックスマッピングと設定を定義します。
2. ドキュメントソースからデータを抽出し、Amazon S3 などのオブジェクトストレージの場所にコピーします。オープンソースのツール (Logstash など)、AWS サービスクライアント (Amazon Kinesis Agent など)、サードパーティーの市販のツール、カスタムプログラムを使用できます。
3. オープンソースのツール (Logstash や Fluent Bit など) またはネイティブ AWS サービス (AWS Lambda や AWS DMS など) を設定して、データを JSON ドキュメントに変換し、オブジェクトストアから Amazon OpenSearch Service ドメインに定期的または継続的にロードします。

詳細については、「[Amazon OpenSearch Service へのストリーミングデータをロードする](#)」を参照してください。

### 3. リモート再インデックス

この場合、ソースのセルフマネージド型 Elasticsearch または OpenSearch クラスターのインデックスは、[reindex document API オペレーション](#)を使用して Amazon OpenSearch Service ドメインに移行されます。reindex document API オペレーションを使用して、既存の Elasticsearch または OpenSearch インデックスからインデックスを作成できます。既存のインデックスは、reindex オペレーションを実行するのと同じクラスター内に配置されている場合も、リモートクラスター内に配置されている場合もあります。Amazon OpenSearch Service は、リモートクラスターでの reindex document API オペレーションの使用をサポートしています。セルフマネージド型 Elasticsearch のインデックスから Amazon OpenSearch Service のインデックスにインデックスを再生成できます。

リモート再インデックスは、リモート Elasticsearch クラスターの Elasticsearch 1.5 以降と、ローカルドメインの Amazon OpenSearch Service 6.7 以降をサポートしています。詳細については、ブログ記事「[Migrate data into Amazon ES using remote reindex](#)」を参照してください。ブログ記事では Amazon Elasticsearch を参照していますが、このガイドは Amazon OpenSearch Service ドメインにも同じように適用されます。

### 4. Logstash の使用

[Logstash](#) は、ソースからデータを収集し、変換またはフィルタリングを実行し、1 つ以上の宛先にデータを送信できるオープンソースのデータ処理ツールです。Amazon OpenSearch Service ドメインにデータを書き込むため、Logstash は次のプラグインを用意しています。

- logstash-input-elasticsearch

- `logstash-input-opensearch`
- `logstash-output-opensearch`

詳細については、「[Loading data into Amazon OpenSearch Service with Logstash](#)」および OpenSearch のブログ記事「[Introducing logstash-input-opensearch plugin for OpenSearch](#)」を参照してください。

## ステージ 5 – カットオーバー

このステージでは、現在の Elasticsearch 環境または OpenSearch 環境からターゲット Amazon OpenSearch Service ドメインにカットオーバーするために採用できるさまざまなアプローチについて説明します。カットオーバーは次の 2 つのステップで実行できます。

- データ同期メカニズムを確立して、ターゲット環境をソースと同期させます。
- ダウンタイムの有無にかかわらず、現在の環境からターゲット環境へのスワップを実行します。

