AWS 大規模な移行の戦略とベストプラクティス

AWS 規範ガイダンス

AWS 規範ガイダンス: AWS 大規模な移行の戦略とベストプラクティス

Table of Contents

AWS 大規模な移行の戦略とベストプラクティス

Amazon Web Services (寄稿者)

2022 年 5 月 (ドキュメント履歴)

多くの AWS お客様は、ビジネスへの影響を最小限に抑えながら、多数のサーバーとアプリケーションを AWS クラウド できるだけ早く に移行したいと考えています。データセンターのリースが更新や終了に近づいているか、組織がテクノロジー変革の最初のステップを実行しているため、組織が大規模な移行プロジェクトを開始している可能性があります。ただし、大規模な は、範囲内のサーバーの数だけ定量化されません。また、人、プロセス、テクノロジー、優先順位などの複雑さを考慮して、移行によって生じる組織変革のレベルも考慮します。

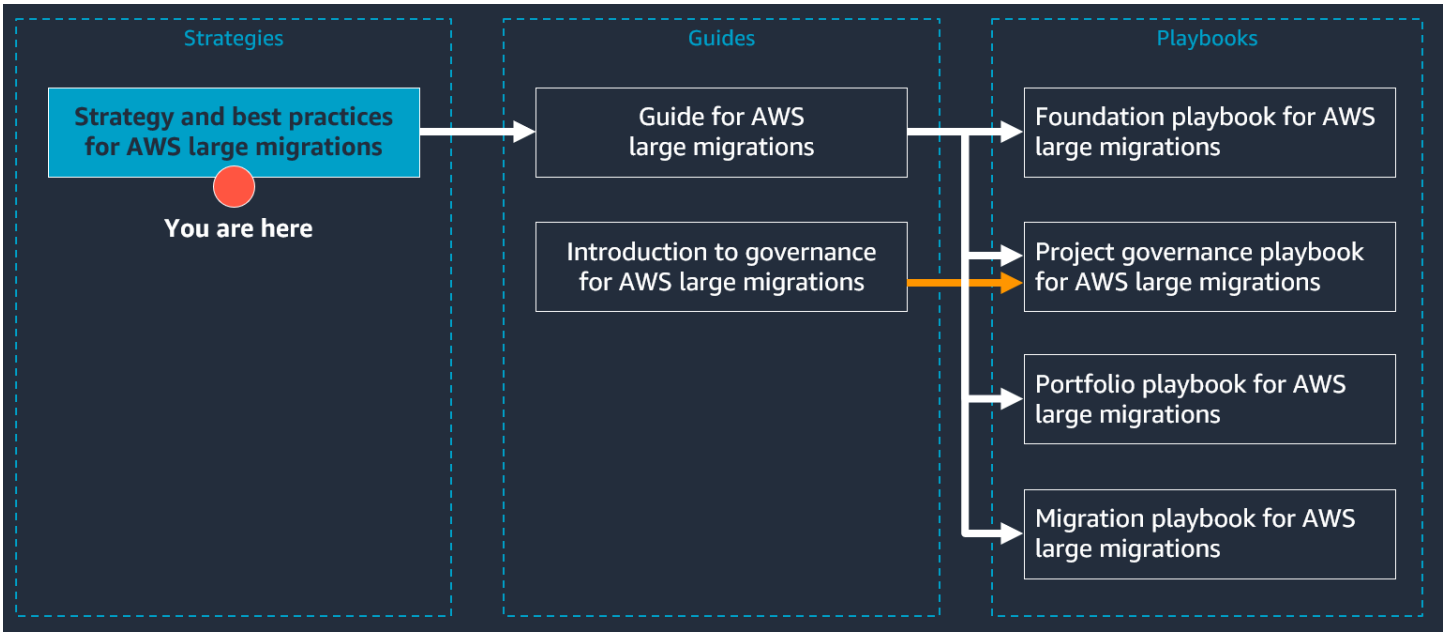
このガイドでは、大規模な移行能力に焦点を当てています AWS。既存のアプリケーションは、ほとんど、またはまったく変更せずに移行できます。クラウドを起動ポイントとして使用して、これらのアプリケーションをクラウドネイティブまたはサーバーレステクノロジーに移行できます。また、アプリケーションをモダナイズして、追加のビジネスメリットを引き出すことができます。

このガイドでは、大規模な移行のベストプラクティスについて説明し、金融サービスや医療など、さまざまなセグメントにわたる顧客からのユースケースを提供します。また、顧客の移行中に学んだ教訓の実例も示します AWS。このガイドの目的は、大規模な移行の初期段階にいるお客様を支援することです。ただし、このガイドのベストプラクティスと戦略は、移行ジャーニーのどの段階でも有益です。すでに に関する 100 レベルの知識があり、AWS 移行の推奨プロセスを理解していることを前提 AWS のサービス としています。

大規模な移行に関するガイダンス

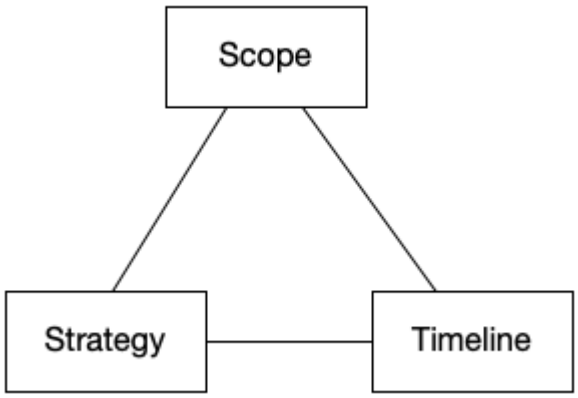
300 台以上のサーバを移行することは「大規模な移行」と見なされます。大規模な移行プロジェクトの人材、プロセス、テクノロジーの課題は、通常、ほとんどの企業にとって新しいものです。このドキュメントは、 への大規模な移行に関する AWS 規範的ガイダンスシリーズの一部です AWS クラウド。このシリーズは、クラウドへの移行を合理化するために、最初から正しい戦略とベストプラクティスを適用するのに役立つように設計されています。

次の図は、このシリーズの他のドキュメントを示しています。最初に戦略を確認し、次にガイドを確認し、プレイブックに進みます。シリーズ全体にアクセスするには、「 への大規模な移行 AWS クラウド」を参照してください。


Strategies
Strategy and best practices for AWS large migrations
You are here
Guides
Guide for AWS large migrations
Introduction to governance for AWS large migrations
Playbooks
Foundation playbook for AWS large migrations
Project governance playbook for AWS large migrations
Portfolio playbook for AWS large migrations
Migration playbook for AWS large migrations

範囲、戦略、タイムライン

すべてのプログラムの構成要素と大規模な移行における関連性を構成するのは、スコープ、戦略、タイムラインの 3 つの重要な要素です。



移行ジャーニーのステージを設定するには、移行プログラムの開始時からこれらの要素を調整して理解する必要があります。これらの要素のいずれかを変更すると、他の要素に影響します。変更がどの程度基本的または賢明であるかにかかわらず、再調整はすべての変更を考慮する必要があります。

スコープ – 移行対象

移行の途中であっても、プログラムの全範囲が未定義になるのは一般的です。これは、後のステージまでさまざまな要因が解凍されない可能性があるためです。例えば、移行の途中で、設定管理データベース (CMDB) に記録されていないシャドウ IT の兆候を発見することがあります。または、これらのアプリケーションの実行に必要なサポートネットワークおよびセキュリティサービス (AWS パートナーへの VPN 接続、証明書に署名する認証機関など) を考慮せずに、計画がサーバービューに重点を置いている可能性があります。範囲の定義に時間を費やし、目標のビジネス上の成果から逆算することをお勧めします。このガイドの後半で説明するベストプラクティスであるアセットを発見するために、検出ツールを使用することになる場合があります。

大規模な移行には未知のものがあるため、範囲は変わります。これらの未知は、関連性をほとんどまたはまったく理解せずに環境のアーキテクチャの一部となったシステム、または計画の遅延や移行を引き起こす本番環境のインシデントの形式である可能性があります。重要なのは、プログラムを前進させるための柔軟な対応と緊急時対応計画を立てることです。

戦略 – 移行する理由

次の AWS 1 つ以上の理由により、 への移行を計画している場合があります。

- アプリケーションチームは、新しい CI/CD パイプラインを実装したり、最新のアプリケーションスタックをデプロイしたり、サポートされていないレガシープラットフォームをモダナイズしたりしたいと考えています。
- インフラストラクチャチームは、リースの有効期限が切れてプロバイダーが電源をオフにする前に、古いデータセンターからすぐに抜け出す必要があります。
- ボードは、お客様が戦略的方向性としてクラウドに移行する必要があると判断し、ビジネスの未来における変化のペースを速めます。

