



AWS クラウド移行用のアプリケーションポートフォリオ評価ガイド

AWS 規範ガイド



AWS 規範ガイド: AWS クラウド移行用のアプリケーションポートフォリオ評価ガイド

Copyright © 2024 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
発見の加速と初期計画	4
初期評価データ要件を理解する	4
データソースとデータ要件	4
検出ツールの必要性の評価	16
ビジネスドライバーと技術ガイドの原則	21
ビジネスドライバー	21
テクニカルガイドの原則	22
データ収集の開始	24
優先順位付けと移行戦略	25
アプリケーションの優先順位付け	26
移行用の R タイプの決定	28
添付ファイル	30
方向性のあるビジネスケースの作成	30
方向性のあるビジネスケースの範囲の修正	31
フォーカス値ドライバー	32
データのニーズ	32
インフラストラクチャの TCO 比較の構築	33
運用コスト最適化の構築	34
完全な方向性のあるビジネスケースへの拡張	36
移行とモダナイゼーションプログラムのセットアップの見積もり	38
優先順位付けされたアプリケーション評価	48
詳細な評価データ要件を理解する	48
詳細なアプリケーション評価	58
全般	59
アーキテクチャ	60
オペレーション	60
パフォーマンス	61
ソフトウェアのライフサイクル	61
移行	61
回復性	61
セキュリティとコンプライアンス	62
データベース	62

依存関係	62
AWS アプリケーション設計と移行戦略	63
アプリケーションの将来の状態	64
再現性	65
要件	65
To-Be アーキテクチャ	65
アーキテクチャ上の意思決定	68
ソフトウェアライフサイクル環境	68
タグ付け	68
移行戦略	68
移行パターンとツール	69
サービス管理と運用	70
カットオーバーに関する考慮事項	70
リスク、前提条件、問題、依存関係	70
実行コストの見積もり	70
.....	71
完全な評価データ要件を理解する	71
アプリケーションポートフォリオのベースラインの確立	82
優先順位付け基準の反復	84
6 R の移行戦略の選択を繰り返す	86
ウェーブプランニング	87
ウェーブプランの作成	89
変更の管理	91
詳細なビジネスケース	91
ケースに必要なシナリオを決定する	92
インフラストラクチャと移行のコストモデルを検証して改善する	93
IT 生産性と IT 運用を改良し、効率価値モデルをサポートする	94
レジリエンスバリューモデルを開発する	101
ビジネスの俊敏性価値モデルを開発する	103
継続的な評価と改善	105
継続的評価データ要件の理解	106
詳細なウェーブアセスメント	106
最適化と最新化のための評価	106
ウェーブプランの繰り返し	107
ビジネスケースの進化と追跡	108
リソース	110

ドキュメント履歴	112
用語集	113
#	113
A	114
B	116
C	118
D	121
E	125
F	127
G	129
H	129
I	131
L	133
M	134
O	138
P	141
Q	143
R	144
S	146
T	150
U	151
V	152
W	152
Z	153
.....	clv

AWS クラウド移行のためのアプリケーションポートフォリオ評価ガイド

ドイツのゴンカルブスとマーク・バーナー、アマゾン ウェブ サービス (AWS)

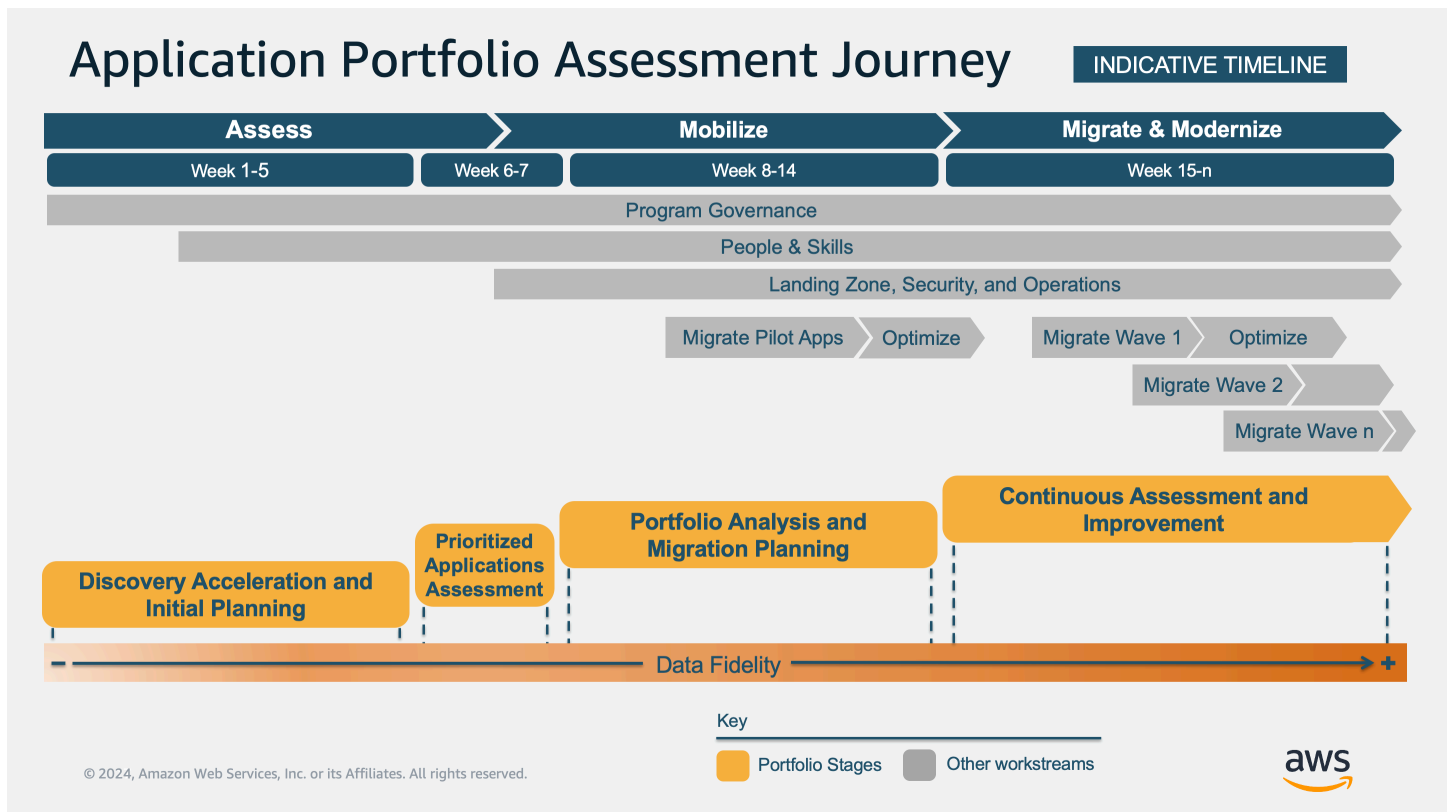
2024 年 5 月 ([ドキュメント履歴](#))

この Amazon Web Services (AWS) 規範ガイダンスドキュメントでは、[アプリケーションポートフォリオ評価戦略](#)の実装について詳しく説明します。このガイドは、アプリケーションおよび関連するインフラストラクチャのポートフォリオの評価を開始して進めるのに役立ちます。評価には、検出、分析、計画が含まれます。インフラストラクチャには、コンピューティング、ストレージ、ネットワークが含まれます。

概要

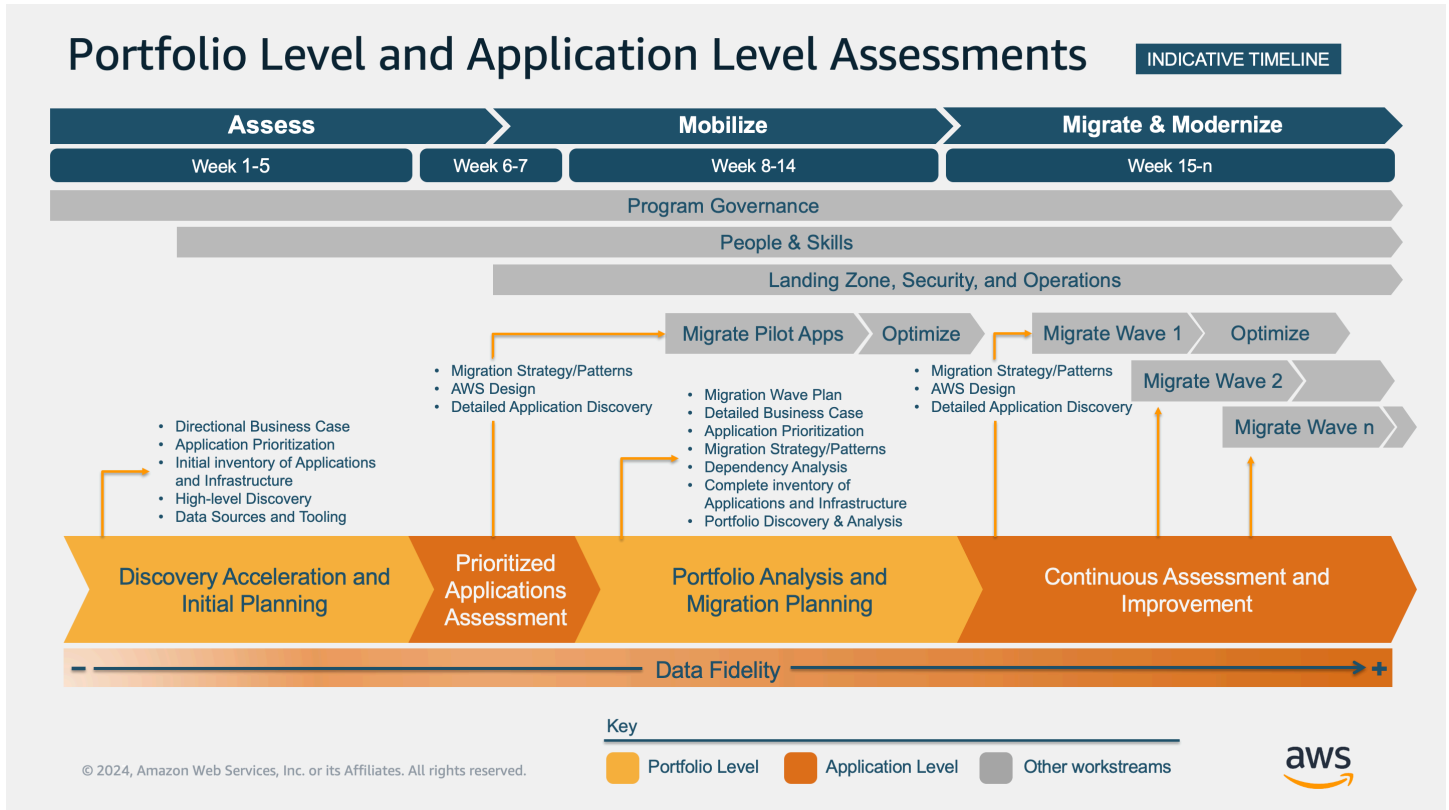
長時間実行されるクラウド移行プログラムでは、プログラムガバナンス、ランディングゾーン (セキュリティコントロールを備えた運用ターゲット環境)、移行、アプリケーションポートフォリオなど、複数のワークストリームを調整する必要があります。これらのワークストリームの名前は、移行プログラムの編成方法によって異なる場合があります。ワークストリームとして、アプリケーションポートフォリオ評価は、これらのプログラムのライフサイクル全体にわたる基本的なアクティビティを表します。評価を通じて得られるポートフォリオを理解することで、継続的なアプリケーションポートフォリオ評価から得られたデータと分析に依存する他のワークストリームへの重要な入力を得られます。

次の図は、ポートフォリオ評価のステージが AWS 移行やその他のワークストリームのフェーズにどのように対応しているかを示しています。ポートフォリオの発見と初期計画段階は、通常最初の 5 週間に評価段階から始まります。優先順位付けされたアプリケーション評価は、6 週間目と 7 週間目に評価フェーズと動員フェーズにまたがります。ポートフォリオ分析と移行計画の段階は、第 8 ~ 14 週の動員段階で行われます。継続的な評価と改善の段階は、移行とモダナイズの段階で、第 15 週から移行プログラムの終了まで行われます。このタイムラインは参考です。ステージの実際の期間は、プログラム組織全体によって異なります。ポートフォリオ評価ステージは、このフレームワークの外部でも有効であり、あらゆる移行プログラム構造に組み込むことができます。



- ディスカバリーアクセラレーションと初期計画は、ポートフォリオの現在の理解に焦点を当てています。これには、方向性のあるビジネスケースの作成、移行の基本合理化モデルの確立、初期移行候補の特定が含まれます。
- 優先順位付けされたアプリケーション評価は、詳細な評価、ターゲット状態アーキテクチャの初期設計、短期的に移行できるアプリケーションの特定 time-to-value を通じて、より高速に提供されます。アプリケーションを迅速に移行することで、チームは移行の経験を積むことができ、初期ランディングゾーンやその他のインフラストラクチャコンポーネントなどのクラウド基盤を確立できます。
- ポートフォリオ分析と移行計画は、アプリケーションポートフォリオの完全かつ明確な up-to-date ビューを構築することに重点を置いています。このビューは、ポートフォリオデータセットを繰り返し強化し、データギャップを埋め、ビジネスケースを進化させ、信頼性の高い移行ウェブプランを作成することによって構築されます。
- 継続的な評価と改善は、継続的なアクティビティとして各移行ウェーブの詳細なアプリケーションとテクノロジー評価を作成することで、大規模な移行をサポートします。この段階には、移行ウェブプランの反復と、移行されたワークロードの最適化とモダナイゼーションのためのさらなる分析が含まれます。

次の図は、評価の各段階の主要なアクティビティと、ポートフォリオレベルの評価とアプリケーションレベルの評価をピボットする方法を示しています。ポートフォリオレベルの評価では、ポートフォリオの高レベルの検出と全体的な分析に焦点を当てます。例えば、ポートフォリオデータのソース、アプリケーションとインフラストラクチャのインベントリ、優先順位付け、方向性のあるビジネスケースなどです。アプリケーションレベルの評価では、1つ以上のアプリケーションの詳細な検出に焦点を当てます。例えば、アプリケーションのアーキテクチャとテクノロジーレベルでの詳細なアプリケーション検出、ターゲット AWS 設計、移行戦略などです。ポートフォリオレベルとアプリケーションレベルの評価は、必要な情報の幅と深さを表します。



発見の加速と初期計画

このポートフォリオ評価の最初の段階は、ポートフォリオレベルでデータを取得して分析する最初のステップに焦点を当てています。主な目的は、ビジネスドライバーを特定し、アプリケーションとインフラストラクチャから一般的なデータを収集して、ポートフォリオの初期ビューを取得することです。このデータには、「[データ要件](#)」セクションで説明されているように、アプリケーション名、環境、製品バージョン、重要度、パフォーマンス値など、高度な技術的属性とビジネス属性が含まれます。この段階を完了することは、プロジェクトの範囲を理解し、初期移行候補を特定し、ビジネスケースを通知する上で重要です。

このステージの主な結果

- 文書化されたビジネス推進要因、成果、目標、技術指針の原則。
- アプリケーションとインフラストラクチャの初期インベントリ、および特定されたデータギャップ。これは、さらに段階的に反復および改良されるポートフォリオの初期ビューです。
- 方向性のあるビジネスケースと移行にかかる推定コスト。
- 初期移行候補のリスト (例: 3-5 アプリケーション)。
- 定義された次のステップ

初期評価データ要件を理解する

データ収集にはかなりの時間がかかり、どのデータが必要でいつ必要かが明確でない場合に、簡単にブロックされる可能性があります。重要なのは、このステージの結果には、少なすぎるデータと多すぎるデータのバランスを理解することです。このポートフォリオ評価の初期段階に必要なデータと忠実度レベルに焦点を当てるには、データ収集に反復的なアプローチを採用します。

データソースとデータ要件

最初のステップは、データソースを特定することです。まず、データ要件を満たすことができる組織内の主要な利害関係者を特定します。これらは通常、サービス管理、運用、キャパシティプランニング、モニタリング、サポートの各チームのメンバーであり、アプリケーション所有者でもあります。これらのグループのメンバーとの作業セッションを確立します。データ要件を伝え、データを提供できるツールと既存のドキュメントのリストを取得します。

これらの会話をガイドするには、次の一連の質問を使用します。

- 現在のインフラストラクチャとアプリケーションのインベントリはどの程度正確で最新ですか？
例えば、会社設定管理データベース (CMDB) の場合、ギャップがどこにあるかはわかっていますか？
- CMDB (または同等のもの) を更新し続けるアクティブなツールやプロセスはありますか？ その場合、どのくらいの頻度で更新されますか？ 最新の更新日はいつですか？
- CMDB などの現在のインベントリには application-to-infrastructure マッピングが含まれていますか？ 各インフラストラクチャアセットはアプリケーションに関連付けられていますか？ 各アプリケーションはインフラストラクチャにマッピングされていますか？
- インベントリには、各製品のライセンスとライセンス契約のカatalogが含まれていますか？
- インベントリには依存関係データが含まれていますか？ サーバーからサーバー、アプリケーションからアプリケーション、アプリケーション、サーバーからデータベースへの通信データが存在することに注意してください。
- アプリケーションとインフラストラクチャの情報を提供できるその他のツールは、環境で利用できますか？ データソースとして使用できるパフォーマンス、モニタリング、および管理ツールが存在することに注意してください。
- アプリケーションやインフラストラクチャをホストするデータセンターなど、さまざまな場所は何ですか？

これらの質問に回答したら、特定されたデータソースを一覧表示します。次に、各に忠実度レベルまたは信頼レベルを割り当てます。ツールなどのアクティブなプログラムによるソースから最近 (30 日以内に) 検証されたデータは、最高レベルの忠実度を持ちます。静的データは忠実度が低く、信頼度が低いと見なされます。静的データの例としては、ドキュメント、ワークブック、手動で更新された CMDBs、プログラムで管理されていないその他のデータセット、または最終更新日が 60 日を超えるデータセットなどがあります。

次の表のデータ忠実度レベルを例として示します。仮定や関連するリスクに対する最大の許容度の観点から組織の要件を評価し、適切な忠実度レベルを決定することをお勧めします。表では、組織的知識とは、文書化されていないアプリケーションとインフラストラクチャに関する情報を指します。

データソース	忠実度レベル	ポートフォリオカバーレッジ	コメント
専門知識	低 - 正確なデータの最大 25%、75% の想定	低	重要アプリケーションに重点を置いた希少

データソース	忠実度レベル	ポートフォリオカバレッジ	コメント
	値、またはデータが 150 日より古い。		
ナレッジベース	中低 - 正確なデータの 35~40%、65~60% の想定値、またはデータが 120~150 日経過しています。	中程度	手動で管理され、詳細レベルに一貫性がない
CMDB	中 - 正確なデータの 50%、想定値またはデータの 50% が 90~120 日経過しています。	中程度	混合ソースからのデータ、複数のデータギャップを含む
VMware vCenter のエクスポート	中高 - 正確なデータの 75~80%、25~20% の想定値、またはデータが 60~90 日経過しています。	高い	仮想化資産の 90% をカバー
アプリケーションパフォーマンスのモニタリング	高 - ほぼ正確なデータ、約 5% の想定値、またはデータが 0~60 日経過しています。	低	重要な本番稼働システムに限定 (アプリケーションポートフォリオの 15% をカバー)

次の表は、各アセットクラス (アプリケーション、インフラストラクチャ、ネットワーク、移行) に必要なデータ属性とオプションのデータ属性、特定のアクティビティ (インベントリまたはビジネスケース)、およびこの評価ステージで推奨されるデータ忠実度を示しています。テーブルでは、次の略語を使用します。

- 必須の R
- (D)、方向性のあるビジネスケースの場合、総所有コスト (TCO) の比較と方向性のあるビジネスケースに必要

- (F) 完全な方向性のあるビジネスケースの場合、TCO 比較と移行コストやモダナイゼーションコストを含む方向性のあるビジネスケースに必要
- O、オプション
- 該当なし、該当なし

アプリケーション

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
一意の識別子	例えば、アプリケーション ID などです。通常、既存の CMDBs やその他の内部インベントリや管理システムで使用できます。一意の IDs 組織で定義されていない場合は必ず作成することを検討してください。	R	R (D)	高い
アプリケーション名	このアプリケーションが組織で認識される名前。必要に応じて、商用 off-the-shelf (COTS) ベンダーと製品名を含めます。	R	R (D)	やや高い
COTS ですか？	はい、いいえ。これは商用アプリ	R	R (D)	やや高い