### データ同期

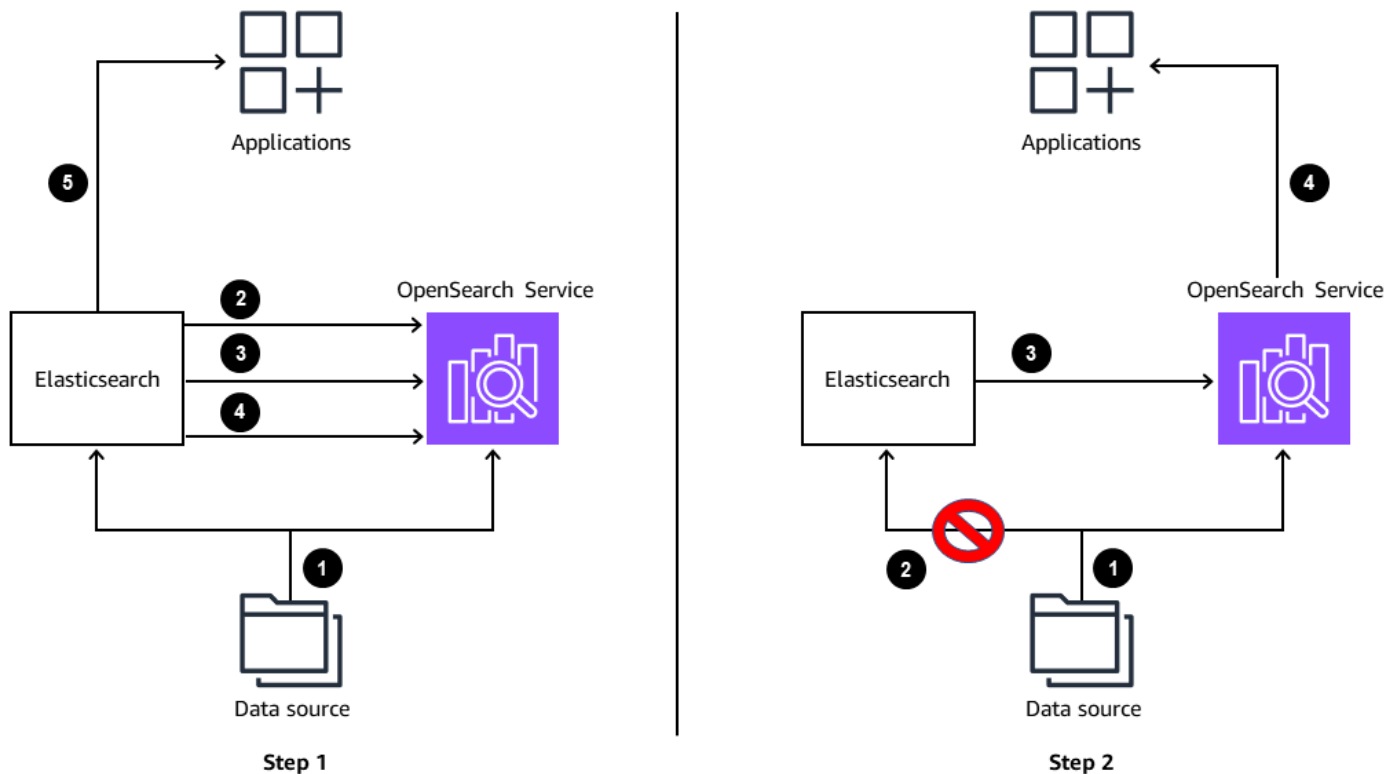
連続データを受信するシステムの場合、データ移行中に新しいデータの受信を停止し、メンテナンスウィンドウで移行を実行する必要がある場合があります (ダウンタイムが発生する可能性あり)。ダウンタイムを許容できない場合は、移行を開始した後に変更をキャプチャできます。ターゲット上で変更を再生して、カットオーバーを実行するまで、ソースと同期された最新の状態を維持します。以下のセクションでは、ソースとターゲットの同期を維持するさまざまな方法について説明します。

#### ログ分析ワークロード

ログ分析ワークロードでは、次の方法で更新と同期を実行できます。

- 保持期間が終了するまで 2 つの環境を並べて実行し、現在の環境とターゲット環境の両方への取り込みを実行できます。ある時点で、アプリケーションをカットオーバーして新しい環境にポイントすることを決定します。場合によっては、ログまたはドキュメントソースから既存のクラスターとターゲット OpenSearch Service 環境の両方に新しいデータを取り込むことができます。その後、現在の環境からコピーすることで、ターゲット環境で古いデータをバックフィルできます。いずれの場合も、ユーザーに影響を及ぼすようなデータギャップがないことを確認する必要があります。
- データ移行の前に、既存の環境への取り込みを一時停止できます。ただし、このアプローチは、データ移行が完了するまで、ユーザーが既存の環境から最新のデータまたは変更されたデータを検索できない可能性があることを意味します。データ移行が完了したら、データインジェストをターゲット環境にポイントし、アプリケーションとクライアントをターゲット環境にスイッチオーバーできます。つまり、移行が完了するまで新しいデータは利用できません。ただし、システムは引き続き検索できます。新しい環境が利用可能になるまで、ソースログとデータをソースに保持する手段が必要です。