理由が何であっても、これらの理由などはすべて、ビジネス組織や IT 組織の頭に浮かぶものになります。ドライバーの内容を理解し、伝達し、優先順位を付けることが重要です。ドライバーを追加するたびに、既に大規模な移行に時間、コスト、範囲、リスクが追加される可能性があります。戦略がタイムラインと範囲に与える影響を完全に把握することが重要です。

移行戦略を定義した後、成功の主な鍵の 1 つは、さまざまな利害関係者やチームの要件の整合性です。移行を実行するには、インフラストラクチャ、セキュリティ、アプリケーション、オペレーションなど、組織全体のさまざまなチームが必要です。これらのチームには、個別の優先順位と、既に開始されている可能性のある他のプロジェクトがあります。これらのチームが異なるスケジュールや優先順位に取り組んでいる場合は、移行計画について合意して実装するのがより困難です。移行チームと主要な利害関係者は、関係するすべてのチームが 1 つの目標に向かって取り組み、優先順位を移行の 1 つのタイムラインに合わせる必要があります。

望ましいビジネス成果をさまざまなチーム間でどのように調整できるかを検討することをお勧めします。例えば、 に移行 AWS して AWS Key Management Service （ AWS KMS) を使用して保管中のストレージを暗号化すると、移行とセキュリティの両方の目標を達成できる可能性があります。

多くの場合、企業はアプリケーションをモダナイズしたいと考えています。これによりインフラストラクチャのアップグレードが発生する可能性があります。一方、インフラストラクチャチームは、インフラストラクチャの変更を最小限に抑え、不本意な対応をしたいと考えています。大規模な移行の考え方は、できるだけ基本的なものでなければなりません。関係するチームは、一度にすべてを実行しようとしないようにする必要があります。

これを実現するには、プロジェクトの早い段階で適切な期待を設定します。キーメッセージは「最初に移行してからモダナイズする」である必要があります。このアプローチにより、組織は技術的負債を削減し、最終的に大規模な運用ができるだけでなく、 が提供 AWS クラウド できるスケーラビリティと俊敏性を使用することで、さまざまなモダナイゼーションアプローチへの道を開くことができます。長期的な検討は、インフラストラクチャチームがインフラストラクチャのデプロイと管理を合理化するのに役立ちます。その結果、ビジネスは機能のリリースサイクルを短縮できます。

タイムライン – 移行はいつ完了する必要がありますか？

ビジネスケースに応じて、割り当てられた時間内に達成できる以上の作業を行っていないことを確認する必要があります。移行のドライバーが固定の完了日に基づいている場合は、そのタイムライン要件を満たす戦略を選択する必要があります。ほとんどの大規模な移行は、これらの時間ベースの制約に基づいているため、移行戦略は、拡張やオーバーランの余地をほとんど持たずに、定義された固定されたタイムラインと成果を持っている必要があります。

これらの時間的制約のあるタイプの移行では、「最初に移行してからモダナイズ」アプローチをお勧めします。これにより、期待値を設定し、チームが個々のプロジェクト計画と予算が移行目標全体と一致していることを確認することができます。プロジェクトのできるだけ早い段階で意見の不一致を見つけ、迅速に失敗し、意見の不一致にステアリング委員会レベルで対処し、適切な利害関係者を関与させ、調整を整えることが重要です。

逆に、移行の主な目的がアプリケーションのモダナイゼーションのメリットを享受することである場合は、プログラムの早い段階で呼び出す必要があります。多くのプログラムは、固定された期限に基づいて初期目標から始まり、未解決の問題や問題を解決したいステークホルダーの要件を計画していません。場合によっては、これらの問題はソースシステムに何年も存在していますが、現在は移行の人為的なブロックになっています。

移行中のモダナイゼーションアクティビティは、ビジネスアプリケーションの機能に影響を与える可能性があります。オペレーティングシステムのバージョン変更など、小規模なアップグレードと認識されるものであっても、プログラムのタイムラインに大きな影響を与える可能性があります。これらは些細なものと見なすべきではありません。

大規模な移行のベストプラクティス

組織の機能を管理する要因によっては、大規模な移行が難しくなる可能性があります。このセクションでは、労力の初期段階で対処し、プロジェクト全体で追跡すれば、大規模な移行を簡素化できる主な要因について説明します。

大規模な移行に関する以下のベストプラクティスは、他のお客様から取得したデータに基づいています。ベストプラクティスは 3 つのカテゴリに分かれています。

- 人員
- テクノロジー
- プロセス

人々の視点

このセクションでは、人的視点の以下の主要分野に焦点を当てます。

- エグゼクティブサポート — 意思決定を行えるシングルスレッドリーダーの特定
- チームコラボレーションとオーナーシップ – さまざまなチーム間のコラボレーション
- トレーニング – さまざまなツールに関するチームのプロアクティブトレーニング

エグゼクティブサポート

このセクションの内容:

- シングルスレッドリーダーを特定する
- シニアリーダーシップチームの調整

シングルスレッドリーダーを特定する

大規模な移行を開始するときは、プロジェクトに 100% 専念し、説明責任を負うシングルスレッドのテクニカルリーダーを特定することが重要です。このリーダーは、一貫した優先順位を維持することで、意思決定を行い、サイロを回避し、ワークストリームを合理化することができます。

大規模な移行のグローバルカスタマーは、プログラムの開始時に毎週 1 台のサーバーから、2 か月目の開始時に毎週 80 台を超えるサーバーにスケールできました。シングルスレッドリーダーとして

の CIO の完全なサポートは、移行中のサーバーの迅速なスケールアップにとって重要でした。CIO は、移行チームと毎週移行カットオーバーコールを行い、問題のリアルタイムのエスカレーションと解決を確保し、移行速度を加速しました。

シニアリーダーシップチームの調整

移行の成功基準に関して、さまざまなチーム間で調整を行うことが重要です。移行の計画と実装は小規模な専有チームによって達成できますが、戦略を定義して周辺機器のアクティビティを実行するときに課題が発生します。これらの潜在的な障害には、次のような IT 組織のさまざまな分野からのアクションやエスカレーションが必要になる場合があります。

- ビジネス
- アプリケーション
- ネットワーク
- セキュリティ
- インフラストラクチャ
- サードパーティーベンダー

アプリケーション所有者、リーダーシップ、調整、シングルスレッドリーダーへの明確なエスカレーションからの直接的なアクションが重要になります。

チームコラボレーションと所有権

このセクションの内容:

- 部門横断的なクラウド有効化チームを作成する
- コア移行チーム外のチームや個人の要件を事前に定義する
- ワークロードの移行時にライセンスの問題がないことを検証する

部門横断的なクラウド有効化チームを作成する

大規模な移行プロジェクトの重要な最初のステップは、組織がクラウドで作業できるようにすることです。これを実現するには、クラウド有効化エンジン (CEE) を構築することをお勧めします。CEE は、組織の移行に向けた運用準備に重点を置いた、権限と説明責任を持つチームです AWS。CEE は、インフラストラクチャ、アプリケーション、運用、セキュリティからの代表者を含む部門横断的なチームである必要があります。チームには以下の責任が課金されます。

- ポリシーの開発
- 組織のクラウド運用モデルを確立するツール、プロセス、アーキテクチャの定義と実装
- ステークホルダーが表すすべての分野にわたるステークホルダーの連携を継続的に促進する

ある医療の顧客が CEE から始めていませんでした。ただし、最初のパイロット移行では、ギャップが特定されました。最終的な移行カットオーバー日まで、厳しい期限が設定され、チームは移行の戦いの場を実装しました。移行の戦場では、インフラストラクチャ、セキュリティ、アプリケーション、ビジネスの関係者が問題の解決を支援できます。

コア移行チーム外のチームや個人の要件を事前に定義する

コアプログラム外のチームや個人を特定し、移行計画段階での関与を定義します。後の段階で移行の勢いを高めるには、アプリケーションチームの関与に特に注意してください。アプリケーションに関する知識、問題を診断する能力、カットオーバーにサインオフする要件が必要です。

移行はコアチームが主導しますが、アプリケーションチームはカットオーバー中の移行計画の検証とテストに関与している可能性があります。お客様は、多くの場合、アプリケーション移行ではなくインフラストラクチャプロジェクトとしてクラウド移行にアプローチします。これにより、移行中に問題が発生する可能性があります。

移行戦略を選択するときは、アプリケーションチームに必要な関与を検討することをお勧めします。例えば、リホスト戦略では、リプラットフォームやリファクタリング戦略と比較して、アプリケーションとチームの関与が少なくて済みます。リプラットフォームやリファクタリング戦略では、アプリケーションの環境がますます変化しています。アプリケーション所有者の可用性が制限されている場合は、リファクタリング、再配置、再購入戦略ではなく、リホストまたはリプラットフォームを使用することを検討してください。