属性名	説明	インベントリと 優先順位付け	ビジネスケース	推奨忠実度レベ ル (最小)
	リケーションか 内部開発か			
COTS 製品と バージョン	商用ソフトウェ ア製品名とバー ジョン	R	R (D)	中程度
説明	プライマリアプ リケーションの 関数とコンテキ スト	R	O	中程度
緊急性	例えば、戦略的 アプリケーションや収益を生み 出すアプリケーション、重要な 機能のサポート などです。	R	O	やや高い
タイプ	データベース、 顧客関係管理 (CRM)、ウェブ アプリケーション、マルチメ ディア、IT 共有 サービスなど	R	O	中程度
環境	例えば、本番稼 働用、本番稼働 前、開発、テス ト、サンドボツ クスなど	R	R (D)	やや高い

属性名	説明	インベントリと 優先順位付け	ビジネスケース	推奨忠実度レベ ル (最小)
コンプライアンスと規制	ワークロードに適用されるフレームワーク (HIPAA、Sox、PCI-DSS、ISO、SOC、FedRAMP など) と規制要件	R	R (D)	やや高い
依存関係	内部および外部のアプリケーションまたはサービスへのアップストリームとダウンストリームの依存関係。運用要素 (メンテナンスサイクルなど) などの非技術的な依存関係	O	O	やや低い
インフラストラクチャマッピング	アプリケーションを構成する物理アセットや仮想アセットへのマッピング	O	O	中程度
ライセンス	商品ソフトウェアライセンスタイプ (Microsoft SQL Server Enterprise など)	O	R	やや高い

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
コスト	ソフトウェアライセンス、ソフトウェアオペレーション、メンテナンスのコスト	該当なし	O	中程度

インフラストラクチャ

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
一意の識別子	例えば、サーバー ID などです。通常、既存の CMDBs またはその他の内部インベントリと管理システムで使用できます。一意の IDs 組織で定義されていない場合は必ず作成することを検討してください。	R	R	高い
ネットワーク名	ネットワーク内のアセット名 (ホスト名など)	R	O	やや高い

DNS 名 (完全修飾ドメイン名、または FQDN)	[DNS 名]	O	O	中程度
IP アドレスとネットマスク	内部および/またはパブリック IP アドレス	R	O	やや高い
アセットタイプ	物理サーバーまたは仮想サーバー、ハイパーバイザー、コンテナ、デバイス、データベースインスタンスなど。	R	R	やや高い
製品名	商用ベンダーと製品名 (VMware ESXi、IBM Power Systems、Exadata など)	R	R	中程度
オペレーティングシステム	例えば、REHL 8、Windows Server 2019、AIX 6.1 などです。	R	R	やや高い
構成	割り当てられた CPU、コア数、コアあたりのスレッド数、合計メモリ、ストレージ、ネットワークカード	R	R	やや高い

使用率	CPU、メモリ、ストレージのピークと平均。データベースインスタンスのスループット。	R	O	やや高い
ライセンス	商品ライセンスタイプ (RHEL Standard など)	R	R	中程度
共有インフラストラクチャですか？	認証プロバイダー、モニタリングシステム、バックアップサービス、および同様のサービスなどの共有サービスを提供するインフラストラクチャサービスを示すにはいまたはいいえ	R	R (D)	中程度
アプリケーションマッピング	このインフラストラクチャで実行されるアプリケーションまたはアプリケーションコンポーネント	O	O	中程度

コスト	ハードウェア、メンテナンス、オペレーション、ストレージ (SAN、NAS、オブジェクト)、オペレーティングシステムライセンス、ラックスペースの共有、データセンターのオーバーヘッドなど、ベアメタルサーバーのフルロードコスト	該当なし	O	やや高い
-----	--	------	---	------

ネットワーク

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
パイプのサイズ (MB/秒)、冗長性 (Y/N)	現在の WAN リンク仕様 (1000 Mb/秒冗長など)	O	R	中程度
リンク使用率	ピーク使用率と平均使用率、アウトバウンドデータ転送 (GB/月)	O	R	中程度
レイテンシー (ms)	接続されたロケーション間の現在のレイテンシー。	O	O	中程度

コスト	現在の 1 か月あたりのコスト	該当なし	O	中程度
移行				
属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
リホスト	各ワークロード (人日)、1 日あたりのカスタマーとパートナーのコストレート、ツールコスト、ワークロード数に関するカスタマーとパートナーの労力	該当なし	R (F)	やや高い
リプラットフォーム	各ワークロード (人日)、1 日あたりの顧客およびパートナーのコストレート、ワークロード数に関する顧客およびパートナーの労力	該当なし	R (F)	やや高い
リファクタリング	各ワークロード (人日)、1 日あたりの顧客およびパートナーのコストレート、ワークロード数に関する顧客お	該当なし	O	やや高い

	よびパートナー の労力			
リタイア	サーバー数、平 均廃止コスト	該当なし	O	やや高い
ランディング ゾーン	既存の (はい/ いいえ) の再利 用、必要な AWS リージョンのリ スト、コスト	該当なし	R (F)	やや高い
人と変化	クラウド運用と 開発でトレーニ ングするスタッ フ数、1 人あた りのトレーニン グコスト、1 人 あたりのトレー ニング時間のコ スト	該当なし	R (F)	やや高い
duration	対象範囲内の ワークロード移 行の期間 (月)	O	R (F)	やや高い
並列コスト	移行中に現状の ままコストを削 除できる時間枠 とレート	該当なし	O	やや高い
	移行中に AWS 製品やサービ ス、およびその 他のインフラス トラクチャコス トが導入される 時間枠とレート	該当なし	O	やや高い

検出ツールの必要性の評価

組織には検出ツールが必要ですか？ポートフォリオ評価には、アプリケーションとインフラストラクチャに関する高い信頼性、up-to-date データが必要です。ポートフォリオ評価の初期段階では、仮定を使用してデータギャップを埋めることができます。

ただし、進捗に応じて、忠実度の高いデータにより、移行計画を正常に作成し、ターゲットインフラストラクチャを正しく推定してコストを削減し、メリットを最大化できます。また、依存関係を考慮し、移行の落とし穴を回避する実装を有効にすることで、リスクを軽減します。クラウド移行プログラムの検出ツールの主なユースケースは、以下を通じてリスクを軽減し、データの信頼度を高めることです。

- 自動またはプログラムによるデータ収集により、検証済みで信頼性の高いデータが得られます。
- データの取得速度の加速、プロジェクトの速度の向上、コストの削減
- CMDB では通常利用できない通信データや依存関係など、データの完全性のレベルCMDBs向上
- 自動アプリケーション識別、TCO 分析、予測実行率、最適化の推奨事項などのインサイトの取得
- 信頼性の高い移行ウェーブプランニング

システムが特定の場所に存在するかどうか不明な場合、ほとんどの検出ツールはネットワークサブネットをスキャンし、ping または Simple Network Management Protocol (SNMP) リクエストに回答するシステムを検出できます。すべてのネットワークまたはシステム設定で ping または SNMP トラフィックが許可されるわけではありません。これらのオプションについて、ネットワークチームや技術チームと話し合います。

アプリケーションポートフォリオの評価と移行のさらなる段階は、正確な依存関係マッピング情報に大きく依存しています。依存関係マッピングは、に必要な AWS インフラストラクチャと設定 (セキュリティグループ、インスタンスタイプ、アカウントの配置、ネットワークルーティングなど) を理解します。また、同時に移動する必要があるアプリケーション (低レイテンシーネットワーク経由で通信する必要があるアプリケーションなど) のグループ化にも役立ちます。さらに、依存関係マッピングはビジネスケースを進化させるための情報を提供します。

検出ツールを決定するときは、評価プロセスのすべての段階を検討し、データ要件を予測することが重要です。データギャップはブロックになる可能性があるため、将来のデータ要件とデータソースを分析してそれらを予測することが重要です。この分野での経験から、ほとんどの停止した移行プロジェクトには、対象範囲内のアプリケーション、関連するインフラストラクチャ、およびそれらの依存関係が明確に識別されないデータセットが限られていることがわかります。この識別の欠如は、誤った

メトリクス、決定、遅延につながる可能性があります。データの取得 up-to-date は、移行プロジェクトを成功させるための最初のステップです。

検出ツールの選択方法

市場内のいくつかの検出ツールは、さまざまな機能と機能を提供します。要件を検討してください。また、組織に最適なオプションを決定します。移行用の検出ツールを決定する際の最も一般的な要因は次のとおりです。

セキュリティ

- ツールデータリポジトリまたは分析エンジンにアクセスするための認証方法は何か？
- データにアクセスできるユーザーと、ツールにアクセスするためのセキュリティコントロールは何か？
- ツールによるデータ収集方法 専用の認証情報が必要ですか？
- ツールがシステムにアクセスしてデータを取得するには、どの認証情報とアクセスレベルが必要ですか？
- ツールコンポーネント間でのデータ転送方法
- このツールは、保管中および転送中のデータ暗号化をサポートしていますか？
- データは環境内外の単一のコンポーネントに一元化されていますか？
- ネットワークとファイアウォールの要件は何か？

セキュリティチームが検出ツールに関する初期の会話に関与していることを確認します。

データ主権

- データはどこに保存され、処理されますか？
- このツールは Software as a Service (SaaS) モデルを使用していますか？
- 環境の境界内にすべてのデータを保持する可能性はありますか？
- 組織の境界を離れる前にデータをスクリーンできますか？

データレジデンシー要件の観点から、組織のニーズを考慮してください。

アーキテクチャ

- どのようなインフラストラクチャが必要で、どのようなコンポーネントが必要ですか？
- 複数のアーキテクチャを利用できますか？

- このツールは、エアロックされたセキュリティゾーンへのコンポーネントのインストールをサポートしていますか？

パフォーマンス

- データ収集がシステムに与える影響

互換性と範囲

- このツールは、私の製品とバージョンのすべてまたはほとんどをサポートしていますか？ ツールドキュメントを確認して、サポート対象プラットフォームがスコープに関する現在の情報と照合されていることを確認します。
- ほとんどのオペレーティングシステムはデータ収集でサポートされていますか？ オペレーティングシステムのバージョンがわからない場合は、検出ツールのリストを、サポートされているシステムの範囲が広いものに絞り込もうとします。

収集方法

- ツールでは、ターゲットシステムごとに エージェントをインストールする必要がありますか？
- エージェントレスデプロイをサポートしていますか？
- エージェントとエージェントレスは同じ機能を提供しますか？
- 収集プロセスとは

特徴

- 利用できる機能は何ですか？
- 総所有コスト (TCO) と推定 AWS クラウド実行率を計算できますか？
- 移行計画をサポートしていますか？
- パフォーマンスを測定しますか？
- ターゲット AWS インフラストラクチャを推奨できますか？
- 依存関係マッピングは実行されていますか？
- どのレベルの依存関係マッピングが提供されますか？
- API アクセスを提供しますか？ (例えば、データを取得するためにプログラムでアクセスできますか?)

強力なアプリケーションとインフラストラクチャの依存関係マッピング機能を持つツールと、通信パターンからアプリケーションを推測できるツールを検討してください。

コスト

- ライセンスモデルとは
- ライセンスの料金はいくらですか？
- 各サーバーの料金は ですか？ 階層型料金ですか？
- オンデマンドでライセンスできる機能が限られているオプションはありますか？

検出ツールは、通常、移行プロジェクトのライフサイクル全体で使用されます。予算が限られている場合は、少なくとも 6 か月を検討してください。ただし、検出ツールがないと、通常、手動作業と内部コストが増加します。

サポートモデル

- デフォルトでは、どのレベルのサポートが提供されますか？
- サポートプランはありますか？
- インシデントへの対応時間はどれくらいですか？

プロフェッショナルサービス

- ベンダーは検出結果を分析するためのプロフェッショナルサービスを提供していますか？
- このガイドの要素について説明できますか？
- ツール + サービスに割引やバンドルはありますか？

検出ツールの推奨機能

複数のツールからのデータのプロビジョニングと結合を時間の経過とともに回避するには、検出ツールで次の最小機能をカバーする必要があります。

- ソフトウェア — 検出ツールは、実行中のプロセスとインストールされているソフトウェアを特定できる必要があります。
- 依存関係マッピング — ネットワーク接続情報を収集し、サーバーと実行中のアプリケーションのインバウンドとアウトバウンドの依存関係マップを構築できる必要があります。また、検出ツールは、通信パターンに基づいてインフラストラクチャのグループからアプリケーションを推測できる必要があります。

- プロファイルと設定の検出 — CPU ファミリー (x86、PowerPC など)、CPU コア数、メモリサイズ、ディスク数とサイズ、ネットワークインターフェイスなどのインフラストラクチャプロファイルをレポートできるはずです。
- ネットワークストレージ検出 — ネットワーク接続ストレージ (NAS) からネットワーク共有を検出してプロファイリングできる必要があります。
- パフォーマンス — CPU、メモリ、ディスク、ネットワークのピーク使用率と平均使用率をレポートできるはずです。
- ギャップ分析 — データ量と忠実度に関するインサイトを提供できるはずです。
- ネットワークスキャン — ネットワークサブネットをスキャンし、不明なインフラストラクチャアセットを検出できる必要があります。
- レポート — 収集と分析のステータスを提供できる必要があります。
- API アクセス — 収集されたデータにアクセスするためのプログラムによる手段を提供できる必要があります。

考慮すべき追加機能

- TCO 分析により、現在のオンプレミスコストと予測コストの AWS コストを比較できます。
- リホストおよびリプラットフォームシナリオにおける Microsoft SQL Server および Oracle システムのライセンス分析と最適化の推奨事項。
- 移行戦略の推奨事項 (検出ツールは、現在のテクノロジーに基づいてデフォルトの移行 R タイプの推奨事項を作成できますか?)
- インベントリのエクスポート (CSV または同様の形式)
- 適切なサイズのレコメンデーション (例えば、レコメンデーションターゲット AWS インフラストラクチャをマッピングできますか?)
- 依存関係の視覚化 (依存関係マッピングをグラフィカルモードで視覚化できるかなど)
- アーキテクチャビュー (例えば、アーキテクチャ図を自動的に作成できますか?)
- アプリケーションの優先順位付け (移行の優先順位付け基準を作成するために、アプリケーション属性とインフラストラクチャ属性に重みや関連性を割り当てることはできますか?)
- ウェーブプランニング (アプリケーションの推奨グループや移行ウェーブプランを作成する機能など)
- 移行コストの見積もり (移行作業の見積もり)

デプロイに関する考慮事項

検出ツールを選択して調達したら、以下の質問を検討して、組織内でのツールのデプロイを担当するチームとの会話を促進します。

- サーバーまたはアプリケーションはサードパーティーによって運用されていますか？ これにより、チームはとプロセスが関与するように指示される可能性があります。
- 検出ツールのデプロイの承認を取得するための大まかなプロセスは何ですか？
- サーバー、コンテナ、ストレージ、データベースなどのシステムにアクセスするための主な認証プロセスは何ですか？ サーバー認証情報はローカルですか、それとも一元管理されていますか？ 認証情報を取得するプロセスは何ですか？ システムからデータを収集するには、認証情報が必要です (コンテナ、仮想サーバーまたは物理サーバー、ハイパーバイザー、データベースなど)。検出ツールが各アセットに接続するための認証情報を取得するのは、特にこれらのアセットが一元化されていない場合、難しい場合があります。
- ネットワークセキュリティゾーンの概要は何ですか？ ネットワーク図は利用できますか？
- データセンターでファイアウォールルールをリクエストするプロセスは何ですか？
- データセンターの運用 (検出ツールのインストール、ファイアウォールリクエスト) SLAs) は何ですか？

ビジネスドライバーと技術ガイドの原則

ビジネスドライバー

組織が既にクラウドへの移行を決定しているか、その決定に近づいているかにかかわらず、クラウド移行のビジネス推進要因を定義して文書化することで、移行の理由が明確になります。その理由が文書化されたら、移行する対象とその移行方法を定義できます。このアクティビティは重要です。次のステップを知らせ、ガイドするために、プロセスのできるだけ早い段階で行うことをお勧めします。

ディスカッションに参加すべき利害関係者を特定し、推進要因を文書化します。通常 CxOs、組織内の上級マネージャー、主要なテクノロジーリーダー、および自分の顧客。顧客がこの議論に参加する可能性は低いですが、組織内の 1 人以上の人物が顧客の見解と目標を表すように指定することをお勧めします。

ビジネスドライバーは、移行ジャーニー全体で測定できるメトリクスにリンクして、成果が達成されたかどうかを検証する必要があります。会社の戦略的目標と年次レポートは、出発点となる可能性があります。

クラウドへの移行の結果として、既存のメトリクスと予測されたメトリクスに基づいて、会社がどこになりたいかに焦点を当てます。目標とビジネス成果を考慮します。また、クラウドの導入が増えるにつれて成功がどのようになるかを検討してください。

次に、各ドライバーの重要度レベルを設定します。優先順位は何ですか？期待される利点は何ですか？メリットはビジネス目標と成果をどのようにサポートしますか？アプリケーションポートフォリオ評価の文脈では、回答は移行するワークロードの優先順位付けと技術的な指針の確立に役立ちます。ただし、ビジネスドライバーは移行プログラム全体を定義し、影響します。

テクニカルガイドの原則

テクニカルガイドの原則は、ポートフォリオ評価の後期段階で移行戦略の選択に役立ちます。現在の段階では、それらを特定することに重点が置かれています。

指針は、ビジネス目標と成果から導き出される一般的なテクノロジー関連およびアプローチ関連の意思決定として確立できます。

例えば、ある企業にはコスト削減という主な目標があり、望ましい成果は、6～12か月以内に特定の日付までにオンプレミスデータセンターを閉鎖することです。その結果、可能な場合は常に、リホストまたは再配置移行戦略を使用して、すべてのアプリケーションをクラウドにリフトアンドシフトすることが指針となります。この場合、この lift-and-shift アプローチは短期的な移行成果を加速させます。アプリケーションがオンプレミスのデータセンターから移動した後、移行されたワークロードを最適化またはモダナイズするための主要なビジネスドライバーに集中できます。

テクニカルガイドの原則を確立するには、まずビジネス推進要因を分析します。ビジネス目標と成果を達成するテクノロジーとテクニックのリストを特定します。次に、リストを絞り込み、適合性または好みに基づいて関連性の順序を割り当てて、望ましい結果を実現します。

指針を文書化し、移行の計画と実行に関係する人々に伝えます。原則と実際の実装の間の懸念と潜在的な競合を強調します。

次の表は、ビジネス推進要因と技術指針の例を示しています。

ビジネスドライバー	結果	メトリクス	テクニカルガイドの原則
イノベーションを加速します。	競争力の向上、ビジネスの俊敏性の向上	1日または1か月あたりのデプロイ数、四半期ごとにリリースされ	マイクロサービスと DevOps 運用モデルを使用してアプリケーションの差別化をリ

ビジネスドライバー	結果	メトリクス	テクニカルガイドの原則
運用コストとインフラストラクチャコストを削減します。	需要と供給が一致する伸縮自在なコストベース (使用した分に対する支払い)	時間の経過に伴う支出の変動	<p>ファクタリングし、新機能の俊敏性と市場投入までのスピードを向上させます。</p> <ol style="list-style-type: none"> 1. インフラストラクチャの適切なサイズ設定でアプリケーションをリホストします。 2. 使用率が低い、または使用していないアプリケーションを廃止します。
運用の耐障害性を高めます。	稼働時間を改善し、平均復旧時間を短縮	SLAs、インシデント数	<ol style="list-style-type: none"> 1. アプリケーションを、最新でサポートされているオペレーティングシステムのバージョンにリプラットフォームします。 2. 重要なアプリケーションに高可用性アーキテクチャを実装します。
データセンターを終了します。	6~12 か月以内の日付によるデータセンターの閉鎖	サーバー移行の速度	Cloud Migration Factory ソリューションを使用してアプリケーションをリホストします。

ビジネスドライバー	結果	メトリクス	テクニカルガイドの原則
オンプレミスにとどまり、俊敏性と耐障害性を高めます。	オンプレミスを維持しながらの競争力と稼働時間を改善	1 日または 1 か月あたりのデプロイ数、四半期あたりの新機能リリース、SLAs、インシデント数	<ol style="list-style-type: none"> 機能をクラウドに拡張することで、システムを最新化します。 Outposts へのリストまたはリプラットフォームについて AWS 評価します。