- データの最初のパスが移行されるまで、現在のログ分析エンジンを引き続き使用できます。次に、最初のパスが開始された後に生成された残りのデータをバックフィルします。残りのデータが最初のパスよりもはるかに小さいと仮定すると、同期が数分で終わる場合も数時間かかる場合もあるため、残りのデータを同期している間は取り込みを一時停止できます。また、同期ウィンドウが小さくなり、ソースからターゲット環境への取り込みを一時停止し、ユーザーに影響を与えることなくターゲット環境にカットオーバーできるようになるまで、このアプローチを使用してパスを数回実行することもできます。次の図は、増分スナップショットと復元を使用してデータを更新または同期する方法を示しています。



### ステップ 1.

1. データは、ソースからデータインGESTパイプラインを経由して、現在の Elasticsearch 環境と Amazon OpenSearch Service ドメインへと流れます。
2. 最初のパスで Elasticsearch から Amazon OpenSearch Service ドメインに移動するまでが最も長くかかります。
3. 最初の更新または同期パスの所要時間はそれよりも短くなります。
4. 2 回目の更新または同期パスには、最小限の時間がかかります。
5. データは Elasticsearch からアプリケーションに引き続き流れます。

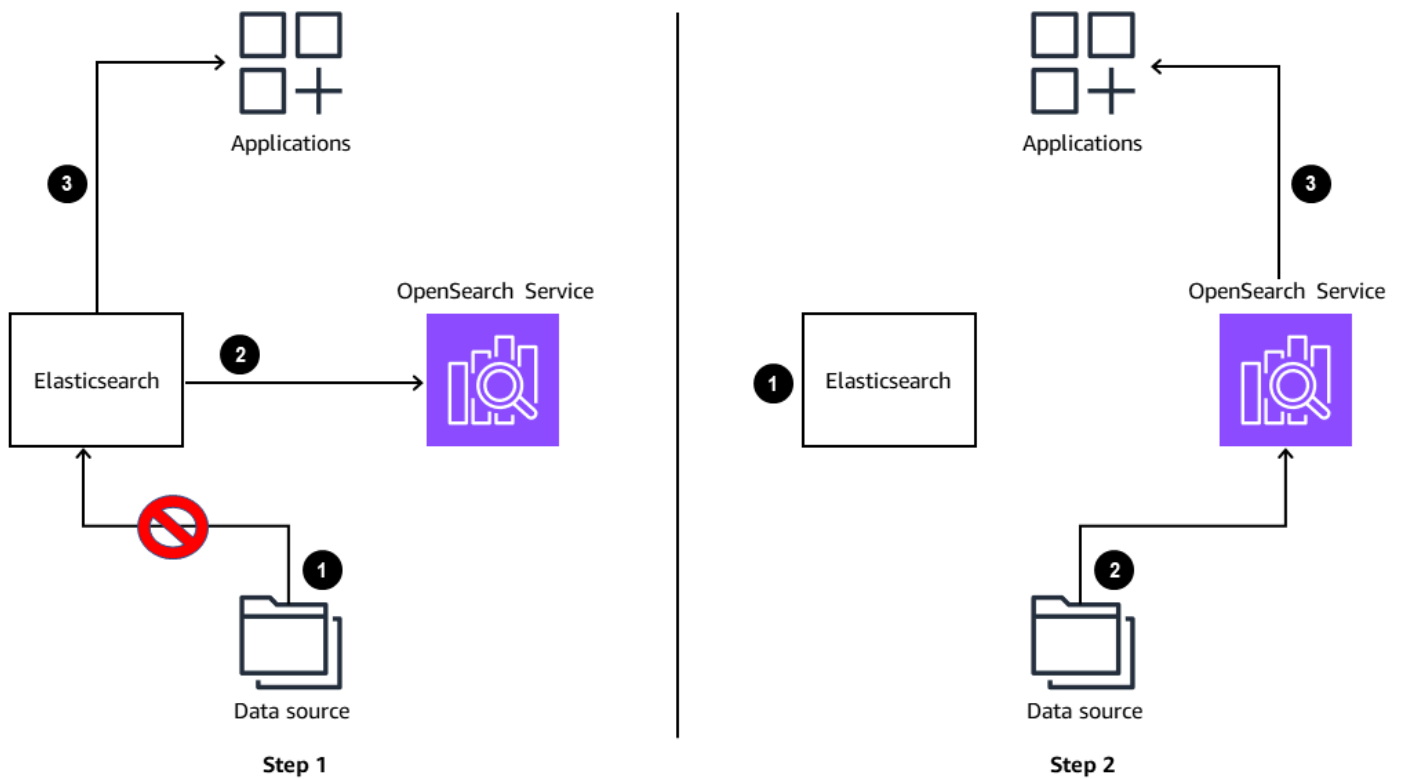
## ステップ 2.