ワークロードの移行時にライセンスの問題がないことを検証する

企業の既製の製品をクラウドに移行すると、ライセンスが変わる可能性があります。ライセンス契約は、オンプレミスの資産に焦点を当てている場合があります。例えば、ライセンスは CPU によって取得されたり、特定の MAC アドレスにリンクされたりします。または、ライセンス契約にパブリッククラウド環境でホストする権限が含まれていない場合があります。ただし、ベンダーとのライセンスの再交渉にはリードタイムが長くなり、移行が難しくなる可能性があります。

移行の範囲が定義されたら、すぐに調達チームまたはベンダー管理チームと協力することをお勧めします。ライセンスは、ターゲットアーキテクチャと移行パターンにも影響を与える可能性があります。

トレーニング

このセクションの内容:

新しいツールとプロセスについてチームをトレーニングする

移行戦略が定義されたら、移行とターゲット運用モデルに必要なトレーニングを理解するために時間を費やします。移行中は、組織にとって AWS Database Migration Service初めてのツールを使用する可能性があります。チームをプロアクティブにトレーニングすることで、移行フェーズで発生する遅延を軽減できます。

実践的な方法でツールを試す機会を提供するアクティブな知識移転方法を探すことをお勧めします。例えば、 AWS プロフェッショナルサービスは、大規模な移行を担当する 3 つのシステムインテグレーター (SI) AWS パートナーに、いくつかの Cloud Migration Factory トレーニングセッションを提供しました。これにより、チームは移行フェーズに移行する際に基本的な知識を持つことができました。また、各 SI AWS パートナーチーム内でファーストラインエスカレーションとして機能する可能性のある対象分野の専門家 (SMEs) の特定にも役立ちます。

テクノロジーの視点

テクノロジーは、大規模な移行を加速するための優れた基盤を提供します。例えば、Cloud Migration Factory ソリューションは、移行にend-to-endの自動化を提供する方法に焦点を当てています。このセクションでは、スコープ、戦略、タイムラインに合わせて、必要なスケールと速度を達成するためにテクノロジーを使用するためのベストプラクティスをいくつか説明します。

包括的な原則は、可能な限り自動化の領域を確認することです。対象範囲内に何千ものサーバーがある場合、タスクを手動で実行すると、コストがかかり、時間がかかる可能性があります。

移行を実行するために、通常、次のようないくつかのツールが使用されます。

- 発見
- 移行の実装
- 設定管理データベース (CMDB)
- インベントリスプレッドシート
- プロジェクト管理

これらのツールは、評価から準備、実装まで、移行のさまざまな段階で使用されます。これらのツールの選択は、ビジネス目標とタイムラインによって決まります。

移行フェーズが計画されたら、次のステップは、移行チームが必要なツールを使用するためのスキルを持っていることを確認することです。チームにスキルや経験がない場合は、ターゲットを絞ったトレーニングを計画してスキルセットを強化します。可能であれば、チームが安全な環境で移行ツールの経験を積むことができるイベントを作成します。例えば、チームがツールの使用体験に移行できるサンドピットサーバーやラボサーバーはありますか？ または、初期開発ワークロードを学習目的で使用することは可能ですか？

自動化、追跡、ツールの統合

このセクションの内容:

移行検出を自動化して必要な時間を短縮する

ほとんどの大規模な移行プログラムは、移行の範囲 (移行対象) を理解し、戦略 (移行方法) を開発することで開始されます。検出は、この重要な側面です。必要なメタデータポイントは、移行戦略決定ツリーを形成するためにキャプチャされます。ワークロードをペースで移行するには、必要な移行メタデータを特定して、移行ファクトリーなどの実装プロセスにインポートする必要があります。移行メタデータを抽出、変換、ロード (ETL) する完全に自動化されたメカニズムにより、検出プロセスにかかる時間と労力が大幅に削減されます。

ある顧客が、移行ファクトリー用に完全に自動化されたデータ取り込みプロセスを開発しました。すべての移行メタデータを含む移行ウェーブプランは、Microsoft SharePoint のスプレッドシートでホストされ、維持されています。ソースが変更されると、手動操作なしで移行ファクトリーにデータをロードする AWS Lambda 関数が開始されました。この自動データ取り込みプロセスにより、お客様は手動作業を減らし、人為的ミスを最小限に抑え、スピードを加速できました。1,000 を超えるサーバーを に移行できました AWS。

反復タスクを自動化する

移行実装フェーズでは、多くの小さなプロセスを頻繁に繰り返す必要があります。 AWS Application Migration Service (MGN) を使用する場合、例えば、移行の対象となる各サーバーに エージェントをインストールする必要があります。

特定のビジネス要件や技術要件に適した移行ファクトリーを構築することは、大規模な移行を成功させるために必要な効率とスピードを実現する最も効果的な方法です。移行ファクトリーは、標準化されたデータセットを使用して移行を加速する統合およびオーケストレーションフレームワークを提供します。すべてのタスクを特定したら、規範的なランブックとともに自動化できるすべての手動タスクを自動化する時間を取ります。

Cloud Migration Factory ソリューションはその一例です。Cloud Migration Factory は、組織に固有の側面を自動化できる移行自動化基盤を提供するように設計されています。例えば、CMDB のフラグを更新して、オンプレミスサーバーを廃止できることを強調できます。このシナリオでは、移行ウェーブの最後にこのタスクを実行するオートメーションを作成できます。Cloud Migration Factory には、すべてのウェーブ、アプリケーション、サーバーメタデータを含む一元化されたメタデータストアがあります。自動化スクリプトは Cloud Migration Factory に接続して、そのウェーブ内のサーバーのリストを取得し、それに応じてアクションを実行できます。Cloud Migration Factory は をサポートしていますAWS Application Migration Service。

追跡とレポートを自動化して意思決定を迅速化する

プログラムの重要業績評価指標 (KPIs) など、ライブデータを追跡してレポートするための自動移行レポートダッシュボードを作成することをお勧めします。移行プロジェクトには、以下を含む組織全体のステークホルダーが参加します。

- アプリケーションチーム
- テスター
- チームの廃止
- アーキテクト
- インフラストラクチャチーム
- リーダーシップ

これらの利害関係者は、ロールを実行するためにライブデータを必要とします。例えば、ネットワークチームは、オンプレミスリソースと 間の共有接続への影響を理解するために、今後の移行の波を知る必要があります AWS。リーダーシップチームは、移行の完了度を把握したいと考えています。

信頼性の高い自動ライブフィードにより、ミスコミュニケーションが防止され、意思決定の基盤が提供されます。

大規模なヘルスケアのお客様が、データセンターの出口に向けて、期限が近づいています。規模と複雑さを考慮すると、当初はステークホルダー間の移行ステータスの追跡と伝達にかなりの時間が費やされていました。移行チームは後で Amazon QuickSight を使用して、データを視覚化する自動ダッシュボードを構築しました。これにより、移行速度を向上させながら追跡と通信を大幅に簡素化できます。

移行を容易にするツールを調べる

移行に適したツールを選択することは、特に組織内の誰も大規模な移行を管理したことがない場合、簡単ではありません。

移行をサポートする適切なツールを選択する時間を取ることをお勧めします。この調査にはライセンスコストが伴う場合がありますが、より広範なイニシアチブを検討するとコスト上のメリットが得られます。または、組織に埋め込まれたツールも同様の結果をもたらす可能性があります。例えば、アプリケーションのパフォーマンスモニタリングツールをエステート全体に既にデプロイしていると、豊富な検出情報を提供できます。

テクノロジーのお客様は、最初は知識不足のため、移行中に自動検出ツールを実行することに消極的でした。その結果、SI AWS パートナーは、サーバー名、オペレーティングシステムのバージョン、依存関係など、エステートを手動で検出するために、アプリケーションごとに 5 時間、10 時間の会議を実行する必要がありました。検出ツールを使用した場合、検出の労力が 1,000 時間以上削減された可能性があると推定されました。

前提条件と移行後の検証

このセクションの内容:

- 移行前の段階でランディングゾーンを構築する
- 前提条件アクティビティの概要
- 継続的な改善のための移行後のチェックを実装する

移行前の段階でランディングゾーンを構築する

移行ウェーブ中に AWS ターゲット仮想プライベートクラウド (VPCs) とサブネットを構築するのではなく、事前にターゲット環境またはランディングゾーンを構築することをお勧めします。適切に設

計されたランディングゾーンの構築は、移行の前提条件です。ランディングゾーンには、モニタリング、ガバナンス、運用、セキュリティコントロールを含める必要があります。

移行前にランディングゾーンを構築して検証することで、新しい環境でワークロードを実行する際の不確実性を最小限に抑えることができます。ランディングゾーンを導入することで、ステークホルダーはアカウントまたは VPC レベルで管理される側面を気にすることなく、ワークロードの移行に集中できます。