データ収集の開始

データ収集は、アプリケーションとインフラストラクチャからメタデータを収集するプロセスです。このプロセスは、評価のすべての段階で反復されます。各ステージでは、データ量と忠実度が増加します。この段階では、初期インベントリの確立に役立つ一般的なデータの収集に焦点を当てています。インベントリは、方向性のあるビジネスケースと初期移行候補の特定を作成するために使用されます。

現在のデータソースを特定したら、できるだけ多くのシステムから情報を収集することをお勧めします。詳細については、このステージの[データ要件](#)を参照してください。

このアプローチには、現在のポートフォリオレビューと、アプリケーションとサービスに関する組織の知識を更新するのに役立つという利点があります。また、何を移動させるかを決定するのにも役立ちます。推奨されるアプローチは、設定管理データベース (CMDB) の出力や情報技術サービス管理 (ITSM) システムなどの既存のデータを確認することです。次に、データ収集の対象となるアセットのリストを作成します。移行の対象範囲と対象外範囲が組織で完全に明確であれば、データ収集を対象範囲内のシステムに制限できます。

ポートフォリオを構築するときは、アプリケーションとその環境またはソフトウェアリリースのライフサイクルを考慮してください。例えば、カスタマーリレーションシップ管理 (CRM) アプリケーションを識別し、テスト環境、開発環境、製品環境があることを指定する代わりに、3 つのアプリケーション (CRM-Test、CRM-Dev、CRM-Prod など) を一覧表示します。または、CRM 名を使用して各環境に一意の ID を割り当て、データリポジトリに個別のレコードとして表示します。これは、これらの環境の移行を個別に計画および追跡するのに役立ちます。例えば、本番環境以外の環境を最

初に移行したい場合があります。環境に応じてアプリケーションのインスタスを一覧表示することで、移行を明確に管理および管理できます。

データ収集中に、特定のデータセンターまたはソースロケーションにあるアプリケーションやサーバーが不明な場合があります。このような場合は、既存の管理ツールからベアメタルリストとハイパーバイザーリストを取得すると便利です。例えば、ハイパーバイザーに接続して、データ収集の対象となる仮想マシンのリストを取得できます。

既存のデータソースを組み合わせると、最初の出力が不完全になる可能性があることに注意してください。重要なのは、このステージの[データ要件](#)と、既存のソースから取得できる内容の観点からギャップ分析を実行することです。完全性の割合とデータ忠実度のレベルを対比することが重要です。低忠実度ソースからの完全性レベルが高いほど、分析の欠陥につながる可能性のあるいくつかの仮定が含まれます。この評価段階では最大データ忠実度は必要ありませんが、データソースは少なくとも中～中程度の忠実度にするをお勧めします。これらの数値を、データギャップを埋めるための仮定の使用を含め、組織のリスク許容度と照らし合わせます。

ギャップ分析は、作業しているデータの量と品質を理解するのに役立ちます。また、この分析は、方向性のあるビジネスケースを作成し、移行するアプリケーションを優先するために実行する必要がある仮定のレベルを確立するのに役立ちます。検出ツールは、ギャップを埋め、忠実度の高いデータを収集するのに役立ちます。データの信頼度を高め、移行成果を加速させるには、できるだけ早く検出ツールをデプロイすることをお勧めします。新しいツールの内部調達、セキュリティ、実装プロセスが完了するまでに数週間または数か月かかる可能性があるため、早期のアクションも重要です。

この段階では、コミュニケーションプランまたはケイデンスとスコープ変更制御メカニズムを確立することをお勧めします。これにより、ステークホルダーが事前に計画を立て、リスクを軽減できるように、ステークホルダーに情報を伝えることができます。明確なコミュニケーションの重要な要素は、アプリケーションポートフォリオと関連インフラストラクチャの信頼できる単一のソースを定義することです。複数のシステムを記録し、アプリケーションとインフラストラクチャのリストを保持することは避けてください。バージョニングとオンラインコラボレーションをサポートするデータを1か所(データベース、ツール、スプレッドシートなど)に保持し、所有者を割り当てます。

優先順位付けと移行戦略

移行計画の主な要素は、優先順位基準を設定することです。この演習のポイントは、アプリケーションを移行する順序を理解することです。戦略は、反復的かつ段階的なアプローチを取り、優先順位付けモデルを進化させることです。

アプリケーションの優先順位付け

この評価の段階では、リスクと複雑さの低いワークロードを優先するための初期基準の確立に焦点を当てます。これらのワークロードはパイロットアプリケーションに適しています。初期移行で低リスクで低複雑さのワークロードを使用すると、リスクが軽減され、チームは経験を積むことができます。これらの基準は、移行ウェブプランの作成時にビジネスドライバーとの優先順位付けを調整するために、さらなる評価段階で進化します。

初期基準では、クラウドでサポートされているインフラストラクチャで実行され、本番環境以外の環境で実行されている、少数の依存関係を持つアプリケーションを優先する必要があります。例としては、開発環境またはテスト環境で 0~3 個の依存関係をそのままリホストできるアプリケーションがあります。これらの基準は、クラウド導入の成熟度と信頼度のレベルに応じて、パイロットアプリケーションと、場合によっては 1 番目と 2 番目の移行ウェブを定義するのに有効です。

使用する初期基準の決定

最初のワークロードの優先順位付けに使用するデータポイントを 2~10 個選択します。これらのデータポイントは、最初のアプリケーションとインフラストラクチャのインベントリから取得されま
す ([データ収集](#) セクションを参照)。

次に、各データポイントの可能な値ごとにスコアまたは重みを定義します。例えば、環境属性が選択され、可能な値が本番、開発、テストの場合、各値にはスコアが割り当てられ、数値が大きいほど優先度が高いことを示します。これはオプションですが、各データポイントに重要度または関連性の乗算係数を割り当てることをお勧めします。このオプションのステップでは、より重要なことを強調する上位レベルの差別化要因を提供します。これにより、値にスコアを割り当てる際に繰り返し基準を一致させることができます。

次の表は、最初の数回の移行ウェブで低リスクのシンプルなアプリケーションを優先する戦略に基づいて、属性の選択例とその値の割り当てを示しています。

属性 (データポイント)	使用できる値	スコア (0~99)	重要度または関連性の乗算係数
環境	テスト	60	高 (1x)
	開発	40	
	本番稼働	20	
ビジネスの重要性	低	60	高 (1x)

属性 (データポイント)	使用できる値	スコア (0 ~ 99)	重要度または関連性の乗算係数
	中程度	40	
	高い	20	
規制またはコンプライアンスフレームワーク	なし	60	高 (1x)
	FedRAMP	10	
オペレーティングシステムのサポート	クラウド対応	60	中高 (0.8x)
	クラウドではサポートされていません	10	
コンピューティングインスタンスの数	1 ~ 3	60	中高 (0.8x)
	4-10	40	
	11 以上	20	
移行戦略	リホスト	70	中 (0.6x)
	リプラットフォーム	30	
	リファクタリングまたはリアーキテクト	10	

アプリケーション間の主要な差別化要因として機能する属性を必ず選択してください。そうしないと、条件によって多くのワークロードが同じ優先度を共有します。モデルを適用したら、結果のランキングの上部と下部を見て、同意するかどうか確認することをお勧めします。一般的に同意しない場合は、ワークロードのスコアリングに使用した基準を再確認できます。

ランキングを取得したら、ポートフォリオ全体のスコアの分布を確認します。スコア自体は関係ありません。重要なのはスコアの差です。例えば、上位の合計スコアが 8,000 で、下位スコアが 800 であることがわかります。結果のスコアをヒストグラムとしてプロットして、分布が良好であることを確認することを検討してください。理想的なディストリビューションは、非常に優先度の高いワークロードがいくつかあり、非常に優先度の低いワークロードがいくつかあり、標準のベル曲線のようになります。アプリケーションの大部分は中間のどこかにあります。

初期の優先順位付けのもう 1 つの重要な側面は、クラウドを早期に導入することに関心を示す内部チームまたはビジネスユニットを含めることです。これらは、特に初期の段階で特定のアプリケーションを移行するためのビジネスサポートを取得するためのかなりの手段になる可能性があります。組織でその場合は、前の表にビジネスユニット属性を含めます。アプリケーションを進んで進んでいるビジネスユニットに高いスコアを割り当てます。ビジネスユニット属性を使用すると、これらのアプリケーションをリストの最上位に配置できます。

結果のランキングに同意したら、上位 5~10 のアプリケーションを選択します。これらは、アプリケーション移行の初期候補になります。3~5 個のアプリケーションを確認するようにリストを絞り込みます。これにより、詳細なアプリケーション評価を実行する際に、的を絞ったアプローチを取ることができます。詳細については、[「優先順位付けされたアプリケーションの評価」](#)を参照してください。

移行用の R タイプの決定

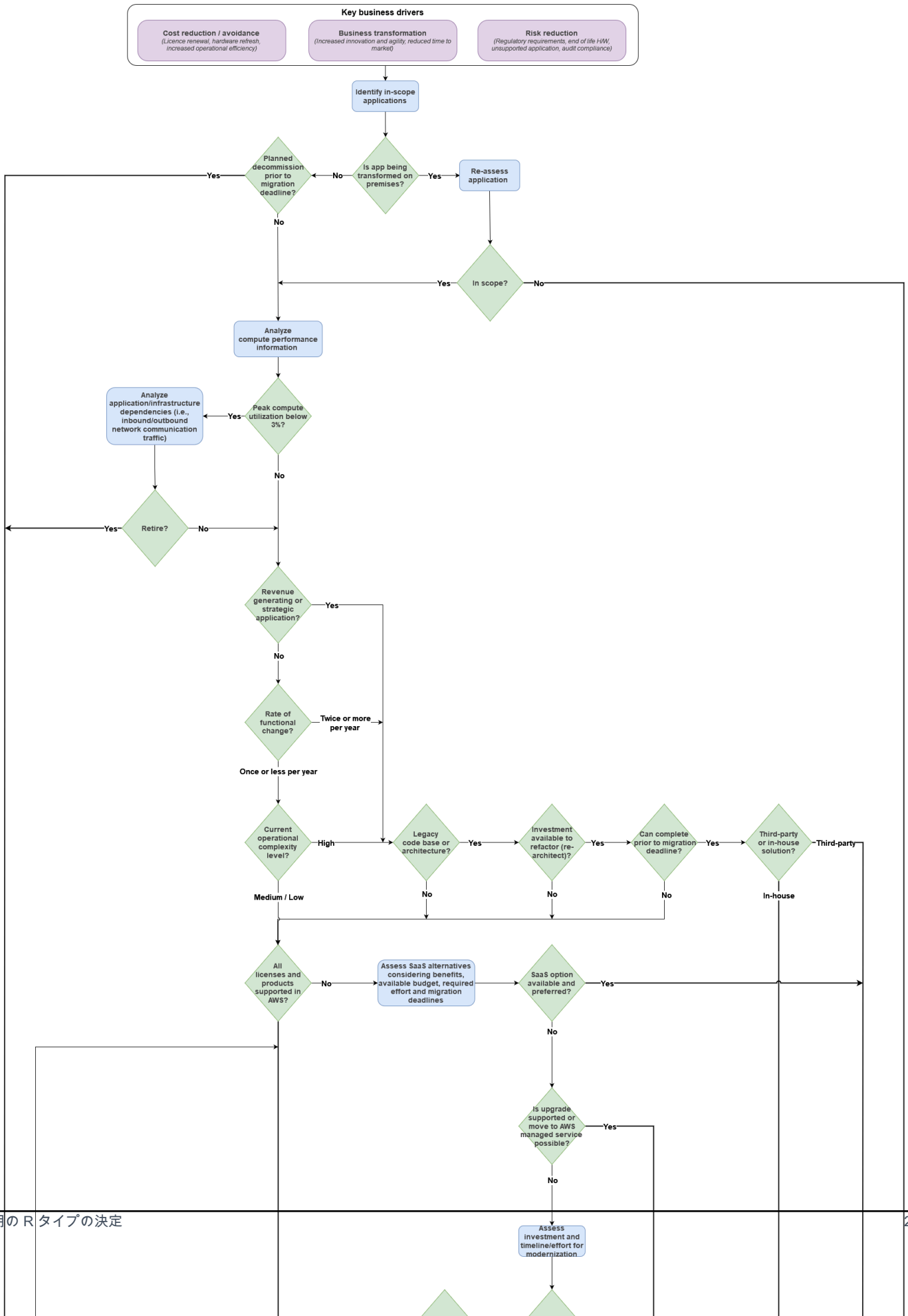
各アプリケーションおよび関連するインフラストラクチャの移行戦略を決定すると、移行速度、コスト、および利点のレベルに影響します。ビジネスドライバー、技術指針、優先順位付け基準、ビジネス戦略など、バランスの取れた要素の組み合わせに基づいて戦略を決定することが重要です。

これらの要因が競合するビューを作成することがあります。例えば、移行の主な推進要因はイノベーションと俊敏性かもしれません。同時に、コストを迅速に削減する必要がある場合があります。範囲内のすべてのアプリケーションをモダナイズすると、長期的にはコストを削減できますが、事前により大きな投資が必要になります。この場合、1 つのアプローチは、リホストやリプラットフォームなど、労力の少ない戦略を使用してアプリケーションを移行することです。これにより、短期的に迅速に効率とコスト削減を実現できます。その後、後ほどアプリケーションのモダナイズにコスト削減を再投資し、さらなるコスト削減を実現します。

ただし、すべてのアプリケーションの完全なリホストから始めると、モダナイゼーションのより大きな利点が遅れます。重要なのは、ビジネス戦略のアプリケーションがモダナイゼーションに優先され、他のアプリケーションを最初にリホストまたはリプラットフォームしてからモダナイズできるように、移行戦略のバランスを見つけることです。

アプリケーションの移行戦略を決定する方法

この評価段階では、移行戦略の選択をガイドするための初期モデルを組み込むことに重点が置かれています。初期アプリケーションの移行戦略を検証するには、モデルをビジネスドライバーと優先順位付け基準と組み合わせて使用します。決定木のデフォルトのロジックは、スコープの初期処理を決定するのに役立ちます。ツリーでは、リファクタリングやリアーキテクトなどの最も複雑なアプローチは、戦略的ワークロード用に予約されています。



この図のカスタマイズ可能な [draw.io](#) バージョンは、[添付ファイル](#) セクションで入手できます。

初期モデルの最初のステップは、ツリーの上にあるビジネスドライバーを、組織で定義されたビジネスドライバーで更新することです。次に、アプリケーション全体ではなくアプリケーションコンポーネントにツリーを適用します。例えば、3つのコンポーネント（フロントエンド、アプリケーションレイヤー、データベース）を持つ3層アプリケーションの場合、各コンポーネントはツリーを個別に転送し、特定の戦略とパターンを割り当てる必要があります。これは、場合によっては、特定の階層をリホストまたはリプラットフォームし、他の階層をリファクタリング（リアーキテクト）する必要があるためです。

独立したコンポーネントを割り当てることで、関連するインフラストラクチャの移行戦略を定義できます。インフラストラクチャ戦略は、サポートするアプリケーションコンポーネントと同じ戦略でも、異なる戦略でもかまいません。例えば、新しいオペレーティングシステムで新しい仮想マシンにリプラットフォームされるアプリケーションコンポーネントは、リプラットフォーム戦略に従い、それをホストする現在の仮想マシンは廃止されます。インフラストラクチャの移行戦略は、アプリケーションコンポーネントに対して選択された戦略に基づいて計算されます。

決定木を使用して移行戦略を確立する前に、いくつかのアプリケーションでロジックをテストし、一般的に結果に同意するかどうかを確認します。6 Rs 決定木は、その正確性を判断するために必要な分析を置き換えないガイドです。ツリーロジックは、特定のケースに適用されない場合があります。これらのケースを例外として扱い、ツリーロジックを変更するのではなく、オーバーライドの理論的根拠を文書化して、ツリーによって駆動される決定をオーバーライドします。これにより、複数の決定木のバージョンが防止され、管理が困難になる可能性があります。一般的なガイダンスでは、ツリーはワークロードの少なくとも70~80%に対して有効である必要があります。残りには例外があります。ツリーロジックの調整は、評価のこの段階では、初期モデルの確立に焦点を当てる必要があります。さらなる反復と改良は、[ポートフォリオ分析](#)や[移行計画](#)など、後の段階で行われます。

添付ファイル

[attachment.zip](#)

方向性のあるビジネスケースの作成

ビジネス全体のステークホルダーは、その過程の各ステップで変革のためにビジネスケースを理解し、賛同する必要があります。

初期段階では、プログラムの計画と確立に必要なリソースを確保できるように、移行プログラムから十分な潜在的な価値をすばやく示すことが重要です。方向性のあるビジネスケースは、早期に収集で

きる限られたデータで、説得力のあるビジネス価値を実現するための合理的な信頼を提供するように設計されています。

プログラムが確立されると、ビジネスケースがさらに開発されます。詳細なケースは、より高い精度、プログラムの価値をより完全に把握し、計画の優先順位に関する洞察を提供します。組織が賛同する計画されたビジネス成果を定義して定量化し、プログラムガバナンスオフィスがプログラムを運営し、その成果を測定するためのベースラインを設定します。

方向性のあるビジネスケースの範囲の修正

方向性のあるビジネスケースは通常、2~4 週間以内に迅速に組み立てられます。コアチームを確立し、必要に応じて AWS パートナーを関与させ、少なくとも[優先順位付けされたアプリケーション評価](#)、[ポートフォリオ分析](#)、[移行計画](#)の各段階を完了できるように、十分な信頼を生み出す必要があります。

通常、ポートフォリオの移行をサポートする方向性のあるビジネスケースは、次のいずれかとして作成されます。

- 現状のインフラストラクチャ環境と移行後の AWS サービスアーキテクチャの単純な総所有コスト (TCO) の比較。比較は、特定のワークロードボリュームの予想実行率の差を示しています。
- AWS 移行コストを含む への移行とそのままの状態での存続について、純現在価値 (NPV)、投資収益率 (ROI)、ペイバック期間、内部収益率 (MIRR) の変更、3~5 年分のキャッシュフロー分析を示すビジネスケース。

方向性のあるビジネスケースの範囲は通常、次のいずれかに制限されます。

- インフラストラクチャテクノロジーコストの比較
- インフラストラクチャテクノロジーと運用コストの比較

一般に、ポートフォリオが大きいほど、ケースの開発は少なくなります。これは、結果に大きな影響を与えることなく、より広範な仮定を実行できるためです。ポートフォリオを小さくすると、変更の影響が大きくなるため、より詳細にする必要があります。

まず、基本インフラストラクチャのコスト比較を構築します。次に、続行する前に、比較が十分に説得力があるかどうかを判断します。通常、400 台を超えるサーバーのポートフォリオは、で運用されてから 3 年以内 AWS、または 5 年以内のサーバー 250 台のインフラストラクチャのコスト削減だけでもプラスのビジネスケースになりますが、これは異なる場合があります。より小さなポートフォリオの場合は、より詳細な情報が必要になる場合があります。

逆に、移行範囲の合計が約 5 ワークロードまたは 50 サーバー未満でない限り、回復力の向上やビジネスの俊敏性から得られる価値など、この段階で他のビジネス価値コンポーネントを調べることはめったに役に立ちません。

フォーカス値ドライバー

インフラストラクチャテクノロジーの TCO 比較では、現状のインフラストラクチャコストのモデルと、同等のパフォーマンスと可用性でワークロードを実行するために必要な AWS サービス部品表の基本モデルを比較します。多くの最適化を行うことができます。ただし、この段階では、評価が容易で、通常は TCO を約 30% 削減できるため、次のリストに重点が置かれています。