1. データは、ソースからデータインジェストパイプラインを経由して OpenSearch Service ドメインに流れます。
2. 現在の Elasticsearch 環境への取り込みが停止します。
3. 最後の更新または同期パスには、最小限の時間がかかります。
4. データは OpenSearch Service からアプリケーションに流れます。

## 検索ワークロード

前述の 3 つのアプローチでは、カットオーバーを実行する前に、ターゲット上のすべてのデータが最新であることを確認する必要があります。検索ワークロードでは、更新または同期に関する以下の推奨事項を検討してください。

- 検索ワークロードの場合、通常、ソースから現在の環境への取り込みを一時停止します。現在の環境からターゲット環境にすべてのデータをコピーし、移行の開始以降に変更されたデータを決定できる変更データキャプチャ (CDC) メカニズムを導入します。次に、変更されたデータを Amazon OpenSearch 環境にコピーします。ほとんどの場合、検索アプリケーションのデータインジェストパイプラインには既に CDC メカニズムが組み込まれているため、通常は、データが現在の環境から移行された後にパイプラインを新しい環境にポイントするだけです。次の図は、検索ユースケースのためにソースからインデックスを完全に構築する方法を示しています。



### ステップ 1.

1. 現在の Elasticsearch 環境への取り込みが一時停止されます。
2. データは Elasticsearch から OpenSearch Service ドメインにコピーされます。
3. データは Elasticsearch からアプリケーションに引き続き流れます。

### ステップ 2.

1. Elasticsearch 環境とデータソースまたはアプリケーションへの接続が失われます。
  2. 変更データキャプチャ (CDC) データはパイプラインに取り込まれ、OpenSearch Service ドメインに流れます。
  3. データは OpenSearch Service ドメインからアプリケーションに流れます。
- 一部の検索ワークロードでは、ソースデータベースまたはデータソースから新しい OpenSearch Service 環境にフルデータのみをロードする必要があります。ロードが完了すると、クライアントアプリケーションは新しい環境にカットオーバーできます。これは、検索ワークロードの移行を実現する最も簡単な方法です。

## スワップまたはカットオーバー

移行ジャーニーの最後のステップは、新しい環境へのスワップまたはカットオーバーです。これは非常に重要なフェーズの 1 つです。この時点で、本番稼働の準備は整っています。データが同期され、最新の状態になり、モニタリングとアラートが設定され、ランブックが最新の状態になり、新しい環境にカットオーバーする準備が整っています。取り込みが正常に流れていること、および新しい環境のメトリクスが正常であることを確認する必要があります。このステージでは、既存の Elasticsearch クラスターまたは OpenSearch クラスターから新しい Amazon OpenSearch Service ドメインへのクライアント接続のカットオーバーを計画し、実行します。クライアントライブラリの変更が必要になる場合がありますので、注意してください。この時点で、互換性とパフォーマンスを検証するために、Amazon OpenSearch Service のすべてのクライアント機能を下位環境でテストしているはずですが、

新しい環境をポイントする必要があるクライアントアプリケーションがある場合は、DNS エントリを古い環境から新しい環境に更新します。次に、アプリケーションの動作を注意深くモニタリングして、ユーザーが適切なエクスペリエンスを得ていることを確認します。

通常、このドキュメントのガイドラインに従った場合は、安全なスイッチオーバーが行われます。ただし、新しい環境で問題が発生した場合に備えて、ソース環境を最新の状態に保つことをお勧めします。古い環境を廃止する前に、スワップ後も数週間、両方の環境を運用し続ける AWS のお客様もいます。ビジネス継続性の要件に合った戦略を選択することをお勧めします。

## ステージ 6 - 運用上の優秀性

Amazon OpenSearch Service ドキュメントには、[運用上のベストプラクティス](#)についての専用セクションがあります。トピックは次のとおりです。

- [モニタリングとアラート](#)
- [シャード戦略](#)
- [安定性](#)
- パフォーマンス
- [セキュリティ](#)
- [コスト最適化](#)
- [Amazon OpenSearch Service ドメインのサイズ設定](#)
- [Amazon OpenSearch Service のペタバイトスケール](#)
- [Amazon OpenSearch Service の専用マスターノード](#)
- [Amazon OpenSearch Service に推奨される CloudWatch アラーム](#)