前提条件アクティビティの概要

ランディングゾーンに加えて、移行前に他の技術的前提条件、特にリードタイムが長いプロセスを調整することが重要です。例えば、オンプレミスから にデータをレプリケートできるように、必要なファイアウォールの変更を行います AWS。技術的な前提条件を早期に伝達することで、必要なリソースの準備と割り当てに役立ちます。前提条件が満たされていないため、移行が停止するのが一般的です。これは進行中の移行の波に影響を与えるだけでなく、問題が修正されている間、将来のすべての移行の日付をプッシュバックする可能性があります。

複数のデータセンターを退避させるという目的で AWS、大規模な移行を実行することを目的とした金融サービス会社。ただし、オンプレミスと の間で利用できる帯域幅は、意図した速度では不十分 AWS でした。残念ながら、帯域幅を増やすには新しい接続が必要で、リードタイムは 3 か月でした。これは、移行速度が最初の 3 か月間制約されたことを意味します。

継続的な改善のための移行後のチェックを実装する

最後に、オペレーションの統合、コストの最適化、ガバナンスとコンプライアンスのチェックなど、移行後の検証を必ず実装してください。移行後の検証には、以前に移行されたワークロードを評価して、将来の波に適用すべき技術的教訓を明らかにすることが含まれます。

さらに、これはコスト管理オペレーションを実装する良い機会です。例えば、移行中に、パフォーマンステストの必要性を減らすために AWS 、インスタンスのサイズをオンプレミスの資産と一致させることにしたとします。これで、データセンターの閉鎖クリティカルパスでのテストは終了しました。Amazon CloudWatch を使用してインスタンスの使用率を評価し、より小さいサイズのインスタンスが適切かどうかを判断できます。

このフェーズの重要性を説明するために、ある大規模なテクノロジーのお客様が大規模な移行を行っていましたが、当初は移行後の検証が含まれていませんでした。100 台を超えるサーバーを移行した後、 AWS Systems Manager エージェント (SSM Agent) が正しく設定されていないことがわかりました。以前に移行したすべてのサーバーを修復する必要があり、移行が停止しました。また、お客様はインスタンスが最初の見積もりの 5 倍の大きさであることも特定したため、各移行ウェーブの最後にコストチェックポイントを実装しました。

プロセスのパースペクティブ

プロセスは一貫性をもたらしますが、各プロジェクトが一意であるため、進化し、変更の影響を受けやすくなります。プロセスを繰り返し実行すると、失敗、学習、導入、反復に伴う大きなメリットにつながる可能性のあるギャップと改善の余地を特定します。これらの変化は、プロジェクトとビジネスが将来活用できる新しいアイデアやイノベーションにつながる可能性があり、品質とチームの自信をもたらす成長への推進力となります。

移行プロセスは、以前にリンクされていない可能性のあるテクノロジーや境界を越えるため、複雑になる可能性があります。この視点は、大規模な移行の特定の要件に関するプロセスとガイダンスを提供します。

大規模な移行の準備

以下のセクションでは、移行ジャーニーを明確な方向性で開始し、成功に不可欠なステークホルダーからの賛同を得るために必要な基本原則の概要を説明します。

このセクションの内容:

- ビジネスドライバーを定義し、タイムライン、範囲、戦略を伝える
- ブロッカーの削除に役立つ明確なエスカレーションパスを定義する
- 不要な変更を最小限に抑える
- end-to-endプロセスを早期に文書化する
- 標準の移行パターンとアーティファクトを文書化する
- 移行メタデータとステータスの信頼できる単一のソースを確立する

ビジネスドライバーを定義し、タイムライン、範囲、戦略を伝える

大規模な移行に近づくと AWS、サーバーを移行する方法が多数あることがすぐにわかります。例えば、次の操作を実行できます。

- を使用してワークロードをリホストしますAWS Application Migration Service。
- アプリケーションをコンテナ化し、Amazon Elastic Container Service (Amazon ECS) または Amazon Elastic Kubernetes Service (Amazon EKS) マネージドコンテナプラットフォームでホストします。
- ワークロードを完全なサーバーレスアプリケーションに再設計します。

正しい移行パスを決定するには、ビジネスドライバーから逆算することが重要です。ビジネスの俊敏性を高めることが最終的な目標である場合は、より高度な変換を伴う 2 番目のパターンを優先できます。年末までにデータセンターを退避することが目標である場合は、 のリホスト速度によりワークロードをリホストすることを選択できます。

大規模な移行には、通常、次のような幅広い利害関係者が関係します。

- アプリケーション所有者
- ネットワークチーム
- データベース管理者
- エグゼクティブスポンサー

移行のビジネス推進要因を特定し、そのリストを、移行プログラムのメンバーからアクセスできるプロジェクト憲章などのドキュメントに含めることが重要です。さらに、目標とするビジネス成果に密接に一致する主要業績評価指標 (KPIs) を作成します。

例えば、ある顧客が、データセンターを退避するという目標ビジネス成果を達成するために、12 か月以内に 2,000 台のサーバーを移行したいと考えています。ただし、セキュリティチームはこの目標に向けて調整されていませんでした。その結果、データセンターの閉鎖日を逃したが、アプリケーションをモダナイズするか、最初にリホストしてタイムリーなデータセンターの閉鎖を可能にし、アプリケーションをモダナイズするかについて、数か月にわたる技術的な議論がありました AWS。

ブロッカーの削除に役立つ明確なエスカレーションパスを定義する

大規模なクラウド移行プログラムには、通常、幅広い利害関係者が参加します。結局のところ、数十年間オンプレミスでホストされているアプリケーションが変更される可能性があります。各ステークホルダーにとって、優先順位が競合することが一般的です。

すべての優先順位が価値を高める可能性がありますが、プログラムは予算の量が制限され、目標成果が定義されます。さまざまなステークホルダーを管理し、目標とするビジネス成果に集中することは難しい場合があります。この課題は、移行の対象となる数百または数千のアプリケーションに乗算すると悪化します。さらに、ステークホルダーは、他の優先事項を持つさまざまなリーダーシップチームにレポートする可能性が高いです。これを念頭に置いて、目標とするビジネス成果を明確に文書化しながら、明確なエスカレーションマトリックスを定義して、ブロッカーを排除することが重要です。これにより、多大な時間を節約し、さまざまなチームを共通の目標に合わせることができます。

これを示す例として、12 か月以内にプライマリデータセンターを退避することを目標とする金融サービス会社が挙げられます。明確な義務や エスカレーションの道筋がなかったため、時間と予算の

制約に関係なく、ステークホルダーが望ましい移行の道筋を構築できました。CIO へのエスカレーション後、明確な義務が設定され、必要な決定をリクエストするためのメカニズムが提供されました。

不要な変更を最小限に抑える

変化は良好ですが、変化が多いほどリスクが高くなります。大規模な移行のビジネスケースが承認されると、特定の日付までにデータセンターを退避するなど、このイニシアチブを推進する目標ビジネス成果がある可能性が最も高くなります。技術者が AWS サービスを最大限に活用するためにすべてを書き換えることは一般的ですが、これはビジネス目標ではない可能性があります。

あるお客様は、会社のウェブスケールインフラストラクチャ全体を 2 年間移行することに重点を置いていました AWS。アプリケーションチームがアプリケーションを書き換えて数か月を費やすのを防ぐためのメカニズムとして、2 週間のルールを作成しました。2 週間のルールを使用することで、顧客は、複数年にわたって何百ものアプリケーションを移動する必要がある場合でも、一貫した頻度で長期的な移行を維持できました。詳細については、ブログ記事「The Two-week Rule: Refactor Your Applications for the Cloud in 10 Days」を参照してください。

ビジネス成果と一致しない変更は最小限に抑えることをお勧めします。代わりに、将来のプロジェクトでこれらの追加変更を管理するメカニズムを構築します。

end-to-endプロセスを早期に文書化する

大規模な移行プログラムの初期段階で、完全な移行プロセスと所有権の割り当てを文書化します。このドキュメントは、移行の実行方法と役割と責任についてすべての利害関係者を教育するために重要です。また、このドキュメントは、問題が発生する可能性がある場所を理解し、移行を進めるにつれてプロセスの更新と反復を提供するのに役立ちます。

移行プロジェクトの開発中に、既存のプロセスを理解し、統合ポイントと依存関係が明確に文書化されていることを確認します。変更リクエスト、サービスリクエスト、ベンダーサポート、ネットワークとファイアウォールのサポートなど、外部プロセス所有者とのエンゲージメントが必要な場所を含めます。プロセスを理解したら、責任、説明責任、相談、通知 (RACI) マトリックスに所有権を文書化して、さまざまなアクティビティを追跡することをお勧めします。プロセスを確定するには、移行の各ステップに関連するタイムラインを特定してカウントダウン計画を作成します。カウントダウンプランは通常、ワークロード移行のカットオーバー日時から逆算して機能します。