- コンピューティングの伸縮性 — 8x5 (24% の使用量)、10x5 (30%)、または 10x6 (36%) を実行している開発サーバーや UAT サーバー、2% で実行されているディザスタリカバリ (DR) サーバーなど、使用量が 100% ではないサーバーを、使用時にのみ課金されるオンデマンドサービスにマッピングします。
- 削減計画による調達 – 本番稼働用サーバーやその他のサーバーを使用量の多い (36% 超) で調達し、コストを最大 75% 削減するための適切な削減計画を計画します。オプションには、1 年契約と 3 年契約があり、割引率を高めるために前払いレベルが異なります。
- ゾンビの削除 — CPU 使用率が 2% 未満のサーバーのうち、不要になったことを確認できるサーバーを特定し、コスト分析から削除します。
- コンピューティングの適正サイズ — CPU とメモリの使用率の時系列データを使用して、サーバーごとに必要なコンピューティング能力とメモリを評価します。次に、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを選択します。
- リレーショナルデータベース管理システム (RDBMS) ライセンスの適正サイズ設定 – データベースサーバーでコンピューティングの適正サイズ化後に RDBMS ライセンスのニーズを再評価し、からライセンス持ち込み (BYOL) とライセンス調達を比較し AWS、Amazon Relational Database Service (Amazon RDS) の可能性を調べてコスト削減を図ります。
- ストレージ – 必要なストレージボリュームの合計を適切なサイズに設定し、ポートフォリオ全体の 1 秒あたりの入出力オペレーション (IOPS) のニーズを特定します。SLA SLAs とコストが異なるオブジェクトストレージに移動できる量を決定します。

データのニーズ

[「初期評価データ要件を理解する」](#)の表は、方向性のあるビジネスケースの各部分を構築するために必要なデータと、それが必須かオプションかを示しています。

ケースを構築するには、初期計画データとコストデータのインフラストラクチャサブセットが必要です。含めるインフラストラクチャを特定する方法は、ビジネス目標によって異なります。

- プログラムの目的は、特定のアプリケーションを移行してモダナイズすることである場合は、共有されるインフラストラクチャを考慮して、アプリケーションが必要とするものに基づいてインフラストラクチャポートフォリオを構築します。
- リースの有効期限が切れるデータセンターから移行するなど、プログラムの目的がインフラストラクチャ中心である場合、インフラストラクチャの TCO 比較にアプリケーションマッピングは必要ありません。

オプションとしてマークされたデータ (サーバーの CPU やメモリのピーク使用率など) は、通常、標準ベンチマーク値に置き換えることができます。これについては、AWS パートナーまたは AWS プロフェッショナルサービスにお問い合わせください。または、ポートフォリオの一部で利用可能なデータポイント (ハイパーバイザーによって収集されたデータなど) から値を推定することもできます。ポートフォリオが大きいほど、より正確です。

インフラストラクチャの TCO 比較の構築

ツールはインフラストラクチャの TCO 比較を構築する上で不可欠です。[AWS プロフェッショナルサービス](#)または [AWS パートナー](#)は、あらゆるタイプの方向性のあるケース、特により広範な移行プロセスを支援するために関与させる予定がある場合は、サポートを提供できます。

以下の操作に使用できるツールがあります。

- インベントリデータを収集します。
- 使用率データを収集します。
- インフラストラクチャのコストベンチマークデータをそのまま正確に提供します。
- ゾンビを特定して削除します。
- 適切なサイズ評価を行います。
- 購入オプションを推奨します。
- ソフトウェアライセンスオプションを比較します。
- シンプルなグラフィカルなキャッシュフロー分析を作成します。

からの[移行工バリエーター](#) AWS は 1 つのオプションです。これらすべての機能を無料のマネージドサービスとして提供します。Migration Evaluator は、AWS アカウントマネージャーまたは AWS Migration Competency Partner を通じて、または[オンラインでリクエスト](#)を送信することでリクエスト

トできます。Migration Evaluator は、インフラストラクチャテクノロジーの TCO 比較を迅速に作成するためのポイントソリューションとして特別に設計されています。

主な利点：

- 無料
- ツールベースの検出が制限されているインベントリデータのエージェントレス検出または手動設定
- デプロイ、設定、データ収集、ベースケースの構築、または方向性のあるビジネスケースを支援する専用サポート
- SaaS オペレーションの利便性がありますが、分析エンジンにロードする前にスクラブをサポートするために、顧客ネットワーク内でデータ収集を完全に実行できます。
- Microsoft ライセンスの適切なサイズ設定に対する強力なサポート
- 完全なデータエクスポート機能

主な制限事項：

- x86 アーキテクチャサーバー (Windows および Linux) のみを評価
- ベンチマークをそのままコストデータとして設定またはキャリブレートするためのオプションが限られている
- モデリングオペレーションのコスト最適化はサポートされていません
- 移行コストモデリングはサポートされていません
- TCO 比較以外のビジネスケースの構築を直接サポートしない

アプリケーションスタックや相互依存関係検出などのポートフォリオ検出および分析機能に商用検出ツールを使用することにした場合、通常、インフラストラクチャの TCO 比較も提供されます。ポートフォリオの検出と評価のためのツールの使用に関するガイダンスについては、[「検出ツールの必要性の評価」](#)を参照してください。

運用コスト最適化の構築

IT 運用の生産性の向上は、多くの場合、移行にとって大きな価値をもたらします。アマゾン ウェブ サービスでビジネス価値を生み出すためのビジネスと組織の変革を促進するホワイトペーパーによると AWS、への移行後、IT 運用スタッフの生産性は平均 62% 向上します。<https://pages.awscloud.com/rs/112-TZM-766/images/AWS-BV%20IDC%202018.pdf?aliid=1614258770>ただし、サイズ設定と方向性のあるケースにこれらの利点を含めるには 2 つの課題があります。

まず、生産性の向上の全範囲を評価するには、広範なデータ収集が必要であり、[詳細なビジネスケース](#)により適しています。この課題は、単純なベンチマークデータでより簡単に評価およびサイズ設定されるが、それでも大きな利点を示すいくつかの要素に焦点を当てることで解決できます。

2つ目は、コスト削減のソースとして生産性に焦点を当てることです。これにより、主要な顧客の利害関係者やプログラムメンバーの間で懸念や否定が生じる可能性があります。メリットがどのように実現され、それが影響を受ける人々にとってどのような意味を持つかを明確にしてください。このような問題は、それがチームの役割を強化するだけであることを明確にすることで回避できます。

- 移行プログラムには、コード自動化としてインフラストラクチャを構築する DevSecOps チームへの参加や、チームの成長を促進する自動化をテストするなど、内部運用スタッフを新しい役割に開発して移行するためのトラックが含まれています。
- この利点は、オペレーションの委任契約の範囲を変更およびサイズ変更することで実現できます。これにより、社内スタッフが価値の高いアクティビティに集中できるようになります。

考慮すべきオペレーショントランスフォーメーションに基づいて、このビジネスケース要素を構築するアプローチ：

- 既存の社内運用チームがある場合は、チームメンバーのスキルを向上させ、期待される生産性の向上を示します。
- または、現在の運用ソリューション AWS から マネージドサービス (AMS) または AWS パートナーが提供する代替マネージドサービスに移行します。

最初のトランスフォーメーションでは、ケースに含めることができる生産性の向上について控えめに財務上の見積もりを得るには、以下をお勧めします。

1. 特にサーバー管理オペレーションの生産性に焦点を当てます。オペレーションの労力の大部分を占める傾向があり、より簡単に評価でき、後でより簡単に検証できます。
2. 各正社員 (FTE) 従業員が管理できるサーバー数のベンチマークに基づいて、必要な人員を計算します。オンプレミスでは、その数は約 150 サーバーです。では AWS、約 400 台のサーバーです。
3. これらのメトリクスを EC2 インスタンスの数と比較したオンプレミスサーバーの数に適用します。
4. 節約した時間を運用チーム全体のブレンドコスト率で乗算します。

その後、結果をいずれかの方法で確認するには、結果を次の表に示すロールごとの平均生産性の向上 (IDC ホワイトペーパー「[Fostering Business and Organizational Transformation to Generate Business Value with Amazon Web Services](#)」から取得したデータ。

ロール	効率の向上
IT インフラストラクチャ管理	62%
IT サポート	59%
アプリケーション管理	43%
データベース管理	19%
アプリケーション開発	25%

2 つ目のトランスフォーメーションでは、対象範囲内のポートフォリオの現在の総運用コストとサポートコストを、検討中のマネージドサービスのコストと直接比較することで、運用コストの削減を追加できます。

マネージドサービスのコストを取得するには、AWS アカウントマネージャーまたは任意の[AWS Managed Services パートナー](#)に、提案された部品 AWS 表、サービスレベルの選択 (Plus または Premium)、および AMS パッケージ (AMS Accelerate または AMS Advanced) を提供します。これにより、変換されたソリューションのサービスコンポーネントのマネージド AWS サービスの合計コストが提供されます。同様に、独自のパラメータに基づいて独自のマネージドサービスパッケージを提供する AWS パートナーから料金を取得できます。

完全な方向性のあるビジネスケースへの拡張

一般的に、方向性の高い完全なビジネスケースを構築するには、IT productivity 要素の有無にかかわらず TCO 比較を構築し、移行とモダナイゼーションのすべてのコストを見積もります。次に、migrate-and-modernize と don't-migrate-and-modernize scenarios のペアをカバーするキャッシュフローを作成します。

最も基本的なケースは、シナリオの 1 つのペアを準備することです。シナリオでは、t-migrate-and-modernize シナリオのシナリオが現在の状況であり、migrate-and-modernize シナリオには次の特性があります。

- トランザクションボリューム、コンピューティング、ネットワーク容量の増加や縮小がない

- ストレージ要件の少量増加が着実に増加
- 既存のシステムのquality-of-service 機能と一致する Q 機能 (可用性、耐久性、スループット、パフォーマンスなど)

これは、非常に小さなポートフォリオを除くすべてのポートフォリオで、方向性のあるケースをうまく構築するという目的に合致します。これは、先に進むための義務を得るために十分な価値を迅速に示します。

ポートフォリオが小さい場合は、クラウド移行の価値の高まりの migrate-and-modernize 側の側面 migrate-and-modernize を示す のペアを追加することは有益です。

- 中程度と大容量の増加要件が、その増加が予想されるワークロード間で混在する
- 高可用性、DR、耐障害性などの強化された耐障害性を含める
- エッジコンピューティング、コンテンツ配信ネットワーク (CDN)、マルチリージョンデータベースレプリケーションでグローバルパフォーマンスが向上しました。
- プログラムのビジネス上の優先事項となったその他の特定の改善されたサービス品質

これらのシナリオでは、現在の非クラウドインフラストラクチャアーキテクチャを新しい仕様に合わせてアップグレードした場合のコストとキャッシュフローへの影響が正確に推定されていることを確認してください。この見積りを取得する最も直接的な方法は、特に、 migrate-and-modernize と don't-migrate-and-modernize scenarios の両方でサポートできる Migration Competency の AWS コンサルティングパートナーでもある場合、システムインテグレーターに見積りをリクエストすることです。

シナリオのペアごとに、以下で構成されるケースを組み立てます。

- シナリオのコスト migrate-and-modernize 。最も基本的なケースでは、これには以下が含まれます。
 - 現在のインフラストラクチャ設定におけるビジネスケース期間の総所有コスト
 - コンピューティング、ストレージ、ネットワークトラフィックの消費量の定期的な増加
- シナリオのコスト migrate-and-modernize。これには以下が含まれます。
 - 詳細な発見、移行計画、詳細なビジネスケース開発、コアチームの確立とスキルの向上、まだ実施されていない場合のランディングゾーンの確立、移行されたワークロードのセキュリティ管理と運用の統合の確立など、プログラムのセットアップ
 - ワークロードの移行とモダナイゼーションのコスト

- ネットワーク接続、[AWS Snowball](#) やなどのデータ移行サービス[AWS DataSync](#)、移行プロセス自体に必要なアーキテクチャの AWS ユーティリティコスト (テストなど) などの移行インフラストラクチャのコスト
- ウェーブが稼働するにつれて移行中の AWS ユーティリティコストが増加し、既存のインフラストラクチャコストが AWS ベースのサービスに置き換えられて廃止されるにつれて増加します。

使用停止したアセットの廃止コストと償却

移行とモダナイゼーションプログラムのセットアップの見積もり

プログラムを正常に設定するには、ベースライン機能と詳細な計画を構築するための一連の基本的なアクティビティを以前に実行する必要がある場合があります。これらの基本的なアクティビティには以下が含まれます。

1. ポートフォリオ分析と移行計画セクションで説明されているように、詳細な[ポートフォリオ検出](#)、[移行計画](#)、詳細なビジネスケース開発を実行するとともに、使用されている検出ツールのコストを文書化します。
2. クラウドビジネスおよびテクニカルコアチームを確立し、トレーニングと雇用を通じて社内スキルを開発します。トレーニングが必要な IT 組織のメンバーを特定し、各メンバーにトレーニング予算を割り当てます。
3. [ランディングゾーン](#)を確立し、必要なコスト、運用、セキュリティガバナンス機能をサポートするように設定します。

AWS コンサルティングパートナーは、項目 1 と 3 の見積りを提供できます。

移行とモダナイゼーションのコストの見積もり

方向性のあるビジネスケースの目標を達成し、次のフェーズに進むのに十分な商用の可能性を実証するには、移行とモダナイゼーションのコスト見積もりをできるだけ基本的なものにします。

そのためには、次の移行戦略に該当するアプリケーションに焦点を当てて、方向性のあるビジネスケースを準備することをお勧めします。

- リタイア
- Retain
- リロケート
- リホスト

- リプラットフォーム
- 再購入

通常、ワークロードの約 70% はリホスト、再配置、またはリプラットフォームでき、さらに 5% はリタイアできます。移行戦略によるアプリケーションの評価は、通常、コスト削減ケースの中核をなします。

リファクタリングまたは再構築のコストの見積もりは複雑になる場合があります。方向性のあるビジネスケースを準備するために与えられた期間内にこれを試すことは実用的ではありません。前述の「[移行の R タイプの決定](#)」で説明したように、移行とモダナイゼーションの最初のフェーズでは、リホスト、再配置、またはリプラットフォーム戦略の使用を検討してください。これらの R 戦略は、初期支払いを迅速化し、実装リスクを軽減し、短期的にビジネスケースを改善する可能性があります。また、アプリケーションチームは、AWS 環境内で実行されているアプリケーションを、そうでないアプリケーションよりもモダナイズすることが大幅に簡単になります。[詳細なビジネスケース](#)の準備が整うと、リファクタリング (再構築) 固有のアプリケーションの見積もりが最適に追加されます。

戦略による移行の労力の見積もり

移行はそれぞれ異なります。予算や計画をコミットする前に、社内のアプリケーションチーム、AWS プロフェッショナルサービス、または AWS パートナー組織など、プロジェクトを担当するチームからの移行アクティビティのワークロードの見積もりを作成します。

方向性のあるケースを構築するために、次の表にさまざまな処理の労力範囲を示します。これらの範囲は、medium-to-largeポートフォリオが移行中であり、移行チームがトレーニングされ、経験があることを前提としています。小規模なポートフォリオの場合は、方向性のあるケースであっても、移行を担当するチームに見積りを準備することをお勧めします。

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
Retain	何もせず、コストも利点もテクノロジーの負債も削減しません。	-	-	-
リタイア	使用されているハードウェア	-	-	-

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
	機器がある場合は、廃止を見積もります。			
リロケート	VMware VMware ツールを使用して、VMware 内のワークロードのコピーを推定します。これには、データのコピー、検証のための煙のテスト、ハードウェアの廃止が含まれます。VMs は、通常、複雑さの低いリホストパターンの場合よりも小さくなります。	-	-	-

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
リホスト	本番サーバーに適した場合は、イメージコピー、煙テスト、高可用性 (HA) およびディザスタリカバリ (DR) テスト、およびハードウェアの廃止を使用して、ワークロードとデータのコピーを推定します。ベストプラクティスは、 AWS Application Migration Service などのツールを使用することです。データベースや他のインフラストラクチャソフトウェアが実行されているかどうか、データベースの複雑さ、クラスター化されているかどうか、統合の複雑さ、データボリュームなどの要因に基づいて、ワークロー	サーバーあたりのアプリケーションあたりの労力	移行	HA/DR テスト
		低	10 ~ 14	3 ~ 5
		中程度	16 ~ 24	4 ~ 6
		高い	26 ~ 38	8 ~ 12

移行戦略	見積りプロセス ドを低、中、高 の複雑さに分割 します。	[Elements] (要素)	人時	人時
------	---------------------------------------	-----------------	----	----

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
リプラットフォーム	オペレーティングシステムまたは RDBMS バージョンへのアップグレードを含むリプラットフォーム移行の場合、リホストの見積もりを取得し、新しいプラットフォームで再構築と煙のテストを実行する時間を追加します。リプラットフォームにプラットフォームのテクノロジーの変更が含まれている場合は、 AWS Schema Conversion Tool やなどの変換ツールの使用にかかる追加の時間を見積もり AWS Database Migration Service 、より完全なアプリケーションテストを行います。テクノロジーを	サーバーあたりのアプリケーションあたりの労力	バージョンアップ	テクノロジーの変更
		低	1~3 を追加する	10~15 を追加
		中程度	2~5 を追加する	20~30 を追加
		高い	4~8 を追加する	40~60 を追加する

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
	更なる例としては、独自の商用データベースからオープンソースの置き換えへの移行があります。			
再購入	新しく購入した SaaS サービスの交換、およびハードウェアの廃止へのデータ抽出、変換、アップロードを推定します。	-	-	-

移行インフラストラクチャコストの見積もり

移行中に使用するインフラストラクチャの見積もりを含めます。通常、これらの見積もりは以下で構成されます。

- 現在の環境からへのワークロードとデータ移行のための接続とデータ交換サービスの予算 AWS
- 移行、テスト、カットオーバープロセス中に移行されたワークロードをホストするために必要な AWS サービス (特にコンピューティングとストレージ) の予算
- 各移行ウェーブの完了に伴う AWS 光熱費の増加
- 移行されたワークロードを実行しない既存のインフラストラクチャの廃止コスト

データ交換については、合計データボリュームを調べ、ネットワークの使用の実現可能性を評価します。移行後に運用上の使用のために [AWS Direct Connect](#) リンクまたは [AWS VPN](#) から WAN 上のポイント AWS に事前にプロビジョニングしている場合は、そのリソースをサービスクォータまで使用できます。

ネットワーク容量が不十分な場合、仮想プライベートネットワーク (VPN) によるインターネット帯域幅の短期的な増加は、多くの場合、非常にコスト効率の高いソリューションです。そうでない

場合、[AWS Snowball](#) や [AWS Snowcone](#) などの AWS メディア交換デバイスは、ほとんどの AWS リージョンでソリューションを提供します。また、非常に大量のデータ移行の場合は、の予算を含めることを検討してください。これにより [AWS DataSync](#)、使用するメディアに関係なく信頼性が向上し、転送が高速化されます。

AWS サービスの拡大と既存のインフラストラクチャの縮小をモデル化することは、ビジネスケースのキャッシュフロー分析要素にとって重要です。この段階では、コストが発生するタイミングを正確に判断するためのウェーブプランは用意されていない可能性があります。次の構成を推奨します。

- のコストを移行よりも一定の割合 AWS で引き上げます。
- 同じ期間にわたって一定の割合で廃止する予定の既存のインフラストラクチャのコストを引き下げます。

既存のインフラストラクチャが縮小する 1~2 か月前に、AWS コストが増加します。これにより、ウェーブごとに移行を実行するために 1 か月間の AWS ユーティリティ使用量が提供されます。これには、テストにかかる時間と、置き換えられたインフラストラクチャでコストが発生しないようにするために必要な廃止作業を完了する追加時間が含まれます。

廃止コストの見積もり

再デプロイできない機器は廃止し、法的かつ環境に優しく廃棄すると、少額のコストが発生する可能性があります。ただし、方向性のあるビジネスケースでは、通常、潜在的に重要な合計は、置き換えられたアセットの残りの書籍価値を償却するコストだけです。

方向性のあるビジネスケースでは、以下を実行することをお勧めします。

- アセットリストを確認します。
- 廃止されるものを特定します。
- 書き込みオフを減らすには、リスト内の新しいデバイスを使用して、より完全に廃止された古いアセットを置き換えることができるように、デバイスを切り替えられる機会を調べます。
- その時点で廃止されるアセットの将来の書籍価値を評価します。
- これを廃止の移行コストとして含めます。

方向性の高い完全なビジネスケースの組み立てと調整

シナリオのペアごとにコストの完全なセットを準備したら、それぞれについて割引されたキャッシュフローステートメントを作成し、グラフ化します。ハードウェアの更新サイクルと同じ期間に方向性