新しく移行された環境を運営するには、ドキュメントに記載されているガイドランスに従うことをお勧めします。

## 結論

Amazon OpenSearch Service は、セルフマネージド型 Elasticsearch または OpenSearch クラスターの開発と運用で必要となる、差別化につながらない手間のかかる作業を軽減します。Amazon OpenSearch Service への移行を検討している場合は、このガイドで説明しているプロセスを使用して、状況に適した移行戦略を計画および選択できます。

移行は、セルフマネージド型クラスターからスナップショットを作成して Amazon OpenSearch Service ドメインに復元するような基本的なものである場合も、既存の機能と統合をテストするような複雑なものである場合もあります。このガイドでは、移行プロジェクトチームが移行のすべての側面を確実にカバーし、堅牢な実装戦略を構築するために使用できる情報を提供します。

Amazon OpenSearch Service ドキュメントには、[運用上のベストプラクティス](#)についての専用セクションがあります。新しく移行された環境を運営するには、ドキュメントに記載されているガイドラインに従うことをお勧めします。

# リソース

- [Amazon OpenSearch Service でのインデックススナップショットの作成](#)
- [Use Amazon S3 to Store a Single Amazon OpenSearch Service Index](#) (ブログ記事)
- [Elasticsearch snapshot and restore](#) (Elasticsearch ドキュメント)
- [S3 Repository Plugin](#) (Elasticsearch ドキュメント)
- [Elasticsearch Repository Settings: Recommended S3 Permissions](#) (Elasticsearch ドキュメント)
- [Elasticsearch Client Settings](#) (Elasticsearch ドキュメント)

# 寄稿者

## 寄稿者

本ドキュメントの寄稿者は次のとおりです。

- プリンシパル OpenSearch ソリューションアーキテクト、Muhammad Ali
- 分析担当シニアスペシャリストテクニカルアカウントマネージャー、Gene Alpert
- シニアプリンシパルソリューションアーキテクト、Jon Handler
- シニア OpenSearch スペシャリストソリューションアーキテクト、Prashant Agrawal
- シニアプロダクトマーケティングマネージャー、Ina Felsheim
- シニア分析ソリューションアーキテクト、Sung-il Kim
- OpenSearch ソリューションアーキテクト、Hajer Bouafif
- プリンシパル OpenSearch スペシャリストソリューションアーキテクト、Kevin Fallis
- シニア OpenSearch スペシャリストソリューションアーキテクト、Muthu Pitchaimani
- OpenSearch ソリューションアーキテクト、マネージャー、Kunal Kusoorkar
- プリンシパル分析ソリューションアーキテクトテクニカルリーダー、Imtiaz Sayed
- シニアソリューションアーキテクト、Soujanya Konka
- OpenSearch スペシャリスト、マネージャー、Marc Clark
- シニア OpenSearch スペシャリスト、Bob Taylor
- ヘルスケアとライフサイエンス、プリンシパル分析ソリューションアーキテクト、Aneesh Chandra PN

## 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
<a href="#">初版発行</a>	—	2023 年 8 月 28 日

# AWS 規範ガイドの用語集

以下は、AWS 規範ガイドによって提供される戦略、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、およびそれらを行き移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 は人材開発、トレーニング、コミュニケーションに関するガイダンスを提供し、組織がクラウド導入を成功させるための準備を支援します。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

ターゲットがデータ AWS のサービスを受信する前のローカルでのデータの暗号化。

## 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。AWS 移行戦略との関連性については、「[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、リソースの保護に役立つように、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で利用できるサービス](#)」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「[でのワークロードのディザスタリカバリ](#)」[AWS: クラウドでのリカバリ](#)」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、ガバナンス要件のコンプライアンスに影響を与える可能性のある[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、障害の影響を制限し、ワークロードの耐障害性を高めるのに役立つ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。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。

## OT

「[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 AWS Secrets Manager パスワードやユーザー認証情報などの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

送信先で、それ AWS のサービスを受け取る によるデータの暗号化。

### 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、 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「[Using AWS Organizations with other AWS services](#) 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 익스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。