このドキュメントのアプローチは、1 年も経たないうちに AWS 正常に に移行し、4 つのデータセンターを離れた多国籍のホームアプライアンス企業にとって効果的です。6 つの異なる組織チームと複数のサードパーティーが関係していたため、管理オーバーヘッドが発生し、back-and-forthの決定や実装の遅延が発生しました。 AWS プロフェッショナルサービスチームは、顧客およびサードパー

ティーとともに、移行アクティビティの主要なプロセスを特定し、それぞれの所有者に文書化しました。結果として得られた RACI マトリックスは、関係するすべてのチームによって共有され、合意されました。RACI マトリックスとエスカレーションマトリックスを使用して、お客様は遅延を引き起こしていたブロッカーと問題を軽減しました。その後、データセンターをスケジュールよりも前に退出できました。

RACI とエスカレーションマトリックスを使用する別の例では、ある保険会社が 4 か月以内にデータセンターを出ることができました。顧客は責任共有モデルを理解し、実装しました。また、詳細な RACI マトリックスに従って、移行中の各プロセスとアクティビティの進行状況を追跡しました。その結果、お客様は最初の 12 週間の実装で 350 台を超えるサーバーを移行できました。

標準の移行パターンとアーティファクトを文書化する

これは、実装用の Cookie カッタを作成することを前提としています。再利用可能なリファレンス、ドキュメント、ランブック、パターンがスケーリングの鍵です。これらのジャーナルには、将来の移行プロジェクトが再利用して回避できる経験、学習、落とし穴、問題、ソリューションが記されており、移行を大幅に加速します。パターンとアーティファクトは、プロセスを改善し、将来のプロジェクトをガイドするのに役立つ投資でもあります。

例えば、ある顧客が 1 年間の移行を行っていて、3 つの異なる SI AWS パートナーによってアプリケーションが移行されていました。初期段階では、各 AWS パートナーは独自の標準、ランブック、アーティファクトを使用していました。これにより、同じ情報がさまざまな方法で顧客チームに提示される可能性があるため、顧客チームに多くのストレスがかかりました。このような初期の困難の後、お客様は移行に使用するすべてのドキュメントとアーティファクトの一元的な所有権を確立し、推奨される変更を送信するプロセスを確立しました。これらのアセットには以下が含まれます。

- 標準の移行プロセスとチェックリスト
- ネットワーク図のスタイルと形式の標準
- ビジネスの重要性に基づくアプリケーションアーキテクチャとセキュリティ標準

さらに、これらのドキュメントと標準に対する変更は毎週すべてのチームに送信され、各パートナーは変更の受信と準拠を確認する必要がありました。これにより、移行プロジェクトのコミュニケーションと一貫性が大幅に向上し、別のビジネスユニットで別の大規模な移行作業が開始されると、そのチームは既存のプロセスとドキュメントを採用でき、成功が大幅に加速しました。

移行メタデータとステータスの信頼できる単一のソースを確立する

大規模な移行を計画する場合、さまざまなチームの足並みを揃え、データ主導の意思決定を可能にするには、信頼できる情報源を確立することが重要です。このジャーニーを開始すると、設定管理デー

タベース (CMDB)、アプリケーションパフォーマンスモニタリングツール、インベントリリストなど、使用できるデータソースが多数見つかります。

または、データソースが少なく、必要なデータをキャプチャするメカニズムを作成する必要があります。例えば、検出ツールを使用して技術情報を明らかにし、IT リーダーを調査してビジネス情報を取得する必要がある場合があります。

さまざまなデータソースを 1 つのデータセットに集約し、移行に使用できることが重要です。その後、実装中の移行を追跡するために、信頼できる単一のソースを使用できます。例えば、移行されたサーバーを追跡できます。

提供されたデータセットを使用した移行の計画に重点を置いて、 AWS すべてのワークロードを に移行したいと考えている金融サービスの顧客。このデータセットにはビジネスの重要性や依存関係情報などの重要なギャップがあったため、プログラムは検出演習を開始しました。

別の例では、同じ業界の企業が、サーバーインフラストラクチャのインベントリout-of-date理解に基づいて移行ウェーブの実装に移行しました。データが正しくなかったため、すぐに移行数が減り始めました。この場合、アプリケーション所有者は理解されなかったため、テスターを間に合うように見つけられませんでした。さらに、データはアプリケーションチームが完了した廃止に合わせていなかったため、サーバーはビジネス目的で使用されずに実行されていました。

大規模な移行の実行

ビジネス成果を確立し、ステークホルダーに戦略を伝えたら、持続可能な移行イベントやウェーブへの大規模な移行の範囲をどのように計画するかに進むことができます。次の例は、ウェーブプランを作成するための主要なガイダンスを提供します。

このセクションの内容:

- 安定したフローを確保するために、移行ウェーブを事前に計画する
- ウェーブ実装とウェーブ計画を別々のプロセスとチームとして維持する
- 大きな成果を得るために小規模から始める
- カットオーバーウィンドウの数を最小限に抑える
- フェイルファスト、エクスペリエンスの適用、反復
- 遡及的情報を忘れないでください

安定したフローを確保するために、移行ウェーブを事前に計画する

移行の計画は、プログラムの最も重要なフェーズの 1 つです。「計画に失敗した場合、失敗する予定」と表示されます。移行ウェーブを事前に計画しておくと、チームが移行状況に対してより積極的になるにつれて、プロジェクトが迅速に流れるようになります。これにより、プロジェクトのスケーリングが容易になり、プロジェクトの需要が増加し、複雑になるにつれて意思決定と予測が向上します。事前に計画することで、チームが変更に適応する能力も向上します。

例えば、大規模な金融サービスの顧客がデータセンターの出口プログラムに取り組んでいるとします。当初、お客様は移行ウェーブを順番に計画し、1 つのウェーブを完了してから次のウェーブの計画を開始します。このアプローチにより、準備時間が短縮されました。ステークホルダーにアプリケーションが移行されたことが通知されても AWS、移行を開始する前にいくつかのステップを実行する必要があります。これにより、プログラムに大幅な遅延が追加されました。顧客がこれを実現した後、移行の波が数か月前に計画された、包括的で将来の焦点を当てた移行計画ストリームを実装しました。これにより、アプリケーションチームは AWS 、パートナーへの通知、ライセンス分析などの移行前アクティビティを実行するのに十分な通知が提供されました。その後、プログラムのクリティカルパスからこれらのタスクを削除できます。

ウェーブ実装とウェーブ計画を別々のプロセスとチームとして維持する

ウェーブプランニングチームとウェーブ実装チームが分離されている場合、2 つのプロセスは並行して機能します。通信と調整では、十分なサーバーやアプリケーションが期待速度を達成する準備ができていないため、移行の速度が低下しません。例えば、移行チームは毎週 30 台のサーバーを移行する必要があるかもしれませんが、現在のウェーブでは 10 台のサーバーしか準備ができていません。この課題は、多くの場合、以下が原因で発生します。

- 移行実装チームはウェーブプランニングに関与しておらず、ウェーブプランニングフェーズで収集されたデータは完了していません。移行実装チームは、ウェーブを開始する前により多くのサーバーデータを収集する必要があります。
- 移行実装は、ウェーブ計画の直後に開始される予定で、間にバッファはありません。

事前にウェーブを計画し、準備とウェーブ実装の開始の間にバッファを作成することが重要です。また、ウェーブプランニングチームと移行チームが連携して適切なデータを収集し、再作業を避けることも重要です。

大きな成果を得るために小規模から始める

小規模から始めて、その後のウェーブごとに移行速度を上げることを計画します。最初のウェーブは、10 台未満の単一の小さなアプリケーションである必要があります。後続のウェーブにアプリ

ケーションとサーバーを追加し、移行速度を最大限に向上させます。複雑性やリスクの低いアプリケーションに優先順位を付け、スケジュールに従って速度を上げることで、チームは連携作業に適応し、プロセスを学ぶ時間を確保できます。さらに、チームはウェーブごとにプロセスの改善を特定して実装できるため、後のウェーブの速度を大幅に向上させることができます。

1 人のお客様が、年間 1,300 台を超えるサーバーを移行していました。パイロット移行といくつかの小さなウェーブから始めることで、移行チームは、後の移行を改善する複数の方法を特定できました。例えば、以前に新しいデータセンターのネットワークセグメントを特定しました。チームは、プロセスの早い段階でファイアウォールチームと協力して、移行ツールとの通信を可能にするファイアウォールルールを設定しました。これにより、将来のウェーブでの不要な遅延を防ぐことができます。さらに、チームは、各ウェーブで検出とカットオーバーのプロセスをさらに自動化するスクリプトを開発できました。小規模から始めることで、チームは初期のプロセス改善に集中できるようになり、信頼度が大幅に向上しました。