のあるビジネスケースを構築することをお勧めします。サーバー、ストレージ、ネットワークデバイスの場合は、通常 5 年です。ハードウェアの更新サイクルと同じ期間を使用する場合、1 回のみ更新のコストは、シナリオごとにそのままのコストに含まれます。

次に、プログラムの次のフェーズに進むための承認を得るために必要な主要な財務メトリクスを計算します。通常、以下が含まれます。

- 評価されたコスト削減と生産性の向上の絶対値を測定するための正味現在価値 (NPV)
- リターンが十分に高速であることを確認するための月単位のペイバック期間
- プロセスが期間にわたって十分なコストを奪っているかどうかを確認する最後の実行レート比較
- 組織が優先する可能性のある資本に対する他の需要と比較して、プログラムの相対的な財務パフォーマンスを評価するための投資収益率 (ROI) と変更された投資収益率 (MIRR)

ケースの最初の反復を使用して、次の例のように、予想される財務パフォーマンスが微調整を行う必要があることを意味するかどうかを判断します。

- ペイバックが遅すぎる場合は、次のような移行を加速してコストを削減するためのオプションを検討してください。
 - AWS パートナーまたは AWS プロフェッショナルサービスを使用して、利用可能なリソースを拡張し、より基本的なパターンでワークロードの移行をさらに並列化します。
 - VMware で実行されているワークロードの場合、少なくとも初期フェーズでは、再配置戦略とリホストまたはリプラットフォーム戦略を比較します。再配置戦略を使用すると、移行コストを削減し、移行速度を向上させることができます。
 - 技術的に可能であれば、より複雑なリプラットフォームまたはリファクタリング (リアーキテク) 戦略を必要とするワークロードを、最初のビジネスケースの範囲外の将来のフェーズにプッシュします。
- ROI と MIRR が低すぎる場合は、次の点を考慮してください。
 - 検討しているシナリオは保守的すぎますか？ 最も可能性の高い容量の増加と伸縮性のニーズを反映したシナリオはありますか？ 目標内のサービス品質の向上を含むコストを比較するシナリオはありますか？
 - 第 1 フェーズで移行するアプリケーションポートフォリオの範囲を絞り込んで、現在の使用率が低いワークロードや、ディザスタリカバリ (DR) のニーズが高いワークロードなど、より強いリターンをもたらすワークロードに集中できますか？
 - アプリケーションポートフォリオの範囲を絞り込んで、当初はあまり商用ではない特定のワークロードを除外できますか？ 例えば、パブリッククラウドインフラストラクチャへのデプロイの

条件が異なるために、サードパーティーのソフトウェアライセンスがより高価になるワークロードを延期できますか？

- 最終的な実行レート比較が想定ターゲットを満たさない場合は、以下を調べます。
 - まず、他のメトリクスが期待を満たしていることを確認します。方向性のあるビジネスケースは、主に移行準備の次の段階を開始することを正当化する十分な財務機会があることを示すことです。
 - 移行の初期フェーズ AWS 後も のコストパフォーマンスを継続的に改善する機会のリストを特定します。

詳細なビジネスケースを準備する際には、機会のリストの評価を含めます。さらに、ケースの継続的なメンテナンスと移行完了後の month-to-month コスト最適化プロセスに機会評価を含めます。

優先順位付けされたアプリケーション評価

前のステージの主な成果の 1 つである [ポートフォリオの発見と初期計画](#) は、詳細な評価のために [アプリケーションのサブセットに優先順位](#) を付けることでした。このセクションでは、アプリケーションの詳細な評価について説明します。

いくつかのアプリケーションの詳細を早い段階で見ると、加速が促進されます。評価のプロセスと今後のアーキテクチャ設計は、潜在的な障害を明らかにし、大規模な移行を妨げている重要なタスクを明確にします。これらのタスクには、のランディングゾーンなどの AWS 基盤を確立したり AWS、既存のランディングゾーンを拡張して検証したりするための要件の収集が含まれます。この評価は、移行のステップと戦略を検討する時期でもあります。

このステージの主な成果は次のとおりです。

- 優先順位付けされたアプリケーションの検証済みリスト
- 文書化された現在の状態アーキテクチャ
- 移行候補に関する文書化された初期ターゲットアーキテクチャと移行戦略
- 特定された移行パターンとツール
- 文書化されたプラットフォーム要件 (セキュリティ、AWS インフラストラクチャ、運用)
- 移行計画に関する文書化されたカットオーバーに関する考慮事項
- 推定 AWS 実行レート

詳細な評価データ要件を理解する

次の表は、移行内のアプリケーションとそれに関連するインフラストラクチャの完全なポートフォリオビューを取得するために必要な情報を示しています。

テーブルでは、次の略語を使用します。

- 必須の R
- O、オプション
- 該当なし、該当なし

アプリケーション

属性名	説明	発見、設計、移行戦略	推定実行レート	推奨忠実度レベル (最小)
一意の識別子	例えば、アプリケーション ID などです。通常、既存の CMDBs またはその他の内部インベントリと管理システムで使用できます。一意の IDs 組織で定義されていない場合は必ず作成することを検討してください。	R	O	高い
アプリケーション名	このアプリケーションが組織で認識される名前。必要に応じて、商用 off-the-shelf (COTS) ベンダーと製品名を含めます。	R	R	高い
COTS ですか？	はいまたはいいえ。これは商用アプリケーションか内部開発か	R	R	高い
COTS 製品とバージョン	商用ソフトウェア製品名とバージョン	R	R	高い
説明	プライマリアプリケーションの	R	O	高い

属性名	説明	発見、設計、移行戦略	推定実行レート	推奨忠実度レベル (最小)
	関数とコンテキスト			
緊急性	例えば、戦略的アプリケーションや収益を生み出すアプリケーション、重要な機能のサポートなどです。	R	O	高い
タイプ	データベース、顧客関係管理 (CRM)、ウェブアプリケーション、マルチメディア、IT 共有サービスなど	R	O	高い
環境	例えば、本番稼働、本番稼働前、開発、テスト、サンドボックスなどです。	R	R	高い
コンプライアンスと規制	ワークロードに適用されるフレームワーク (HIPAA、Sox、PCI-DSS、ISO、SOC、FedRAMP など) と規制要件	R	O	高い

属性名	説明	発見、設計、移行戦略	推定実行レート	推奨忠実度レベル (最小)
依存関係	内部および外部のアプリケーションまたはサービスへのアップストリームとダウンストリームの依存関係	R	該当なし	高い
インフラストラクチャマッピング	アプリケーションを構成する物理アセットや仮想アセットへのマッピング	R	R	高い
ライセンス	商品ソフトウェアライセンスタイプ (Microsoft SQL Server Enterprise など)	R	R	高い
コスト	ソフトウェアライセンス、ソフトウェアオペレーション、メンテナンスのコスト	該当なし	R	やや高い
ビジネスユニット	マーケティング、財務、営業など	R	O	高い

属性名	説明	発見、設計、移行戦略	推定実行レート	推奨忠実度レベル (最小)
所有者の詳細	アプリケーション所有者の連絡先情報	R	O	高い
アーキテクチャタイプ	ウェブアプリケーション、2層、3層、マイクロサービス、サービス指向アーキテクチャ (SOA) など	R	R	高い
目標復旧時点 (RPO)、目標復旧時間 (RTO)、/ サービスレベルアグリーメント (SLA)	現在のサービス - 管理属性	R	R	高い
収益を生み出すアプリケーションかビジネス戦略アプリケーションか	はい。アプリケーションが会社の収益に直接または間接的に影響を与える場合、またはビジネスによって戦略的と見なされる場合。	R	O	やや高い

属性名	説明	発見、設計、移行戦略	推定実行レート	推奨忠実度レベル (最小)
ユーザー数 (同時)	例えば、内部ユーザー、外部ユーザー、または内部ユーザーや外部ユーザー/顧客など	R	R	やや高い
ユーザーの場所	ユーザーセッションのオリジン	R	R	やや高い
リスクと問題	既知のリスクと問題	R	O	やや高い
移行に関する考慮事項	移行に関連する可能性のある追加情報	R	R	やや高い
移行戦略	例えば、移行用の AWS 6 R の 1 つ	R	R	やや高い
データベースの詳細	例えば、パーティショニング、暗号化、レプリケーション、拡張機能、Secure Sockets Layer (SSL) サポートなど	R	R	高い

属性名	説明	発見、設計、移行戦略	推定実行レート	推奨忠実度レベル (最小)
サポートチーム	例えば、アプリケーションオペレーションチーム名などです。	R	O	やや高い
モニタリングソリューション	このアプリケーションのモニタリングに使用される製品	R	O	やや高い
バックアップ要件	で必要なバックアップスケジュール AWS	R	R	やや高い
DR 情報	例えば、このアプリケーションのディザスタリカバリコンポーネント	R	R	やや高い
ターゲット AWS 要件	コンポーネント、アカウントプレイスメント、ネットワーク、セキュリティなど	R	R	高い

インフラストラクチャ

属性名	説明	発見、設計、移行戦略	推定実行レート	推奨忠実度レベル (最小)
一意の識別子	例えば、サーバー ID などで	R	O	高い

	す。通常、既存の CMDBs またはその他の内部インベントリと管理システムで使用できます。一意の IDs 組織で定義されていない場合は必ず作成することを検討してください。			
ネットワーク名	ネットワーク内のアセット名 (ホスト名など)	R	O	高い
DNS 名 (完全修飾ドメイン名、または FQDN)	[DNS 名]	O	O	やや高い
IP アドレスとネットマスク	内部 IP アドレスおよび/またはパブリック IP アドレス	R	R	高い
アセットタイプ	物理サーバーまたは仮想サーバー、ハイパーバイザー、コンテナ、デバイス、データベースインスタンスなど。	R	R	高い

製品名	商用ベンダーと製品名 (VMware ESXi、IBM Power Systems、Exadata など)	R	R	高い
オペレーティングシステム	例えば、RHEL 8、Windows Server 2019、AIX 6.1 などです。	R	R	高い
構成	割り当てられた CPU、コア数、コアあたりのスレッド数、合計メモリ、ストレージ、ネットワークカード	R	R	高い
使用率	CPU、メモリ、ストレージのピークと平均。データベースインスタンスのスループット。	R	R	高い
ライセンス	商品ライセンスタイプ (RHEL Standard など)	R	R	高い

共有インフラストラクチャですか？	認証プロバイダー、モニタリングシステム、バックアップサービス、および同様のサービスなどの共有サービスを提供するインフラストラクチャサービスを示すにはいまたはいいえ	R	O	高い
アプリケーションマッピング	このインフラストラクチャで実行されるアプリケーションまたはアプリケーションコンポーネント	R	O	高い
通信データ	例えば、プロセスレベルでのサーバー間	R	該当なし	やや高い
ターゲット AWS 要件	インスタンスタイプ、アカウント、サブネット、セキュリティグループ、ルーティングなど	R	R	高い

移行戦略、パターン、ツール	例えば、移行用 6 R の 1 つ、特定の技術パターン、移行ツールなどです。	R	O	高い
リスクと問題	既知のリスクと問題	R	O	やや高い

詳細なアプリケーション評価

詳細なアプリケーション評価の目的は、ターゲットアプリケーションとその関連インフラストラクチャ (コンピューティング、ストレージ、ネットワーク) を完全に理解することです。落とし穴を回避するには、忠実度の高いデータが必要です。例えば、組織がアプリケーションを完全に理解していると仮定するのが一般的です。これは自然であり、多くの場合に当てはまります。ただし、ビジネスへのリスクを最小限に抑えるには、プログラムによるデータを可能な限り取得して、組織の知識と静的なドキュメントを検証することが重要です。これにより、検出プロセスの負荷が軽減されます。ビジネス固有の情報、戦略的ロードマップなど、代替ソースから得られるデータ要素に集中できます。

重要なのは、移行中および移行後の直前の変更を避けることです。例えば、移行時には、サーバーを継続的な移行ウェブに含める必要がある可能性のある未確認の依存関係に基づく変更を避けることが重要です。移行後すぐに、関連するプラットフォーム要件に基づく変更を避けて、トラフィックを許可したり、追加のサービスをデプロイしたりすることが重要です。このような予期しない変更により、セキュリティや運用上の問題のリスクが高まります。詳細なアプリケーション評価を実行するときは、プログラムによる検出ツールを使用してトラフィックパターンと依存関係を検証することを強くお勧めします。

評価の最初に、アプリケーションの利害関係者を特定する必要があります。これらは通常、次のようになります。

- ビジネスユニットリード
- アプリケーション所有者
- アーキテクト
- オペレーションとサポート
- クラウド対応チーム
- コンピューティング、ストレージ、ネットワークなどの特定のプラットフォームチーム

詳細な検出には 2 つのアプローチがあります。トップダウン検出は、アプリケーション、またはユーザーから開始され、インフラストラクチャまで続きます。これは、アプリケーションの識別が明確である場合に推奨されるアプローチです。逆に、ボトムアップ検出はインフラストラクチャから始まり、アプリケーションやサービス、そのユーザーまで到達します。このアプローチは、移行プログラムがインフラストラクチャチームによって推進されている場合や、マッピングが不明な場合 application-to-infrastructure に役立ちます。一般に、両方の組み合わせを使用する可能性があります。

アプリケーションを深く掘り下げるには、既存のアーキテクチャ図から始めることをお勧めします。これらが利用できない場合は、現在の知識に基づいて作成します。単純なリホストまたは再配置移行戦略であっても、このタスクの重要性を過小評価しないでください。アーキテクチャ図をプロットすると、クラウドで小さな変更を迅速に対処できる非効率を特定できます。

トップダウンアプローチとボトムアップアプローチのどちらを実行するかに応じて、最初の図にアプリケーションコンポーネントとサービス、またはサーバーやロードバランサーなどのインフラストラクチャコンポーネントがプロットされます。主要なコンポーネントとインターフェイスを特定したら、検出ツールと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プログラムデータを使用して検証します。ツールは依存関係分析をサポートし、コンポーネント間の通信情報を提供する必要があります。このアプリケーションを構成する各コンポーネントを特定する必要があります。次に、内部と外部の両方で、他のアプリケーションやサービスへの依存関係を文書化します。

依存関係とマッピングを検証するためのツールがない場合は、手動アプローチが必要です。例えば、インフラストラクチャコンポーネントにログインし、スクリプトを実行して、開いているポートや確立された接続などの通信情報を収集できます。同様に、実行中のプロセスとインストールされているソフトウェアを特定できます。手動検出に必要な労力を過小評価しないでください。プログラムによるツールを使用すると、ほとんどの依存関係を数日でキャプチャしてレポートできますが、間隔が長い依存関係 (通常はごく一部の依存関係) は除きます。手動検出では、すべてのデータポイントの収集とマージに数週間かかることがあり、それでもエラーや欠落データが発生しやすくなります。

優先順位付けされた各アプリケーションとマッピングされたインフラストラクチャの [データ要件](#) セクションで指定された情報の取得に進みます。次に、次のアンケートを使用して、詳細な評価プロセスをガイドします。特定された利害関係者とミーティングを行い、これらの質問に対する回答について話し合います。

全般

- このアプリケーションの重要度レベルはどのくらいですか？ 収益は生成されていますか？ これはビジネス戦略アプリケーションですか、それともサポートビジネスアプリケーションですか。他のシステムで共有されているコアインフラストラクチャサービスですか？

- このアプリケーションに進行中の変換プロジェクトはありますか？
- これは内部向けアプリケーションですか、外部向けアプリケーションですか。

アーキテクチャ

- 現在のアーキテクチャタイプ (SOA、マイクロサービス、モノリスなど) は何ですか？ アーキテクチャにはいくつの階層がありますか？ しっかりと結合されているか、緩く結合されているか。
- コンポーネント (コンピューティング、データベース、リモートストレージ、ロードバランサー、キャッシュサービスなど) は何ですか？
- APIs API 名、オペレーション、URLs、ポート、プロトコルなど、これらについて説明します。
- コンポーネント間、およびこのアプリケーションやサービスと他のアプリケーションやサービスとの間で許容される最大レイテンシーはどのくらいですか？

オペレーション

- このアプリケーションはどの場所で動作しますか？
- アプリケーションとインフラストラクチャを運用しているユーザー これらは社内チームまたは AWS パートナーチームによって運営されていますか？
- このアプリケーションがダウンするとどうなりますか？ 影響を受けるのは誰か？ その影響は何ですか？
- ユーザーまたは顧客はどこにいますか？ アプリケーションにどのようにアクセスしますか？ 同時ユーザー数はいくつですか？
- 最後にテクノロジーを更新したのはいつですか？ 今後更新が予定されていますか？ その場合、いつですか？
- このアプリケーションの既知のリスクと問題は何ですか？ 停止、中程度の重大度、重要度の高いインシデントの履歴は何ですか？
- 使用サイクル (営業時間) はどのくらいですか？ 運用タイムゾーンは何ですか？
- 変更のフリーズ期間はどのくらいですか？
- このアプリケーションをモニタリングするにはどのようなソリューションを使用しますか？

パフォーマンス

- 収集されたパフォーマンス情報には何が表示されますか？ 使用量は急増しているか、一定で予測可能か。使用可能なパフォーマンスデータの時間枠、間隔、日付はどのくらいですか？
- このアプリケーションの一部であるか、このアプリケーションとやり取りするスケジュールされたバッチジョブはありますか？

ソフトウェアのライフサイクル

- 現在の変化率 (毎週、毎月、四半期ごと、毎年) はどのくらいですか？
- 開発ライフサイクル (テスト、開発、QA、UAT、本番稼働前、本番稼働など) とは
- アプリケーションとインフラストラクチャのデプロイ方法は何ですか？
- デプロイツールとは
- このアプリケーションまたはインフラストラクチャは、継続的インテグレーション (CI) / 継続的デリバリー (CD) を使用していますか？ 自動化のレベルはどのくらいですか？ 手動タスクとは
- アプリケーションとインフラストラクチャのライセンス要件は何ですか？
- サービスレベルアグリーメント (SLA) とは
- 現在のテストメカニズムは何ですか？ テストステージとは

移行

- 移行に関する考慮事項は何ですか？

この時点で、このアプリケーションを移行する際の考慮事項に注意してください。より完全で正確な評価を行うには、さまざまな利害関係者からこの質問に対する回答を取得します。次に、相手の知識や意見を対照します。

回復性

- 現在のバックアップ方法は何ですか？ バックアップにはどの製品が使用されますか？ バックアップスケジュールとは バックアップ保持ポリシーとは
- 現在の目標復旧時点 (RPO) と目標復旧時間 (RTO) は何ですか？
- このアプリケーションにはディザスタリカバリ (DR) プランがありますか？ その場合は、DR ソリューションとは何ですか？

- 最後の DR テストはいつ行われましたか？

セキュリティとコンプライアンス

- このアプリケーションに適用されるコンプライアンスと規制のフレームワークは何ですか？ 最終監査日と次回の監査日はいつですか？
- このアプリケーションは機密データをホストしていますか？ データ分類とは？
- データは転送中、保管中、またはその両方で暗号化されていますか？ 暗号化メカニズムとは？
- このアプリケーションは Secure Sockets Layer (SSL) 証明書を使用していますか？ 発行機関とは？
- ユーザー、コンポーネント、その他のアプリケーションやサービスの認証方法を教えてください。

データベース

- このアプリケーションはどのデータベースを使用していますか？
- データベースへの一般的な同時接続の数はいくつですか？ 接続の最小数と最大数を教えてください。
- 接続方法 (JDBC、ODBC など) は何ですか？
- 接続文字列は文書化されていますか？ その場合は、どこにありますか？
- データベーススキーマとは？
- データベースはカスタムデータ型を使用していますか？

依存関係

- コンポーネント間の依存関係は何ですか？ 解決できず、コンポーネントを一緒に移行する必要がある依存関係に注意してください。
- コンポーネントは複数の場所に分かれていますか？ これらのロケーション間の接続 (WAN、VPN など) はどのようなものですか？
- このアプリケーションの他のアプリケーションやサービスへの依存関係は何ですか？
- 運用上の依存関係は何ですか？ 例えば、パッチ適用ウィンドウなどのメンテナンスとリリースのサイクルなどです。

AWS アプリケーション設計と移行戦略

アプリケーションの将来の状態を設計および文書化することは、移行の成功の重要な要素です。シンプルでも複雑でも、あらゆるタイプの移行戦略の設計を作成することをお勧めします。設計を作成すると、アーキテクチャが変更されないと予想される場合でも、潜在的なブロッカー、依存関係、アプリケーションを最適化する機会が明らかになります。

また、移行戦略レンズ AWS を使用して、 のアプリケーションの将来の状態に近づくことをお勧めします。この段階では、この移行 AWS の結果としてアプリケーションがどのように表示されるかを必ず定義してください。結果として得られる設計は、移行後のさらなる進化の基盤となります。

次のリストには、設計プロセスに役立つリソースが含まれています。

- [AWS アーキテクチャセンター](#)は、AWS Well-Architected フレームワークなどのツールとガイドを組み合わせています。また、アプリケーションに使用できるリファレンスアーキテクチャも提供します。
- [Amazon Builders' Library](#)には、Amazon がソフトウェアを構築および運用する方法に関するいくつかのリソースが含まれています。
- [AWS ソリューションライブラリ](#)は、によって検証されたクラウドベースのソリューションを多数提供しており AWS、技術的およびビジネス上の問題が数十件あります。リファレンスアーキテクチャの大規模なコレクションが含まれています。
- [AWS 規範的ガイド](#)は、設計プロセスと移行のベストプラクティスに役立つ戦略、ガイド、パターンを提供します。
- [AWS ドキュメント](#)には、ユーザーガイドや API リファレンスなどの AWS サービスに関する情報が含まれています。
- [入門リソースセンター](#)には、で構築を開始できるように、基礎を学ぶための実践的なチュートリアルと詳細な説明がいくつか用意されています AWS。

クラウドジャーニーのどの段階にいるかによっては、AWS 基礎が既に存在する場合があります。これらの AWS 基盤には以下が含まれます。

- AWS リージョンが特定されました。
- アカウントが作成されているか、オンデマンドで取得できます。
- 一般的なネットワークが実装されました。
- 基礎 AWS サービスは アカウント内にデプロイされています。

逆に、プロセスの早い段階で基盤がまだ確立 AWS されていない可能性があります。確立された基盤がないと、アプリケーション設計の範囲が制限されたり、それらを定義するためにさらなる作業が必要になる可能性があります。このような場合は、ランディングゾーンの基本設計をアプリケーション設計作業と並行して定義して実装することをお勧めします。アプリケーション設計は、AWS アカウント構造、ネットワーク、仮想プライベートクラウド (VPCs)、クラスレスドメイン間ルーティング (CIDR) 範囲、共有サービス、セキュリティ、クラウド運用などの要件を特定するのに役立ちます。

[AWS Control Tower](#) は、ランディングゾーンと呼ばれる安全なマルチアカウント AWS 環境をセットアップして管理する最も簡単な方法を提供します。は、AWS Organizations を使用してランディングゾーン AWS Control Tower を作成します。これにより、何千人もの顧客とのコラボレーションにおける継続的なアカウント管理とガバナンス、AWS ベストプラクティスベースの実装が可能になります。

アプリケーションの将来の状態

まず、このアプリケーションの初期移行戦略を確立します。この時点で、戦略は将来の状態設計の一部として変更される可能性があるため、初期と見なされます。これにより、潜在的な制限が明らかになる可能性があります。初期の前提条件を検証するには、[「6 Rs 決定ツリー」](#)を参照してください。また、潜在的な移行フェーズを文書化します。例えば、このアプリケーションは 1 つのイベントに移行されますか (すべてのコンポーネントが同時に移行されますか)。または、これは段階的な移行ですか (一部のコンポーネントは後で移行される)。

特定のアプリケーションの移行戦略は一意ではない場合があることに注意してください。これは、複数の R タイプを使用してアプリケーションコンポーネントを移行できるためです。例えば、最初のアプローチは、アプリケーションを変更せずにリフトアンドシフトすることです。ただし、アプリケーションのコンポーネントは、さまざまな処理を必要とするさまざまなインフラストラクチャアセットに存在する場合があります。例えば、アプリケーションは 3 つのコンポーネントで構成され、それぞれが別のサーバーで実行され、いずれかのサーバーはクラウドでサポートされていないレガシーオペレーティングシステムを実行します。そのコンポーネントにはリプラットフォームアプローチが必要ですが、サポートされているサーバーバージョンで実行されている他の 2 つのコンポーネントはリホストできます。移行する各アプリケーションコンポーネントおよび関連するインフラストラクチャに移行戦略を割り当てることが重要です。

次に、コンテキストと問題を文書化し、現在の状態を定義する既存のアーティファクトをリンクします。

- このアプリケーションが移行される理由
- 提案された変更は何ですか？

- 利点は何ですか？
- 重大なリスクやブロッカーはありますか？
- 現在の欠点は何ですか？
- 範囲内と範囲外のものは何ですか？

再現性

設計作業全体を通して、このアプリケーションのソリューションとアーキテクチャを他のアプリケーションに再利用する方法を検討してください。このソリューションは一般化できますか？

要件

セキュリティを含め、このアプリケーションの機能要件と非機能要件を文書化します。これには、選択した移行戦略に応じて、現在と将来の状態の要件が含まれます。詳細なアプリケーション評価で収集された情報を使用して、このプロセスをガイドします。

To-Be アーキテクチャ

このアプリケーションの将来のアーキテクチャについて説明します。ソース環境 (オンプレミス) とターゲット AWS 環境 (ターゲット AWS リージョン、アカウント、VPC、アベイラビリティゾーンなど) VPCs構成要素を含む再利用可能な図テンプレートを作成することを検討してください。

移行されるコンポーネントと新しいコンポーネントのテーブルを作成します。このアプリケーションとやり取りする他のアプリケーションやサービス (オンプレミスまたはクラウド) を含めます。

次の表に、コンポーネントの例を示します。リファレンスアーキテクチャや検証済みの設定を表すものではありません。

名前	説明	詳細
アプリケーション	外部サービス (インバウンド接続)	サービスは、公開された API からのデータを消費します。
DNS	名前解決 (内部)	ベースラインアカウント設定の一部としてデプロイされた Amazon Route 53

名前	説明	詳細
Application Load Balancer	バックエンドサービス間でトラフィックを分散する	オンプレミスのロードバランサーを置き換えます。プール A を移行します。
アプリケーションセキュリティ	DdoS 保護	を使用して実装 AWS Shield
セキュリティグループ	仮想ファイアウォール	ポート 443 (インバウンド) のアプリケーションインスタンスへのアクセスを制限します。
サーバー A	フロントエンド	Amazon Elastic Compute Cloud (Amazon EC2) を使用してリホストします。
サーバー B	フロントエンド	Amazon EC2 を使用してリホストします。
サーバー C	アプリケーションロジック	Amazon EC2 を使用してリホストします。
サーバー D	アプリケーションロジック	Amazon EC2 を使用してリホストします。
Amazon Relational Database Service (Amazon RDS) – Amazon Aurora	データベース	サーバー E と F を置き換えます
モニタリングとアラート	変更管理	Amazon CloudWatch
監査ログ	変更管理	AWS CloudTrail
パッチ適用とリモートアクセス	メンテナンス	AWS Systems Manager
リソースアクセス	安全なアクセスコントロール	AWS Identity and Access Management (IAM)

名前	説明	詳細
認証	ユーザーアクセス	Amazon Cognito
証明書	SSL/TLS	AWS Certificate Manager
API 1	外部 API	Amazon API Gateway
オブジェクトストレージ	イメージホスティング	Amazon Simple Storage Service (Amazon S3)
認証情報	認証情報の管理とホスティング	AWS Secrets Manager
AWS Lambda 関数	データベース認証情報と API キーの取得	AWS Lambda
インターネットゲートウェイ	アウトバウンドインターネットアクセス	VPC へのインターネットゲートウェイ
プライベートサブネット 1	バックエンドと DB	アベイラビリティーゾーン 1 – VPC 1
プライベートサブネット 2	バックエンドと DB	アベイラビリティーゾーン 2 – VPC 1
パブリックサブネット 1	フロントエンド	アベイラビリティーゾーン 1 – VPC 1
パブリックサブネット 2	フロントエンド	アベイラビリティーゾーン 2 – VPC 1
バックアップサービス	データベースと EC2 インスタンスのバックアップ	AWS Backup
DR	Amazon EC2 の耐障害性	CloudEndure デイザスタリカバリ

コンポーネントを特定したら、任意のツールを使用して図にプロットします。初期設計を、アプリケーション所有者、エンタープライズアーキテクト、プラットフォームチーム、移行チームなど、主要なアプリケーションステークホルダーと共有します。次の質問をすることを検討してください。

- チームは通常、設計に同意していますか？
- 運用チームはサポートできますか？
- 設計は進化できますか？
- 他にオプションはありますか？
- 設計はアーキテクチャ標準とセキュリティポリシーに準拠していますか？
- 不足しているコンポーネント (コードリポジトリ、CI/CD ツール、VPC エンドポイントなど) はありますか？

アーキテクチャ上の意思決定

設計プロセスの一環として、アーキテクチャ全体またはその特定の部分に対してより多くのオプションが見つかる可能性があります。これらのオプションを、優先または選択したオプションの理論的根拠とともに文書化します。これらの決定は、アーキテクチャ上の決定として文書化できます。

メインオプションがリスト化され、新しいリーダーが別のオプションを使用する決定の背後にあるオプションと理由を理解するのに十分な詳細で説明されていることを確認します。

ソフトウェアライフサイクル環境

現在の環境への変更を文書化します。例えば、テスト環境と開発環境は で再作成され AWS、移行されません。

タグ付け

各インフラストラクチャコンポーネントの必須および推奨のタグ付けと、この設計のタグ付け値について説明します。

移行戦略

設計のこの時点で、移行戦略に関する最初の仮定を検証する必要があります。選択した R 戦略にコンセンサスがあることを確認します。全体的なアプリケーション移行戦略と個々のアプリケーションコンポーネントの戦略を文書化します。前述のように、アプリケーションコンポーネントによって移行に異なる R タイプが必要になる場合があります。

さらに、移行戦略を主要なビジネス推進要因と成果に合わせます。また、さまざまな移行イベントにおけるコンポーネントの移動など、移行への段階的なアプローチについても説明します。

6 R の決定の詳細については、[AWS Migration Hub 「戦略の推奨事項」](#) を参照してください。

移行パターンとツール

アプリケーションおよびインフラストラクチャコンポーネント用の移行戦略を定義することで、特定の技術パターンを探索できるようになりました。例えば、リホスト戦略は、[AWS Application Migration Service](#) などの移行ツールで実装できます。状態やデータをレプリケートする必要がない場合は、Amazon マシンイメージ (AMI) とアプリケーションデプロイパイプラインを使用してアプリケーションを再デプロイすることで、同じ結果を実現できます。

同様に、アプリケーションをリプラットフォームまたはリファクタリング (リアーキテクト) するには、[AWS App2Container](#)、[AWS Database Migration Service](#) ()、[AWS Schema Conversion Tool](#) (AWS DMS)、AWS SCTなどのツールを使用できます[AWS DataSync](#)。コンテナ化には、[Amazon Elastic Container Service](#) (Amazon ECS)、[Amazon Elastic Kubernetes Service](#) (Amazon EKS)、または [AWS Fargate](#) を使用できます。再購入するときは、特定の製品に AMI を使用するか、[AWS Marketplace](#) の Software as a Service (SaaS) ソリューションを使用できます。

目標を達成するために利用できるさまざまなパターンとオプションを評価します。長所と短所、移行運用の準備状況を考慮してください。分析に役立てるには、次の質問を使用します。

- 移行チームはこれらのパターンをサポートできますか？
- コストと利点のバランスはどれくらいですか？
- このアプリケーション、サービス、またはコンポーネントをマネージドサービスに移動できますか？
- このパターンを実装するための労力は何ですか？
- 特定のパターンの使用を妨げる規制やコンプライアンスポリシーはありますか？
- このパターンは再利用できますか？ 再利用可能なパターンが推奨されます。ただし、パターンが 1 回だけ使用される場合があります。代替の再利用可能なパターンよりも、シングルユースパターンの労力のバランスを検討してください。

[AWS 規範ガイダンス](#)には、さまざまな移行パターンと手法が含まれています。

サービス管理と運用

への移行のためのアプリケーション設計を作成するときは AWS、運用準備を検討してください。アプリケーションチームとインフラストラクチャチームの準備状況要件を評価するときは、次の質問を考慮してください。

- 運用する準備はできていますか？
- インシデント対応手順は定義されていますか？
- 期待されるサービスレベルアグリーメント (SLA) とは
- 職務の分離は必要ですか？
- さまざまなチームがサポートアクションを調整する準備ができていますか？
- 誰が何を担当しますか？

カットオーバーに関する考慮事項

移行戦略とパターンを考慮すると、アプリケーションが移行された時点で知っておくべき重要なことは何ですか？ カットオーバー計画は、設計後のアクティビティです。ただし、予想されるアクティビティと要件に関する考慮事項があれば文書化してください。例えば、該当する場合は概念実証を実行する要件を文書化し、テスト、監査、または検証の要件の概要を説明します。

リスク、前提条件、問題、依存関係

未解決のリスク、仮定、潜在的な問題をすべて文書化します。これらの項目に明確な所有権を割り当て、全体的な設計と戦略の実装を承認できるように進捗状況を追跡します。さらに、この設計を実装するための主要な依存関係を文書化します。

実行コストの見積もり

ターゲット AWS アーキテクチャのコストを見積もるには、[AWS 料金計算ツール](#) を使用します。設計で定義されているインフラストラクチャコンポーネントを追加し、推定実行コストを取得します。アプリケーションコンポーネントに必要なソフトウェアライセンスと、使用する AWS サービスにまだ含まれていないソフトウェアライセンスを考慮します。

ポートフォリオ分析と移行計画

この評価段階は、ポートフォリオの発見と[初期計画セクション](#)で開始されたポートフォリオレベルの発見と分析を完了することに焦点を当てています。目標は、アプリケーションとインフラストラクチャの初期ポートフォリオを繰り返してベースラインを確立することです。このベースラインには、すべての依存関係の特定、移行の合理化モデルの反復、詳細なビジネスケースの作成、移行ウェブプランの概説が含まれます。その結果、必要なデータ忠実度は高くなります。この段階では、時間投資が必要になります。評価結果を高速化するには、検出ツールなどのプログラムによるデータソースをできるだけ多く使用することをお勧めします。

このステージの主な結果は次のとおりです。

- 忠実度の高いアプリケーションとインフラストラクチャのインベントリ
- 各アプリケーションの高レベルの移行戦略
- 信頼性の高い移行ウェブプラン
- 詳細なビジネスケース

完全な評価データ要件を理解する

次の表は、移行内のアプリケーションとそれに関連するインフラストラクチャの完全なポートフォリオビューを取得するために必要な情報を示しています。

テーブルでは、次の略語を使用します。

- 必須の R
- O、オプション
- 該当なし、該当なし

アプリケーション

属性名	説明	インベントリと優先順位付け	詳細なビジネスケース	推奨忠実度レベル (最小)
一意の識別子	例えば、アプリケーション ID などです。通常、	R	R	高い

属性名	説明	インベントリと優先順位付け	詳細なビジネスケース	推奨忠実度レベル (最小)
	既存の CMDBs またはその他の内部インベントリと管理システムで使用できません。一意の IDs 組織で定義されていない場合は必ず作成することを検討してください。			
アプリケーション名	このアプリケーションが組織で認識される名前。必要に応じて、商用 off-the-shelf (COTS) ベンダーと製品名を含めます。	R	R	高い
COTS ですか？	はいまたはいいえ。これは商用アプリケーションか内部開発か	R	R	高い
COTS 製品とバージョン	商用ソフトウェア製品名とバージョン	R	R	高い
説明	プライマリアプリケーションの関数とコンテキスト	R	R	高い

属性名	説明	インベントリと優先順位付け	詳細なビジネスケース	推奨忠実度レベル (最小)
緊急性	例えば、戦略的アプリケーションや収益を生み出すアプリケーション、重要な機能のサポートなどです。	R	R	高い
タイプ	データベース、顧客関係管理 (CRM)、ウェブアプリケーション、マルチメディア、IT 共有サービスなど	R	R	高い
環境	例えば、本番稼働、本番稼働前、開発、テスト、サンドボックスなどです。	R	R	高い
コンプライアンスと規制	ワークロードに適用されるフレームワーク (HIPAA、Sox、PCI-DSS、ISO、SOC、FedRAMP など) と規制要件	R	R	高い

属性名	説明	インベントリと優先順位付け	詳細なビジネスケース	推奨忠実度レベル (最小)
依存関係	内部および外部のアプリケーションまたはサービスへのアップストリームとダウンストリームの依存関係。運用要素 (メンテナンスサイクルなど) などの非技術的な依存関係	R	O	高い
インフラストラクチャマッピング	アプリケーションを構成する物理アセットや仮想アセットへのマッピング	R	R	高い
ライセンス	商品ソフトウェアライセンスタイプ (Microsoft SQL Server Enterprise など)	R	R	やや高い
コスト	ソフトウェアライセンス、ソフトウェアオペレーション、メンテナンスのコスト	該当なし	R	やや高い

属性名	説明	インベントリと優先順位付け	詳細なビジネスケース	推奨忠実度レベル (最小)
ビジネスユニット	マーケティング、財務、営業など	R	R	高い
所有者の詳細	アプリケーション所有者の連絡先情報	R	R	高い
DR 情報	ディザスタリカバリコンポーネント	R	R	高い
移行戦略	例えば、に移行するための 6 R の 1 つ AWS	R	R	高い
サポートチケット	停止、速度低下、トランザクションスロットリング、バッチウィンドウのオーバーランによる生産性と財務への影響を評価するのに役立つ 12~24 か月分のデータ	O	R	中程度

インフラストラクチャ

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
一意の識別子	例えば、サーバー ID などです。通常、既存の CMDBs またはその他の内部インベントリと管理システムで使用できます。一意の IDs 組織で定義されていない場合は必ず作成することを検討してください。	R	R	高い
ネットワーク名	ネットワーク内のアセット名 (ホスト名など)	R	R	高い
DNS 名 (完全修飾ドメイン名、または FQDN)	[DNS 名]	R	O	高い
IP アドレスとネットマスク	内部 IP アドレスおよび/またはパブリック IP アドレス	R	R	高い
アセットタイプ	物理サーバーまたは仮想サーバー、ハイパーバイザー、コンテナ、デバイス、データベ	R	R	高い

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
	スインスタンスなど。			
製品名	商用ベンダーと製品名 (VMware ESXi、IBM Power Systems、Exadata など)	R	R	高い
オペレーティングシステム	例えば、RHEL 8、Windows Server 2019、AIX 6.1 などです。	R	R	高い
構成	割り当てられた CPU、コア数、コアあたりのスレッド数、合計メモリ、ストレージ、ネットワークカード	R	R	高い
使用率	CPU、メモリ、ストレージのピークと平均。データベースインスタンスのスループット。	R	R	高い
ライセンス	商品ライセンスタイプ (RHEL Standard など)	R	R	高い

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
共有インフラストラクチャですか？	認証プロバイダー、モニタリングシステム、バックアップサービス、および同様のサービスなどの共有サービスを提供するインフラストラクチャサービスを示すにはいまたはいいえ	R	R	高い
アプリケーションマッピング	このインフラストラクチャで実行されるアプリケーションまたはアプリケーションコンポーネント	R	R	高い

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
コスト	ハードウェア、メンテナンス、オペレーション、ストレージ (SAN、NAS、オブジェクト)、オペレーティングシステムライセンス、ラックスペースの共有、データセンターのオーバーヘッドなど、ベアメタルサーバーのフルロードコスト	該当なし	R	やや高い
推定データ転送量 (入力/出力)	例えば、30 日間の 1 日あたりのインフラストラクチャアセットごと	O	R	中程度

ネットワーク

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
パイプのサイズ (MB/秒)、冗長性 (Y/N)	現在の WAN リンク仕様 (例: 1000 Mb/s 冗長)	R	R	やや高い

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
リンク使用率	ピーク使用率と平均使用率、アウトバウンドデータ転送 (GB/月)	R	R	やや高い
レイテンシー (ms)	接続されたロケーション間の現在のレイテンシー。	R	O	高い
コスト	現在の 1 か月あたりのコスト	該当なし	R	やや高い

移行

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
リホスト	各ワークロード (人日)、1 日あたりのカスタマーとパートナーのコストレー、ツールコスト、ワークロード数に関するカスタマーとパートナーの労力	該当なし	R	やや高い
リプラットフォーム	各ワークロード (人日)、1 日あたりの顧客およびパートナーの	該当なし	R	やや高い