カットオーバーウィンドウの数を最小限に抑える

大規模な移行には、規模を拡大するための統制のとれたアプローチが必要です。一部の分野で柔軟性が高すぎることは、大規模な移行にはアンチパターンです。週単位のカットオーバーウィンドウの数を制限することで、カットオーバーアクティビティに費やされる時間はより大きくなります。

例えば、カットオーバーウィンドウの柔軟性が高すぎる場合、それぞれ 5 台のサーバーで 20 回のカットオーバーが発生する可能性があります。代わりに、それぞれ 50 台のサーバーで 2 つのカットオーバーを行うことができます。各カットオーバーの時間と労力は似ているため、カットオーバーが少なくなるほど、スケジューリングの運用上の負担が軽減され、不要な遅延が制限されます。

ある大規模なテクノロジー企業は、契約の有効期限が切れる前に、リースされたデータセンターから移行しようとしていました。有効期限切れになれば、費用がかかり、短期的な更新期間となります。移行の前半では、アプリケーションチームは、カットオーバーのわずか数日前に何らかの理由で移行をオプトアウトするなど、過去 1 分間の移行スケジュールを決定できました。これにより、プロジェクトの初期段階で多くの遅延が発生しました。多くの場合、顧客は直近 1 分間に他のアプリケーションチームと交渉して入力する必要がありました。顧客は最終的に計画の統制を強化しましたが、この初期のミスにより、移行チームに継続的なストレスがもたらされました。スケジュール全体の遅延により、一部のアプリケーションが時間内にデータセンターから外れることがなくなりました。

フェイルファスト、エクスペリエンスの適用、反復

すべての移行には、最初は落とし穴があります。早期失敗は、チームが学習し、ボトルネックを理解し、学んだ教訓をより大きな波に適用するのに役立ちます。移行の最初の数波は、次の理由で遅くなることが予想されます。

- チームメンバーは互いに、またプロセスに適応しています。
- 大規模な移行には、通常、さまざまなツールや人が関係します。
- end-to-endのプロセスの統合、テスト、失敗、学習、継続的な改善には時間がかかります。

問題は一般的であり、最初の数回のウェーブ中に発生します。一部のチームは新しいことを試して失敗することを好まない可能性があるため、これを理解し、組織全体に伝えることが重要です。障害が発生すると、チームが失望し、将来の移行の妨げになる可能性があります。最初の問題がジョブの一部であることを全員が理解し、全員が試行して失敗するように促すことが、移行を成功させるための鍵となります。

ある企業が、24～36 か月で 10,000 台を超えるサーバーを移行する予定でした。この目標を達成するために、1 か月あたり約 300 台のサーバーを移行する必要がありました。ただし、1 日目から 300 台のサーバーを移行したわけではありません。最初の 2 つのウェーブはウェーブを学習することでした。これにより、チームは作業がどのように機能し、誰が作業を行う権限を持っているかを理解できます。また、CMDB や CyberArk との統合など、プロセスを改善する統合も特定しました。学習ウェーブを使用して失敗、改善、失敗し、プロセスと自動化を改善しました。6 か月後、毎週 120 台以上のサーバーを移行できました。

遡及的情報を忘れないでください

これはアジャイルプロセスの重要な部分です。チームがコミュニケーション、調整、学習、同意、前進する場所です。最も基本的なレベルでの遡及的には、振り返って何が起こったのかを議論し、何がうまくいき、何を改善する必要があるのかを判断します。その後、それらの議論に基づいて改善を構築できます。遡及性は、教訓を学んだ考え方を形式やプロセスでまとめます。大規模な移行を成功させるには、プロセス、ツール、チームが常に進化し、改善する必要があるため、遡及性は重要です。遡及性は、その点で重要な役割を果たします。

従来の教訓セッションはプログラムの最後まで行われないため、多くの場合、これらの教訓は次の移行ウェーブの開始時にレビューされません。大規模な移行では、学んだ教訓を次のウェーブに適用し、ウェーブ計画プロセスの重要な部分となる必要があります。

1 人の顧客について、カットオーバーから学んだ教訓について議論し、文書化するために、毎週の遡及調査が行われました。これらのセッションでは、プロセスの観点から、または自動化の観点から

合理化の範囲がある領域を発見しました。これにより、カットオーバー中のサードパーティーツールの検証や Amazon CloudWatch エージェントのインストールなど、手動タスクを最小限に抑えるために、特定のアクティビティ、所有者、自動化スクリプトを含むカウントダウンスケジュールが実装されました。

別の大規模な技術会社では、以前の移行ウェーブの問題を特定するために、チームと定期的な遡及調査を実施しました。これにより、プロセス、スクリプト、オートメーションが改善され、プログラム全体で平均移行時間が 40% 短縮されました。

追加の考慮事項

多くの分野を大規模な移行プログラムに組み込む必要があります。以下のセクションでは、考慮する必要がある他の項目について説明します。

このセクションの内容:

作業中にクリーンアップする

コストが予想の 10 倍で、移行に使用されるリソースがシャットダウンされてクリーンアップされるまでプロジェクトが完了していない場合、移行は成功とは見なされません。このクリーンアップは、移行後のアクティビティの一部である必要があります。これにより、未使用のリソースやサービスを環境に残して、コストが増加することがなくなります。移行後のクリーンアップは、環境を公開する脅威や脆弱性を防ぐための優れたセキュリティプラクティスでもあります。

への移行の主な成果 AWS クラウド は、コスト削減とセキュリティの 2 つです。未使用のリソースを残すと、クラウドへの移行というビジネス目的が損なわれる可能性があります。クリーンアップされない最も一般的なリソースは次のとおりです。

- テストデータ
- データベースをテストする
- ファイアウォールルール、セキュリティグループ、ネットワークアクセスコントロールリスト (ネットワーク ACL) IP アドレスなどのアカウントをテストする
- テスト用にプロビジョニングされたポート
- Amazon Elastic Block Store (Amazon EBS) ボリューム
- スナップショット

- レプリケーション (オンプレミスから へのデータレプリケーションの停止など AWS)
- 領域を消費するファイル (移行に使用される一時データベースバックアップなど)
- 移行ツールをホストするインスタンス

不正なクリーンアッププラクティスの一例では、SI AWS パートナーは移行が成功した後にレプリケーションエージェントを削除しませんでした。 AWS 監査では、すでに移行されていたレプリケーションサーバーと EBS ボリュームのコストが毎月 20,000 USD (USD) であることが検出されました。この問題を軽減するために、 AWS Professional Services は、古いサーバーがまだレプリケートされているときに SI AWS パートナーに通知する自動監査プロセスを作成しました。SI AWS パートナーは、未使用の古いインスタンスに対してアクションを実行できます。

将来の移行では、プラットフォームのスムーズな導入を確保するために、移行後のハイパーケア期間を 48 時間と定義するプロセスが採用されました。次に、お客様のインフラストラクチャチームがオンプレミスサーバーに対する廃止リクエストを送信しました。廃止リクエストが承認されると、それぞれのウェーブのサーバーがアプリケーション移行サービスコンソールから削除されることがアドバイスされました。

追加の変換のために複数のフェーズを実装する

大規模な移行を実行するときは、データセンターの閉鎖やインフラストラクチャの変革など、中核的な目標に集中し続けることが重要です。小規模な移行では、スコープクリープの影響は最小限に抑えられます。ただし、数日間の追加作業に数千のサーバーを掛けると、プログラムにかなりの時間がかかる可能性があります。さらに、追加の変更により、サポートチームのドキュメント、プロセス、トレーニングの更新が必要になる場合もあります。

潜在的なスコープクリープを克服するために、移行に複数フェーズのアプローチを実装できます。例えば、データセンターを退避することが目標である場合、フェーズ 1 には、ワークロードをできるだけ AWS 速くリホストすることのみが含まれる場合があります。ワークロードがリホストされた後、フェーズ 2 では、目標のビジネス成果をリスクにさらすことなく、変革的なアクティビティを実装できます。

例えば、ある顧客が 12 か月以内にデータセンターを出る予定だったとします。ただし、移行には、新しいアプリケーションパフォーマンスモニタリングツールのロールアウトやオペレーティングシステムのアップグレードなど、他の変換アクティビティが含まれていました。1,000 台を超えるサーバーが移行対象であったため、これらのアクティビティにより移行に大幅な遅延が生じました。さらに、このアプローチでは、新しいツールの使用に関するトレーニングが必要でした。顧客は後で、リホストに重点を置いた複数フェーズのアプローチを実装することにしました。これにより、移行速度が向上し、データセンターの閉鎖日を過ぎないリスクが軽減されました。