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
	コストレート、ワークロード数に関する顧客およびパートナーの労力			
リファクタリング	各ワークロード (人日)、1日あたりの顧客およびパートナーのコストレート、ワークロード数に関する顧客およびパートナーの労力	該当なし	R	やや高い
リタイア	サーバー数、平均廃止コスト	該当なし	R	やや高い
ランディングゾーン	既存の (はい/いいえ) の再利用、必要な AWS リージョンのリスト、コスト	該当なし	R	やや高い
人と変化	クラウド運用と開発でトレーニングするスタッフ数、1人あたりのトレーニングコスト、1人あたりのトレーニング時間のコスト	該当なし	R	やや高い

属性名	説明	インベントリと優先順位付け	ビジネスケース	推奨忠実度レベル (最小)
duration	対象範囲内のワークロード移行の期間 (月)	O	R	やや高い
並列コスト	移行中に現状のままコストを削除できる時間枠とレート	該当なし	R	やや高い
	移行中に AWS 製品やサービス、およびその他のインフラストラクチャコストが導入される時間枠とレート	該当なし	R	やや高い

アプリケーションポートフォリオのベースラインの確立

信頼性の高い移行ウェーブプランを作成するには、アプリケーションのポートフォリオとそれに関連するインフラストラクチャのベースラインを確立する必要があります。ポートフォリオベースラインは、技術的な依存関係や移行戦略など、移行範囲の包括的なビューを提供します。ポートフォリオベースラインは、移行の対象となるアプリケーションを明確にし、[「評価データ要件の理解」セクションで概説されているデータ](#)ポイントが収集されることを示します。同様に、関連するすべてのインフラストラクチャ (コンピューティング、ストレージネットワーク) が理解され、アプリケーションにマッピングされます。

技術的な依存関係は、次の 4 つのカテゴリで説明できます。

- 依存関係は application-to-infrastructure、ソフトウェアと物理ハードウェアまたは仮想ハードウェア間のリンクを確立します。例えば、CRM アプリケーションとインストールされている仮想マシンの間には依存関係があります。
- アプリケーションコンポーネントの依存関係は、さまざまなインフラストラクチャアセットで実行されているコンポーネントがどのように相互作用するかを記述します。アプリケーションコンポー

メントの依存関係の例としては、仮想マシンで実行されるウェブフロントエンド、別の仮想マシンで実行されるアプリケーションレイヤー、データベースクラスターで実行されるデータベースなどがあります。

- 依存関係は application-to-application、アプリケーションまたはアプリケーションコンポーネントと他のアプリケーションまたはそのコンポーネントとのやり取りに関連しています。application-to-application 依存関係の例としては、支払い処理アプリケーションと株式管理アプリケーションがあります。これらのアプリケーションは独立していますが、定義された API オペレーションを使用して常にやり取りします。
- インフラストラクチャ application-to-infrastructure サービス自体がアプリケーションであることを考えると、サービスの依存関係は技術的には application-to-application 依存関係です。ただし、これらを個別に分類することをお勧めします。主な理由は、インフラストラクチャサービスは通常、多くのアプリケーションによって共有されるため、依存関係の長い証跡があることです。また、通常、異なる移行戦略とパターンに従います。例えば、ロードバランサーには複数のアプリケーションのバランシングプールを含めることができます。重要なのは、プールへの依存関係です。これは依存アプリケーションとともに個別に移行される可能性が高いですが、ロードバランサー自体は保持されるか、retired.in を追加することで、application-to-infrastructure サービスの依存関係を個別化することで、誤った依存関係グループを回避できます。誤った依存関係グループは、複数のビジネスアプリケーションがグループ化されている場合であり、インフラストラクチャサービスへの共通の依存関係を持つを同時に移行する必要があることを意味します。例えば、Active Directory などの認証サービスは、大規模なアプリケーショングループに関連付けられている可能性があります。重要なのは、これらのアプリケーションに個別にアプローチし、クラウド環境で AWS Directory Service for Microsoft Active Directory などのサービスを有効にして依存関係に対処することです。

ポートフォリオのベースラインを確立するときは、アプリケーションコンポーネントごとに移行戦略を確認することをお勧めします。移行戦略は、移行用の 6 R の 1 つになります ([「6 R の移行戦略の反復」](#) セクションを参照)。ポートフォリオベースラインでは、6 つの R のいずれかを各アプリケーションに関連付ける必要があります。6 R 戦略は、アプリケーションの各インフラストラクチャコンポーネントにも関連付ける必要があります。

依存関係や移行戦略を含むポートフォリオのベースラインバージョンを確立するには、自動検出ツールを使用します ([「検出ツールの必要性の評価」](#) を参照)。アプリケーション所有者やインフラストラクチャチームなどの主要な利害関係者から収集された情報でデータを補完します。このステージのデータ要件セクションで説明されている属性と忠実度のレベルに一致する完全なポートフォリオインベントリを取得するまで、データを収集し続けます。結果として得られるデータセットは、移行の推進に役立ちます。

移行の範囲と使用可能なツールによっては、このアクティビティが完了するまでに数週間かかる場合があることを考慮してください。

優先順位付け基準の反復

移行ウェブプランを作成する前に、アプリケーションの優先順位付け基準を繰り返して、パイロットアプリケーションの選択から長期的なウェブプランニングに転換することをお勧めします。

前のセクションでは、シンプルなクラウド対応アプリケーションを優先するデフォルトの優先順位付け基準を導入しました（「[アプリケーションの優先順位付け](#)」を参照）。これは、初期段階では、移行プロセスを改良し、学習した教訓を組み込むために、重要ではないアプリケーションから開始することをお勧めします。ただし、この段階では、長期計画を作成するには、アプリケーションを移行する順序をビジネスドライバーに合わせて調整する必要があります。新しい基準を適用すると、ウェブプランニングの重要な入力となるアプリケーションの新しいランキングが生成されます。

アプリケーションポートフォリオから利用可能なデータポイントを確認し、ビジネスドライバーに基づいてアプリケーションの優先順位付けを決定する属性を選択します。

まず、ビジネスドライバーを検証します（「[ビジネスドライバーと技術指針](#)」を参照）。次に、ビジネスドライバーに基づいて、移行するアプリケーションの優先順位付けに役立つ属性を選択します。

次の表は、イノベーションのビジネス推進要因に沿った優先順位付け基準の例を示しています。

属性またはデータポイント	使用できる値	スコア (0 ~ 99)	重要度または関連性の乗算係数
オペレーティングシステム	AIX	80	高 (1x)
	Solaris	80	
	HP-UX	80	
	メインフレーム	70	
	Windows	50	
	Linux	20	
ビジネスの重要性	高い	60	高 (1x)

属性またはデータポイント	使用できる値	スコア (0 ~ 99)	重要度または関連性の乗算係数
	中程度	40	
	低	20	
アーキテクチャ	緊密に結合	60	高 (1x)
	疎結合	20	
運用モデル	繁体字 - CI/CD なし	60	中高 (0.8x)
	基本的な CI/CD	40	
	フル DevOps	20	
コンピューティングインスタンスの数	1 ~ 3	60	中高 (0.8x)
	4-10	40	
	11 以上	20	
移行戦略	リファクタリング (リ アーキテクト)	70	中 (0.6x)
	リプラットフォーム	40	
	再購入	30	
	リホスト	10	

次の表は、迅速なコスト削減のためのビジネス推進要因に沿った優先順位付け基準の例を示しています。

属性またはデータポイント	使用できる値	スコア (0 ~ 99)	重要性または関連性の乗算係数
データベース製品	Oracle	70	高 (1x)

属性またはデータポイント	使用できる値	スコア (0 ~ 99)	重要性または関連性の乗算係数
	Microsoft SQL	70	
	その他	20	
オペレーティングシステム	Windows	70	高 (1x)
	Linux	70	
	その他	20	
CPU 使用率 (平均)	36% 以上	60	高 (1x)
	36% 未満	40	
コンピューティングインスタンスの数	11 以上	60	中高 (0.8x)
	4-10	40	
	1 ~ 3	20	
移行戦略	リタイア	80	中 (0.6x)
	リホスト	70	
	リプラットフォーム	50	
	リファクタリング (リアーキテクト)	10	

優先順位付け基準をテストし、一般的に出力に同意するまで繰り返します。ベースラインバージョンを取得するには、少なくとも 3~4 回の反復が必要です。

6 R の移行戦略の選択を繰り返す

この段階では、6 Rs 決定ツリーを繰り返して進化させることをお勧めします。[「移行用の R タイプの決定」](#) セクションでは、デフォルトの決定木が導入されました。ツリーを改訂し、最初のパイロットアプリケーションの移行全体で学習したことを考慮し、ビジネスドライバー、優先順位付け基準、

および独自の状況に引き続き一致させることをお勧めします。サンプルアプリケーションを使用して決定木を検証し、それでも期待される戦略が生成されることを確認します。それ以外の場合は、それに応じてロジックを更新します。結果のツリーは、アプリケーションのポートフォリオのベースラインを確立し、各アプリケーションコンポーネントの移行戦略を割り当てる上で重要です。

前の [6 Rs セクション](#) で説明したように、6 Rs はインフラストラクチャにも適用され、それに応じて割り当てることも同様に重要です。特定のアプリケーションコンポーネントには移行戦略がありませんが、インフラストラクチャレベルでは、各インフラストラクチャアセットは、サポートするアプリケーションコンポーネントに対して確立された戦略とは異なる可能性のある特定の移行戦略に従います。

6 R 決定ツリーはアプリケーションコンポーネントにのみ適用されます。インフラストラクチャの移行戦略は、アプリケーション用に選択された戦略から算出されます。例えば、リプラットフォームされるアプリケーションコンポーネントの場合、それをホストする現在のインフラストラクチャは廃止される可能性があります。

移行戦略が各アプリケーションコンポーネントおよび関連するインフラストラクチャに割り当てられていることを確認します。この情報は、必要な労力、キャパシティ、スキルの見積もりや、移行ウェーブプランの作成時に重要な要素になります。

6 R の決定の詳細については、[AWS Migration Hub 「戦略の推奨事項」](#) を参照してください。

ウェーブプランニング

ウェーブプランニングでは、依存関係グループは、解決できない技術的および非技術的な依存関係を持つアプリケーションとインフラストラクチャのコレクションです。これらの依存関係のため、依存関係グループ内のアプリケーションとインフラストラクチャは、同時に、または特定の日付に移行する必要があります。例えば、仮想マシンで実行されているアプリケーションと、別の仮想マシンで実行されているデータベースでは、レイテンシー要件が低いか、トラフィック量が多く、クエリが複雑なアプリケーションは、クラウドで1つのコンポーネントを運用するのではなく、一緒に移行される可能性が高くなります。同様に、同様の低レイテンシー要件を持つ API を介してやり取りする独立したアプリケーションも同時に移行されます。

移行ウェーブは通常 4~8 週間で、1 つ以上の移行イベントを含めることができます。依存関係グループはウェーブに結合され、ウェーブに 1 つ以上の依存関係グループを含めることができます。ウェーブには、移行に必要な他のアクティビティも含まれています。これには、AWS インフラストラクチャのセットアップ (ランディングゾーン、セキュリティ、運用など)、移行ツール、データレプリケーション、カットオーバー計画、テスト、移行後のサポートなどの移行アクティビティが含まれます。

成功を測定し、進捗状況を追跡するには、ウェーブを成果とビジネスドライバーに合わせる必要があります。これは、ウェーブ期間とウェーブに含まれる依存関係グループにも影響します。ウェーブの完了には、測定可能なアチーブメントを反映する必要があります。ウェーブの計画は、技術指針の原則などの他の要素を組み合わせることもできます。例えば、ウェーブは環境 (開発、テスト、本番など) または移行戦略 (リホストウェーブ、リプラットフォームウェーブなど) で定義できます。

効果的で信頼性の高い移行ウェーブプランを作成するには、アプリケーションポートフォリオ、関連するインフラストラクチャ (コンピューティング、ストレージ、ネットワーク)、依存関係マッピング、移行戦略の全体像を把握する必要があります。

アプリケーションポートフォリオのベースラインの確立に関するセクションでは、技術的な依存関係の 4 つのカテゴリについて説明しました。これらの依存関係は、移行ウェーブの作成と依存関係グループの定義に役立ちます。依存関係グループは、依存関係の重要度によって決まります。さらに、非技術的な依存関係も考慮する必要があります。例えば、アプリケーションのリリーススケジュール、メンテナンスウィンドウ、四半期末処理などの主要な営業日がウェーブプランに影響します。

依存関係がソフトかハードかを判断します。ソフト依存関係は、2 つ以上のアセット間、またはアセットから制約との関係であり、コンポーネントの場所には依存しません。例えば、同じローカルネットワーク (または同じインフラストラクチャ) で動作する 2 つのシステムは、一方のシステムをクラウドに移動し、もう一方のシステムはオンプレミスのままにすることで分割できます。もう 1 つの例は、メンテナンスアクティビティに影響を与えずにメンテナンスウィンドウ中に移行できるシステムです。

ハード依存関係とは、2 つ以上のアセット、またはアセットから制約との関係であり、場所によって異なります。例えば、同じローカルネットワークで動作し、アプリケーションサーバーとデータベースサーバー間の通信の低レイテンシーに大きく依存する 2 つのシステムには、厳しい依存関係があります。これらのシステムの 1 つだけをクラウドに移動すると、機能またはパフォーマンスの問題は解決できません。同様に、リソースの可用性 (移行を実行するチームなど) などの非技術的な理由や、2 つのシステムを特定の時間枠にのみ移行できるメンテナンスウィンドウなどの運用上の制約により、これらのアセットに厳しい依存関係が生じる可能性があります。

移行ウェーブプランを作成するには、依存関係を分析して依存関係グループを決定し、理想的には特殊な検出ツールなどの信頼性の高いデータソースからこの情報をアプリケーションの優先順位付け基準や運用状況と組み合わせます。優先順位付けランキングの最上位にあるアプリケーションは、最初の移行ウェーブの対象となる必要があります。リソースの可用性、リスク許容度、ビジネスおよび技術上の制約、経験、スキルに基づいて、ウェーブ容量 (ウェーブに含めることができるアプリケーションの数) を決定します。プロセス全体を通じてサポートする AWS スペシャリストを提供できるプロフェッショナルサービスまたは AWS 移行コンピテンシーパートナーとの連携を検討してください。

優先順位付け基準は、アプリケーションをクラウドに移行する順序の初期指標です。ただし、依存関係グループは、特定の時点で移動されるアプリケーションの実際の決定要因になります。これは、優先度が高いとランク付けされたアプリケーションは、ランキングの中間または最下位にあるアプリケーションに厳しい依存関係を持つ可能性があるためです。

移行戦略はウェブの構成にも影響します。例えば、数週間または数か月の分析、設計、テスト、準備が必要になる可能性のあるリファクタリング戦略を必要とする優先度の高いアプリケーションは、後続の波に陥る可能性があります。

ウェブプランの作成

アプリケーションの波を移行するための前提条件は、アプリケーションポートフォリオデータと、波で移行されるアプリケーションのグループの詳細なアプリケーション評価です。詳細な評価には、ウェブ内のアプリケーションのリスト、関連するインフラストラクチャの詳細、ターゲット設計、各アプリケーションの移行戦略を含める必要があります。

ウェブの所有権とガバナンスを確立することは、ウェブワーク、プログラムの依存関係、変更管理、問題、リスクを管理および追跡する上で重要です。計画を管理するためのガバナンスフレームワークが整っていることを確認します。

ウェブプランの概要を表示するには、デフォルトのウェブコンストラクトから始めます。ウェブ内では何が起こりますか？ 最初の入力定義された後、ウェブが開始する可能性があります。通常、アクティビティは次のようになります。

1. カットオーバープランを絞り込みます。このアクティビティでは、他の社内および外部チームとの調整など、移行時に実行する必要があるランブックとステップの概要を説明する必要があります。
2. ロールバックプランを絞り込みます。問題が発生した場合にアプリケーションをロールバックするにはどうすればよいですか？
3. ターゲットインフラストラクチャを準備します。例えば、AWS ランディングゾーン (AWS アカウント、セキュリティ、ネットワーク、インフラストラクチャサービス、その他のサポートインフラストラクチャ) を作成または拡張できます。
4. ターゲットインフラストラクチャをテストします。
5. 移行ツールを運用します。例えば、レプリケーションエージェントをインストールし、データ転送を開始します。
6. カットオーバープランとランブックのドライランを実行します。参加しているチームメンバー全員をグループ化し、すべてのステップを事前に確認します。
7. データレプリケーションとインフラストラクチャのデプロイをモニタリングします。

8. でインフラストラクチャとアプリケーションの運用の準備が整っていることを確認します AWS。
9. セキュリティの準備状況を確認します。
- 10 該当する場合は、コンプライアンス要件と規制要件 (ワークロード検証の移行前と移行後など) を確認します。
11. アプリケーションを に移行 AWS し、本番稼働前のテストを実行します。
- 12 運用チームと移行チームが問題を解決するために完全に対応できる 3 日間など、移行後のサポートを提供し、最適化を適用します。
- 13 移行後のレビューを実施します。学んだ教訓を文書化し、将来の波に組み込んでください。
- 14 運用上の引き継ぎとレポート用のメトリクスの取得を確認して、ウェーブクロージャを実行します。

これらの各アクティビティにかかる時間は、スコープの複雑さ、ウェーブキャパシティ、関係する人、および固有の状況によって決まります。可能な限り、遅延や移行ブロッカーの影響を軽減するため、ウェーブを小さくすることをお勧めします。チームで、ウェーブのデフォルト期間を決定します。

次に、日付を分析して、空のウェーブの初期の高レベル構造を作成します (まだアプリケーションが割り当てられていません)。以下の質問を検討してください。

- 移行プログラムの合計期間はどれくらいですか？
- 期限はどのくらいですか？
- データセンターの終了日は固定されていますか？
- コロケーション契約の終了日はありますか？
- アプリケーションとインフラストラクチャの更新サイクルは何ですか？
- アプリケーションのメンテナンスとリリースのサイクルは何ですか？
- 移行を避ける日付はありますか (リリースとメンテナンスのサイクル、年末、祝日、月末処理など)。

これらの考慮事項を使用して、ウェーブを計画にプロットします。移行プロセスを高速化するには、可能な場合はウェーブを重複させることをお勧めします。波が重複する鍵は、波内で何が起こるかを定義して検討することです。通常、デプロイアクティビティ、ターゲットインフラストラクチャの検証、データ同期はウェーブの前半に行われます。後半では、実際の移行、テスト、運用の引き継ぎに焦点を当てます。つまり、プロセスの各半分にはさまざまなチームが関与しており、ある程度の効率を得ることが可能です。例えば、ターゲットインフラストラクチャの準備に関与したチームが作業を

完了するとすぐに、次のウェーブの要件に取り組むことができます。一般に、移行に対するファクトリーのようなアプローチを容易にするために、ほとんどのウェーブの長さや構造が似ていることをお勧めします。ただし、ウェーブ計画プロセスでは、依存関係や運用要件を満たすように特定のウェーブのサイズを拡張できます。

次に、特定された依存関係グループに基づいて、ウェーブの最大サイズを、含めることができる依存関係グループの数の観点から決定します。ウェーブサイズは通常、リスク選好度 (許容できる並列変更の量など) とリソースの可用性 (使用可能なリソース、スキル、予算で実行できる並列変更の量など) によって決まります。ただし、早期計画では、リソースの要件と可用性によって制限されることはありません。複数の依存関係グループを含むウェーブは、将来の反復で小さなウェーブに分解できます。

特定のウェーブの依存関係グループが確認されたら、ウェーブを移行するためのリソース要件を確認します。リソース要件に基づいてウェーブサイズ (含まれる依存関係グループの数) を調整することを検討してください。これにより、ウェーブが小さくなったり大きくなったりする可能性があります。すべてのウェーブが定義されるまで、必要に応じてウェーブプランを繰り返します。

変更の管理

アプリケーションおよび関連するインフラストラクチャのポートフォリオは、移行プログラムのライフサイクル中に変更されます。実行時間の長い移行プログラムは、通常のビジネスの進化と変化と共存します。アプリケーションは、移行を待つにつれて進化し続けます。サーバーが追加または削除され、新しいインフラストラクチャがオンプレミスにデプロイされます。ウェーブまたは依存関係グループの範囲には変更が必要になることが予想されます。特に、移行日が近づいた場合、以前に不明な依存関係が特定された場合、または新しいサーバーがインベントリに含まれている場合は、変更が必要です。これは、移行自体中に発生することがあります。