結論

大規模な移行では、小規模な移行と比較してさまざまな課題があります。これは主に、スケールによって導入された複雑さが原因です。例えば、エージェントを単一のサーバーにインストールするのは非常に簡単で、約 5 分かかります。ただし、移行の対象となるサーバーが 5,000 台ある場合、これには約 416 時間かかり、次の課題が生じます。

- 異なるプロセスを必要とするオペレーティングシステムが複数ある可能性があります。
- 以前の合併や買収により、管理すべき個別の Microsoft Active Directory ドメインが存在する場合があります。
- ウェーブごとにエージェントのインストールを調整し、進行状況を追跡して報告するには、効果的なプロセスとツールが必要です。

この戦略は、幅広い顧客を支援する AWS プロフェッショナルサービスの経験に基づいて、大規模な移行のベストプラクティスの概要を示しています。これには、人、プロセス、テクノロジーの視点が含まれます。開始する場合や移行中の場合は AWS、 AWS プロフェッショナルサービスのコンサルタントがお手伝いします。担当者に連絡して AWS 会話を開始します。

次のステップでは、 への大規模な移行を計画して完了できるように設計された AWS 規範ガイダンスシリーズを確認することをお勧めします AWS クラウド。完全なシリーズについては、「 への大規模な移行 AWS クラウド」を参照してください。

リソース

AWS 大規模な移行

大規模な移行に関する AWS 完全な 規範ガイダンスシリーズにアクセスするには、「 への大規模な移行 AWS クラウド」を参照してください。

関連する AWS 規範ガイダンスリソース

- クラウド移行ファクトリーによる大規模なサーバー移行の自動化
- への移行中に廃止されるアプリケーションを評価するためのベストプラクティス AWS クラウド
- 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
- 移行の準備状況の評価
- Mobilize your organization to accelerate large-scale migrations

その他のリファレンス

- AWS クラウド移行ファクトリーソリューション
- での無料のクラウド移行サービス AWS
- AWS Database Migration Service
- AWSで移行

動画

- への大規模な移行の実行 AWS (AWS re:Invent 2020)
- CloudEndure Migration Factory のベストプラクティス (AWS re:Invent 2020)

寄稿者

この戦略は、 AWS プロフェッショナルサービス内のグローバル大規模移行タイガーチームによって作成されました。チームは、 AWS お客様 AWS に代わって数千台のサーバーを に正常に移行しました。本ドキュメントの寄稿者は次のとおりです。

- Chris Baker、プリンシパル製品エンジニア
- Dwayne Bordelon、シニアクラウドアプリケーションアーキテクト
- Rodolfo Jr. Cerrada、シニアアプリケーションアーキテクト
- Pratik Chunawala、プリンシパルクラウドアーキテクト
- Bill David、Principal Customer Solutions Manager
- Dev Kar、シニアコンサルタント
- Wally Lu、プリンシパルコンサルタント
- Jon Madison、プリンシパルクラウドアーキテクト
- Abhishek Naik、シニアソリューションアーキテクト
- Damien Renner、シニア移行スペシャリスト
- Amit Rudraraju、シニアクラウドアーキテクト

ドキュメント履歴

次の表に、この戦略の重要な変更を示します。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
CloudEndure Migration サービスを削除	CloudEndure Migration サービスへの参照を削除しました。AWS Application Migration Service は、 へのlift-and-shift 移行に推奨される主要な移行サービスです AWS クラウド。	2022 年 5 月 11 日
AWS ソリューションの名前を更新しました	参照される AWS ソリューションの名前を CloudEndure Migration Factory から Cloud Migration Factory に更新しました。	2022 年 5 月 2 日
更新されたリソース	「Introduction and Resources 」セクションを、大規模な移行シリーズの最新のドキュメントで更新しました。	2022 年 3 月 8 日
初版発行	—	2021 年 9 月 16 日

AWS 規範ガイダンス用語集

以下は、 AWS 規範ガイダンスが提供する戦略、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します。
- 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の最適化レベルを導入します。例: オンプレミスの Oracle データベースを の Oracle 用 Amazon Relational Database Service (Amazon RDS) に移行します AWS クラウド。
- 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: カスタマーリレーションシップ管理 (CRM) システムを Salesforce.com に移行します。
- 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。例: オンプレミスの Oracle データベースを の EC2 インスタンス上の Oracle に移行します AWS クラウド。
- 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。サーバーを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。

- 使用停止 — 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コントロールを参照してください。

抽象化されたサービス

「マネージドサービス」を参照してください。

ACID

アトミック性、一貫性、分離性、耐久性を参照してください。

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。より柔軟ですが、アクティブ/パッシブ移行よりも多くの作業が必要です。

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の方法。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

行のグループで動作し、グループの単一の戻り値を計算する SQL 関数。集計関数の例には、SUMおよび が含まれますMAX。

AI

「人工知能」を参照してください。

AIOps

「人工知能オペレーション」を参照してください。

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。

アプリケーションコントロール

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 の需要要素であり、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?」を参照してください。

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。

アベイラビリティーゾーン

他のアベイラビリティーゾーンの障害から 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ー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)

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は、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 るのに役立ちます。 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

個人や組織を混乱させたり、損害を与えたりすることを意図したボット。

BCP

「事業継続計画」を参照してください。

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Data in a behavior graph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。エンディアン性も参照してください。

二項分類

バイナリ結果 (2 つの可能なクラスのうちの 1 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

2 つの異なる同一の環境を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 (青) 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 (緑) 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、ボットハーダーまたはボットオペレーターとして知られる、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。ボットは、ボットとその影響をスケールするための最もよく知られているメカニズムです。

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。

ブレークグラスアクセス

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なります。詳細については、 Well-Architected ガイダンスの「ブレークグラス手順の実装」インジケータ AWS を参照してください。

ブラウンフィールド戦略

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ン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。

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、ホワイトペーパー AWSでのコンテナ化されたマイクロサービスの実行 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。

Canary デプロイ

エンドユーザーへのバージョンのスローリリースと増分リリース。確信が持てば、新しいバージョンをデプロイし、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

「データキャプチャの変更」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します。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テクノロジー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 に移行するときに通常実行する 4 つのフェーズ AWS クラウド :

- 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

「設定管理データベース」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または が含まれますBitbucket Cloud。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などのビジュアル形式から情報を分析および抽出する AI の分野。例えば、Amazon SageMaker AI は CV 用の画像処理アルゴリズムを提供します。

設定ドリフト

ワークロードの場合、設定は想定状態から変化します。ワークロードが非準拠になる可能性があり、通常は段階的かつ意図的ではありません。

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

一元管理とガバナンスを備えた分散型の分散型データ所有権を提供す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。信頼された ID のみが、期待されるネットワークから信頼された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通常、大量の履歴データが含まれており、通常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る。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。

ディープラーニング

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。

デプロイ

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、Implementing security controls on AWSのDetective controls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ュファクチャリング・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

スタースキーマでは、ファクトテーブル内の量的データに関するデータ属性を含む小さなテーブル。ディメンションテーブル属性は通常、テキストフィールドまたはテキストのように動作する

離散数値です。これらの属性は、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一般的に使用されます。

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)

災害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に使用する戦略とプロセス。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 (ボストン: 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

「開発値ストリームマッピング」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、応答時間を短縮できます。

電子データ交換 (EDI)

組織間の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プロセス。

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使用できる、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

星スキーマの中央テーブル。事業運営に関する量的データを保存します。通常、ファクトテーブルには、メジャーを含む列とディメンションテーブルへの外部キーを含む列の 2 つのタイプの列が含まれます。

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で段階的なテストを使用する哲学。これはアジャイルアプローチの重要な部分です。

障害分離の境界

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ーゾーン 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

同様のタスクの実行を求める前に、タスクと必要な出力を示す少数の例を LLM に提供します。この手法は、プロンプトに埋め込まれた例 (ショット) からモデルが学習するコンテキスト内学習のアプリケーションです。少数ショットプロンプトは、特定のフォーマット、推論、またはドメインの知識を必要とする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コントロール」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

段階的なアプローチを使用する代わりに、変更データキャプチャによる継続的なデータレプリケーションを使用して、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)

一般化およびラベル付けされていないデータの大規模なデータセットでトレーニングされている大規模な深層学習ニューラルネットワーク。FMs は、言語の理解、テキストと画像の生成、自然言語の会話など、さまざまな一般的なタスクを実行できます。詳細については、「基盤モデルとは」を参照してください。

G

生成 AI

大量のデータでトレーニングされ、シンプルなテキストプロンプトを使用してイメージ、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る AI モデルのサブセット。詳細については、「生成 AI とは」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 ワークフローはレガシーと見なされ、トランクベースのワークフローはモダンで推奨されるアプローチです。

ゴールデンイメージ

そのシステムまたはソフトウェアの新しいインスタンスをデプロイするためのテンプレートとして使用されるシステムまたはソフトウェアのスナップショット。例えば、製造では、ゴールデンイメージを使用して複数のデバイスにソフトウェアをプロビジョニングし、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、、Amazon GuardDuty AWS Security Hub、、 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。

ハイアベイラビリティ (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に使用されるデータセットから保留される、ラベル付きの履歴データの一部。モデル予測をホールドアウトデータと比較することで、ホールドアウトデータを使用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

IaC

「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

「産業用モノのインターネット」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

既存のインフラストラクチャを更新、パッチ適用、または変更する代わりに、本番環境のワークロード用に新しいインフラストラクチャをデプロイするモデル。イミュータブルインフラストラクチャは、本質的に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、予測性が高くなります。詳細については、 AWS 「 Well-Architected フレームワーク」の「イミュータブルインフラストラクチャを使用したデプロイ」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。

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 によって導入された用語で、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ゼーションを指します。

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。

産業分野における 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、「Building an industrial Internet of Things (IIoT) digital transformation strategy」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 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

「モノのインターネット」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)

大量のデータに対して事前トレーニングされた深層学習 AI モデル。LLM は、質問への回答、ドキュメントの要約、テキストの他の言語への翻訳、文の完了など、複数のタスクを実行できます。詳細については、LLMs」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

「ラベルベースのアクセスコントロール」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。エンディアン性も参照してください。

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または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、不正アクセスにつながる可能性があります。マルウェアの例としては、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。Amazon Simple Storage Service (Amazon S3) と Amazon DynamoDB は、マネージドサービスの例です。これら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、制御するためのソフトウェアシステムで、原材料を工場の完成製品に変換します。

MAP

「移行促進プログラム」を参照してください。

メカニズム

ツールを作成し、ツールの導入を推進し、調整を行うために結果を検査する完全なプロセス。メカニズムは、動作中にそれ自体を強化して改善するサイクルです。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。

メッセージキューイングテレメトリトランスポート (MQTT)

リソースに制約のある IoT デバイス用の、パブリッシュ/サブスクライブパターンに基づく軽量 machine-to-machine (M2M) 通信プロトコル。

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)

に移行するためのビジネスケースを検証するための情報を提供するオンラインツール AWS クラウド。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 に移行するために使用するアプローチ AWS クラウド。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、「」の「アプリケーションをモダナイズするための戦略 AWS クラウド」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、『』の「アプリケーションのモダナイゼーション準備状況の評価 AWS クラウド」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

「移行ポートフォリオ評価」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。

ミュータブルインフラストラクチャ

本番ワークロードの既存のインフラストラクチャを更新および変更するモデル。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

「オリジンアクセスコントロール」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

「 オペレーションの統合」を参照してください。

OLA

「運用レベルの契約」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。

オープンプロセス通信 - 統合アーキテクチャ (OPC-UA)

産業用オートメーション用のmachine-to-machine (M2M) 通信プロトコル。OPC-UA は、データの暗号化、認証、認可スキームを備えた相互運用性標準を提供します。

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)

インシデントや潜在的な障害の理解、評価、防止、または範囲の縮小に役立つ質問とそれに関連するベストプラクティスのチェックリスト。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するハードウェアおよびソフトウェアシステム。製造では、OT と情報技術 (IT) システムの統合が、Industry 4.0 変換の主な焦点です。

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 のすべてのイベント AWS CloudTrail をログに記録する、 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のため、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。

オリジンアクセスコントロール (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コントロールが可能です。

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

「プログラム可能な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

アクセス許可を定義 (アイデンティティベースのポリシーを参照)、アクセス条件を指定 (リソースベースのポリシーを参照)、または の組織内のすべてのアカウントに対する最大アクセス許可を定義 AWS Organizations （ サービスコントロールポリシーを参照) できるオブジェクト。

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。詳細については、マイクロサービスでのデータ永続性の有効化 を参照してください。

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、「移行準備状況ガイド」を参照してください。

述語

`true` または を返すクエリ条件。一般的に`false`は `WHERE`句にあります。

述語プッシュダウン

転送前にクエリ内のデータをフィルタリングするデータベースクエリ最適化手法。これにより、リレーショナルデータベースから取得して処理する必要があるデータの量が減少し、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、Implementing security controls on AWSのPreventative controls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 のエンティティ。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内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じてプライバシーを考慮するシステムエンジニアリングアプローチ。

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を防ぐ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、プロビジョニング前にリソースをスキャンします。リソースがコントロールに準拠していない場合、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)

設計、開発、発売から成長と成熟まで、製品のデータとプロセスのライフサイクル全体にわたる管理。

本番環境

「環境」を参照してください。

プログラム可能なロジックコントローラー (PLC)

製造では、マシンをモニタリングし、製造プロセスを自動化する、信頼性の高い適応可能なコンピュータです。

プロンプトの連鎖

1 つの LLM プロンプトの出力を次のプロンプトの入力として使用して、より良いレスポンスを生成します。この手法は、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善または拡張したりするために使用されます。これにより、モデルのレスポンスの精度と関連性が向上し、より詳細でパーソナライズされた結果が得られます。

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。

パブリッシュ/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。スケーラビリティと応答性を向上させます。たとえば、マイクロサービスベースの MES では、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できるチャネルにイベントメッセージを発行できます。システムは、公開サービスを変更せずに新しいマイクロサービスを追加できます。

Q

クエリプラン

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る手順などの一連のステップ。

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

責任、説明責任、相談、通知 (RACI) を参照してください。

RAG

「取得拡張生成」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

責任、説明責任、相談、情報 (RACI) を参照してください。

RCAC

「行と列のアクセスコントロール」を参照してください。

リードレプリカ

読み取り専用に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。

再設計

「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)

サービスの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 とは独立しています。詳細については、AWS リージョン「アカウントで使用できる を指定する」を参照してください。

回帰

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。

プラットフォーム変更

「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

中断に抵抗または回復するアプリケーションの機能。高可用性とディザスタリカバリは、 で回復性を計画する際の一般的な考慮事項です AWS クラウド。詳細については、AWS クラウド「レジリエンス」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。サポートを含めると、そのマトリックスは RASCI マトリックスと呼ばれ、サポートを除外すると RACI マトリックスと呼ばれます。

レスポンシブコントロール

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、Implementing security controls on AWSのResponsive controls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。

取得拡張生成 (RAG)

LLM がレスポンスを生成する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を参照する生成 AI テクノロジー。たとえば、RAG モデルは、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する場合があります。詳細については、「RAG とは」を参照してください。

ローテーション

攻撃者が認証情報にアクセスすることをより困難にするために、シークレットを定期的に更新するプロセス。

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、フェデレーティ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 Management Console 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

「監視コントロールとデータ取得」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。シークレット値は、バイナリ、1 つの文字列、または複数の文字列にすることができます。詳細については、Secrets Manager ドキュメントの「Secrets Manager シークレットの内容」を参照してください。

設計によるセキュリティ

開発プロセス全体でセキュリティを考慮するシステムエンジニアリングアプローチ。

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、予防的、検出的、応答的、プロアクティブの 4 つの主なタイプがあります。

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に自動的に応答または修復するように設計された、事前定義された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。自動応答アクションの例としては、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

のエントリポイントの 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、スループットなど、サービスのパフォーマンス側面の測定。

サービスレベルの目標 (SLO)

サービスレベルのインジケータによって測定される、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

セキュリティ情報とイベント管理システムを参照してください。

単一障害点 (SPOF)

システムを中断する可能性のあるアプリケーションの 1 つの重要なコンポーネントの障害。

SLA

「サービスレベルの契約」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、『』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 AWS クラウド」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

1 つの大きなファクトテーブルを使用してトランザクションデータまたは測定データを保存し、1 つ以上の小さなディメンションテーブルを使用してデータ属性を保存するデータベース組織構造。この構造は、データウェアハウスまたはビジネスインテリジェンスの目的で使用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ーゾーンに存在する必要があります。

監視制御とデータ収集 (SCADA)

製造では、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稼働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潜在的な問題を検出したり、パフォーマンスをモニタリングしたりする方法でシステムをテストします。Amazon CloudWatch Synthetics を使用して、これらのテストを作成できます。

システムプロンプト

LLM にコンテキスト、指示、またはガイドラインを提供して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、ユーザーとのやり取りのルールを確立するのに役立ちます。

T

tags

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の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 枚のピザで養う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する行のグループに対して計算を実行する SQL 関数。ウィンドウ関数は、移動平均の計算や、現在の行の相対位置に基づく行の値へのアクセスなどの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 「ワークロード認定フレームワーク」を参照してください。

Write Once, Read Many (WORM)

データを 1 回書き込み、データの削除や変更を防ぐストレージモデル。承認されたユーザーは、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

ゼロデイ脆弱性を利用する攻撃、通常はマルウェア。

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 にタスクを実行する手順を提供しますが、タスクのガイドに役立つ例 (ショット) はありません。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。