スコープの変更は、依存関係グループとウェーブに影響します。変更を処理し、影響を最小限に抑えるには、スコープ制御メカニズムを確立することが重要です。スコープ変更管理メカニズムには、スコープの単一の信頼できるソースの定義が必要です。これは、スコープを管理するためのツールでも、移行プログラムのガバナンスで定義されている .csv ファイル、スプレッドシート、またはデータベースでもかまいません。変更を特定し、影響を分析し、変更に関連する利害関係者に伝えて、関係者がアクションを実行できるようにする必要があります。ウェーブプランは結果として繰り返されます。

詳細なビジネスケース

この段階では、ビジネスケースの範囲を検証して拡張し、トランスフォーメーションプログラムをサポートするためにより詳細なレベルを提供することをお勧めします。すぐに組み立てられる最初の方

向性のあるビジネスケースは、基本的なステップと次のレベルの詳細な計画に投資するのに十分な信頼を提供するように設計されています。

詳細なビジネスケースを作成すると、次の方法でこの計画プロセスがサポートされます。

- 移行およびモダナイズすべき内容、選択するオプション、作業の段階と優先順位付け方法に関する決定事項を示す財務分析を提供する
- 詳細を再検討して、元の方向性のある財務ケースを検証、改善、開発します。
 - インフラストラクチャのコスト削減の可能性
 - 社内の IT 生産性と外部委託運用の効率性
 - プログラムのセットアップ、移行、モダナイゼーションに必要な投資額の見積もり
- 移行がもたらすさらなる価値の推進要因を追跡するプロセスの特定、スケールの見積もり、設定

詳細なビジネスケースでは、以下を確立します。

- 少なくとも移行の第 1 フェーズを実装するためのマニフェストと投資を確保するという目標ベース
- プログラムに対するベースラインの最小財務パフォーマンスの期待値
- さまざまな移行設計と優先順位付けの決定が行われる財務基盤を明確にし、プログラムの過程で状況や人が変化したときに、新しいリーダーシップが情報に基づいた選択を行えるようにしています。
- ワークロードの移行と運用の開始時に最初の使用状況データが利用可能になった後に調査するコスト最適化の増分領域に関するインサイト
- 回復力と俊敏性の向上により、クラウドトランスフォーメーションがビジネスにもたらす価値を見積もる
- 回復力と俊敏性の向上による財務収益を見積もるために使用される、関連する KPIs、メトリクス、前提条件。これにより、プログラムから主なメリットを実現するためのベースラインが設定されます。

ケースに必要なシナリオを決定する

詳細なビジネスケースを構築する場合、通常、ビジネスケースが使用されるさまざまな目的をサポートする複数のシナリオを開発する必要があります。

最小変更シナリオ – 最低限の財務パフォーマンスの期待を評価するには、ステータスクォーへの最低限の変更を想定したシナリオを準備します。このシナリオは、最悪のシナリオとして、移行に投資

する権限を取得する場合に役立つサポートです。このシナリオでは、キャパシティの増加の最小予想レベルと、可用性やレジリエンスなどの他の quality-of-service ニーズに対する最小限の変更をモデル化します。最小の変更は、現在の運用モデルにとって最小のコストと最小のリソース非効率をもたらします。

最も可能性の高いシナリオ — プログラム戦略と優先順位付けの決定を伝えるには、ビジネスが期待していることを反映したシナリオを準備します。このシナリオには、ピーク時の使用率の増加または減少の可能性と、ビジネスから高レベルのサービス品質 (特に可用性と回復力) の需要を満たすためのアップグレードコストを含める必要があります。

その他の特定のシナリオ — ビジネスケースに大きな影響を与える可能性のある仮定をまだ作成する必要がある場合は、仮定が当てはまる場合とそうでない場合の両方のシナリオを開発します。ただし、これらの代替シナリオの数は最小限に抑えることをお勧めします。合計で 3 ~ 4 つを超えるシナリオを作成すると、進行が遅くなり、コストがかかり、混乱し、保守が困難になります。可能な限り、実験を行い、より大きな仮定を削除してください。

インフラストラクチャと移行のコストモデルを検証して改善する

ポートフォリオ分析を完了し、ターゲット AWS サービスの設計とサイジングを準備したら、シナリオ AWS ごとに現在の運用モデル (COM) と将来の運用モデル (FOM) の運用コストの見積もりを絞り込みます。通常、以下の見積もりを絞り込む必要があります。

- ハイパーバイザーホストサーバー、ベアメタルサーバー、ストレージ、ネットワークデバイス、セキュリティアプライアンスのハードウェア更新、インストール、メンテナンスの COM インフラストラクチャコスト。シナリオに必要な容量の実際の料金と割引レベルで計算します。
- スペース、冷却、電力、ラック、無停電電源装置 (UPS)、ケーブル、物理セキュリティシステムなど、COM データセンターとコロケーションされた施設コスト。拡大に合わせてサイズ設定され、容量を満たすように指定されているもの、シナリオの高可用性とディザスタリカバリ (DR) レベル。
- シナリオの接続、帯域幅、スループット、レイテンシーのニーズに関する契約料金を使用して計算された、WAN リンク、コンテンツ配信ネットワーク、仮想プライベートネットワーク (VPNs) のコストを含む COM ネットワークサービスのコスト。
- 既存の契約に基づく COM アプリケーションおよびインフラストラクチャソフトウェアのコスト。シナリオの使用量の増加または削減を提供します。
- 洗練されたサービスアーキテクチャ、インスタンスサイズ、優先料金モデル、予想される使用量、使用状況の変動性に基づいて、必要に応じて技術サポートやマネージドサービスを含む FOM AWS ユーティリティコスト。

- 最終的なアプリケーション設計、アプリケーションを実行するインフラストラクチャの構成、時間の経過に伴う増加、ライセンスの移管可能性ルールに基づく FOM アプリケーションライセンス。
- FOM 移行とモダナイゼーションのコスト見積もり。シナリオのベースライン移行ウェブプランを反映するように改良され、特にリプラットフォーム、再購入、またはリファクタリングされるワークロードごとにコストを提供するように詳述されています。
- アセットの償却コストと契約早期終了コストの見積もり、ベースライン移行ウェブプランの廃止タイミングを反映するように改訂された FOM 廃止コスト、償却を最小限に抑えるためにどのアセットを再利用できるか、どのアセットを切り替えられるかの検証、物理アセットとメディアの廃棄コストなど。
- 移行の並列実行コストは、各移行カットオーバーと既存の各サービス廃止のタイミングを反映するように調整されています。

IT 生産性と IT 運用を改良し、効率価値モデルをサポートする

方向性のあるビジネスケースと同様に、IT 運用とサポートに関する価値モデルを改良し、開発するための主なアプローチは 2 つあります。選択するアプローチは、COM が社内で管理されているか、請負業者または外部委託サービスで管理されているかによって異なります。

社内チームの生産性向上

IT 運用とサポートを社内で管理する場合、ビジネスケースの焦点は次の通りです。

- 対象範囲に含まれる移行と運用自動化による生産性の向上を特定して定量化する
- 社内チームのために解放された時間が、他の通常価値の高いアクティビティに簡単かつ生産的に適用できることを検証し、チームにとって進歩の機会とより大きな報酬、組織にとっての価値を高める

チーム内の各役割の各メンバーがさまざまな通常のアクティビティに費やす時間を評価し、さまざまなアクティビティのワークロードの予想される削減に関するガイダンスを提供します。

次の表は、IT 運用の大部分を消費し、チーム内のさまざまな役割にわたって労力をサポートするタスクのアクティビティ別のワークロード削減の一般的なレベルに関する初期ガイダンスです。この表には、生産性の実現方法の説明が含まれています。

注: リストされているアクティビティは通常、複数の異なる役割のチームメンバーによって実行されるため、各タスクの生産性の節約は、チーム内のすべての役割にわたって評価する必要があります。例えば、インフラストラクチャタワー (コンピューティング、ストレージ、ネットワークなど) ごと

に編成された IT 運用チームでは、各タワーのタワーリードに設備投資の計画と予算編成が共通する場合があります。

運用およびサポートアクティビティ	削減額レベル	生産性ドライバー
インフラストラクチャ設計	中程度	設計が簡素化され、考慮すべきパラメータが少なくなります。
設備投資の計画と予算	高い	OPEX 中心の Elastic サービスは、予算と計画に関する事実上すべての問題を排除します。
購入	高い	AWS アカウントが確立されると、調達が大幅に簡素化されます。
キャパシティプランニング	中～非常に高い	ネットワークとコンピューティングキャパシティ管理のワークロードは通常、すべて排除され、ストレージでは大幅に簡素化されます。
チューニング	非常に高い	インスタンスのサイズはいつでも変更できるため、マネージドサービスにはチューニングは必要ではなく、他のサービスにもほとんど必要ありません。
ハードウェア障害の管理	非常に高い	クラウドでのハードウェアの処理のすべての側面は、によって透過的に処理されます AWS。

運用およびサポートアクティビティ	削減額レベル	生産性ドライバー
サーバーの可用性と通信のモニタリング	高い	ツールのサポートと自動化により、AWS モニタリングと通信は大幅に簡素化されます。
セキュリティ管理	中程度	ワークロードは、AWS セキュリティ機能や AWS クラウドハードウェア、ソフトウェア、ネットワーク、および施設に対する セキュリティ責任 AWS によって大幅に削減されます。
ネットワークとストレージのアップグレード、メンテナンス、パッチ。	非常に高い	クラウドにおけるネットワークとストレージのメンテナンスのすべての側面は、によって透過的に処理されます AWS。
ラッキングとスタッキング - ハードウェアロジスティクス	非常に高い	クラウドでのハードウェア管理のすべての側面は、によって透過的に処理されます AWS。
バックアップ	中程度	バックアップは、AWS ツール、柔軟なストレージシステム、および自動化によって広範囲に簡素化されます。

運用およびサポートアクティビティ	削減額レベル	生産性ドライバー
マネージドサービス (Amazon S3、Amazon RDS AWS Lambda、など AWS Fargate)	非常に高い	マネージドサービスは、によって完全に管理されている環境で実行されるため AWS、メンテナンス、パッチ適用、モニタリング、プロビジョニングの管理アクティビティは必要ありません。
デバイスとサービスのセットアップとコミッショニング	非常に高い	に移行された資産のハードウェアセットアップのアクティビティは通常、VPNs または AWS データセンター AWS Direct Connect への接続を確立するための WAN 接続 デバイスを除いて、削減 AWS されます。
エンドポイント保護とウイルス対策保護	高い	エンドポイント保護とウイルス対策サービスのアプリケーションとメンテナンスは、通常、移行設計の一環として広範囲に自動化されています。
脅威、脆弱性、リスク評価	高い	AWS は、コアプラットフォームに重点を置いたこの要素のサポートを提供し、ガアーキテクチャをセキュア化するために AWS 提供するメカニズムにより、評価が簡素化されます。

運用およびサポートアクティビティ	削減額レベル	生産性ドライバー
データセンターインフラストラクチャプロジェクト管理	高い	インフラストラクチャサービスの拡張、更新、廃止のためのインストール作業のためのプロジェクト管理。インフラストラクチャソフトウェアとサービスの一部の管理は残っていますが、これはオンプレミスインフラストラクチャよりもはるかにシンプルで、ハードウェアアクティビティが排除されます。
データセンター施設管理	中～非常に高い	すべてのサーバー、ストレージデバイス、セキュリティアプライアンス、および関連するラックに起因する施設管理作業は、移行されるすべてのものに対して削除されます。ただし、通常、WAN リンク ネットワーク デバイスやハイブリッド アーキテクチャでオンプレミスに保持されているインフラストラクチャのための施設を提供する作業もあります。

運用およびサポートアクティビティ	削減額レベル	生産性ドライバー
アプリケーションアーキテクチャ、開発、管理、テスト	低	アジャイル開発ツールチェーンと、必要に応じてテスト環境を構築するためのアプリケーションスタックのインスタンス化と破壊の自動化を組み合わせることで、アプリケーション開発のリードタイムを短縮し、多くの手動テストステップを排除します。
アプリケーションソフトウェアのインストールと設定	中程度	アプリケーションスタックの完全なインストールと設定は、などのサービスを使用して簡単に自動化 AWS CloudFormation され、を使用して簡単に設定できるランディングゾーンを使用して簡素化されます AWS Control Tower。
IT サポート	中程度	L1 と L2 のサポートの軽減は、セルフサービスプロビジョニングに Service Catalogue 機能を使用することによって容量とパフォーマンスの問題を削減し、低コストの高可用性アーキテクチャの使用を増やす (停止の削減と自動スケールリングとエッジコンピューティングの設定) ことによって実現されます。

運用およびサポートアクティビティ	削減額レベル	生産性ドライバー
データベース管理	最小-低	これらのアクティビティはほとんど変更されません。通常、のリソースは AWS オンプレミスインフラストラクチャと同じレベルにあります。
インフラストラクチャとセキュリティの要件のキャプチャ、分析、設計	最小限	
ドキュメント	最小限	
アプリケーションとパフォーマンスのモニタリング	最小限	
L3 テクニカルサポート、クエリへの回答、トラブルシューティングと問題解決	最小限	
アプリケーションソフトウェアのインストールと設定	最小限	
アプリケーション L3 のサポート (予算編成と長期キャパシティプランニングを除く)	最小限	

次の表は、ワークロードの削減レベルごとに予想される削減額を示しています。

[レベル]	想定
非常に高い	85% ~ 100%
高い	60% ~ 90%
中程度	30% ~ 70%
低	10% ~ 35%
最小限	0% ~ 10%

これらのメトリクスは、生産性の向上を評価し、詳細なビジネスケースに含めるための出発点となります。実際の生産性の向上は、特定の状況によって異なります。範囲の中間と下端の両方で生産性の節約を計算して、一般的なシナリオと保守的なシナリオを推定すると便利です。

プログラムの進行に伴い、各アクティビティに費やされた時間の実際のデータをロールごとにキャプチャすることが有益です。このデータは、オペレーションの見積りのための改善されたベースを構築し、新しいプロジェクトやサービスの拡張のコストをサポートします。

外部委託された IT 運用とサポートのコスト削減

IT 運用とサポートが主に請負業者に委託または管理されている場合、パートナー主導 (AMS) を含むマネージドサービスソリューションを提供する AWS パートナーに見積りをリクエストすることで、将来の運用モデル [AWS Managed Services](#) (FOM) AWS のコスト配分を準備することができます。また、「方向性のあるビジネスケースの作成」セクションの「[運用コストの最適化の構築](#)」のサブセクションで説明されているように、AWS アカウントマネージャーに連絡して AMS の料金を直接リクエストすることもできます。 [???](#)

詳細なビジネスケースでは、ベンチマークの数値を、改訂された AWS サービス部品表と予想されるサービス消費量、AMS パッケージと必要なオプション、および必要なサービスレベルに基づく引用に置き換えます。コストには、1 回限りの実装コンポーネントと消費ベースの実行レートが含まれます。

残りの IT オペレーション、に移行されないサービスで保持する必要があるサポート AWS、契約違反 (早期終了など) がある場合は 1 回限りのコストを含めます。

レジリエンスバリューモデルを開発する

では AWS、高可用性、ディザスタリカバリ、および耐障害性に優れた幅広いアーキテクチャを構築できます。従量制料金とは、サービスが使用時にのみ課金されることを意味します。これら 2 つの要素を組み合わせることで、耐障害性に優れたコストパフォーマンスが得られます。

さらに、AWS c のお客様はこれを使用してワークロードの耐障害性を向上させています。[IDC 2018 調査](#)では、参加しているお客様が年間 73% の停止回数を達成し、平均復旧時間 (MTTR) を 58% 短縮し、生産性の低下を 94% 短縮した例を示しています。同じ調査では、レジリエンスの向上によって得られる財務上のメリットは、IT インフラストラクチャのコスト削減のメリットよりも 50% 大きいことがわかりました。

さらに、アプリケーションのソフトウェア開発ライフサイクルをモダナイズすることで、さらなる回復力を実現できます。ビジネスの俊敏性を高めるためにテスト自動化を備えた CI/CD パイプライン

が導入されていると、開発サイクルの早い段階でソフトウェアの欠陥が発見され、ソフトウェアのメンテナンスコストが大幅に削減されます。

ビジネスケースでこの価値を評価して含めるには、まずアプリケーションのビジネスオーナーと協力して、移行する各ワークロードの全体的なメリットの機会を把握します。これには、次の項目が含まれる場合があります。

- サービスの中断の数、平均期間、性質：
 - サービスの中断の例としては、停止、パフォーマンスのスローダウン、バッチおよびメンテナンスウィンドウの計画的なオーバーラン、主要な機能のバグ、ピーク時のアクセススロットリングなどがあります。
- e コマースシステムなどの収益を生み出すサービスの中断による収益への影響：
 - 中断時間とトランザクションレートに基づいて、サービスの中断によって完了できないトランザクションの可能性が高い数
 - 影響を受ける各トランザクションの平均値
- サポートエンジニアが本番稼働システムの欠陥を解決するための追加コストと、開発プロセスの早い段階で発見するコストの比較
- 内部ユーザーの生産性への影響と損失時間のコスト

次に、予想される を評価し、サービスの中断による損失時間をより控えめに減らして、回復力を高めることができます。例えば、次の項目を含めることを検討してください。

- 高可用性アーキテクチャを使用した停止回数と MTTR の削減、目標復旧時間 (RTO) と目標復旧時点 (RPO) の改善
- 自動スケーリングなどの機能を使用した、速度低下の軽減、容量スロットリングの排除、バッチ処理のオーバーランの回避
- CI/CD パイプラインの実装とインフラストラクチャのスピンアップとスピンドアウンの自動回帰テストにより、本番環境でのみ検出されるアプリケーションのバグの数を削減し、コストを最小限に抑えます。

これらをまとめると、アプリケーションのポートフォリオを移行してモダナイズし、ケースの年ごとに予想され、より保守的なビジネス価値の数値を計算できます。メリットは移行スケジュールに合わせて増加し、貢献しているアプリケーションの使用の増加の期待に合わせてボリュームをスケールインする必要があります。

ビジネスの俊敏性価値モデルを開発する

ビジネスの俊敏性は、AWS お客様がに移行する主な理由です AWS。 [IDC 2018 年顧客調査](#)によると、ビジネス俊敏性のメリットは、測定される総メリットの 47% を占め、インフラストラクチャのコスト削減によるメリットの 5 倍以上を占めています。 AWS

あらゆるトランスフォーメーションから生じるすべてのビジネス俊敏性のメリットを正確に予測することは困難です。ただし、多数のユーザーをサポートするアプリケーションやビジネス上の差別化要因となるアプリケーションに焦点を当てることで、この利点の重要な部分をモデル化し、ベースラインの詳細なビジネスケースに含めることができます。

移行が進むにつれて、より多くのメリットが定量化可能になったら、ビジネスの俊敏性バリューモデルを段階的に改良し、拡張します。これにより、ビジネスケースが関連付けられ、プログラムを主導する主要な意思決定サポートツールとして使用できます。

ビジネスの俊敏性バリューモデルを構築するには、次のガイダンスを使用します。

- 次のような、ビジネスパフォーマンスの最大化を促進する機会があるワークロードを選択します。
 - 収益を生み出すワークロード
 - 効率の向上とビジネスからのコストの排除につながるビジネス運用ワークロード
 - 大規模なユーザーベースをサポートするビジネス生産性向上ツール
- 収益と効率を生み出すワークロードについては、次の操作を行います。
 - アプリケーションのメジャーアップグレードとマイナーアップグレードが推進されると予想される収益の増加または運用効率を現実的かつ控えめに評価します。
 - AWS アプリケーション開発速度の向上とインフラストラクチャのデプロイ時間の短縮により、1 年あたりのメジャーリリースとマイナーリリースの増加を推定します。これに関する一部のベースラインメトリクスは、IDC レポートに記載されています。
 - 現実的で保守的なメリットの期待値を計算します。ビジネスケースの期間にわたってマッピングし、それぞれのワークロードが移行されてからしばらくして、効率を最大に引き上げることができます。
- ビジネス生産性向上ツールの場合は、次の操作を行います。
 - メジャーアプリケーションとマイナーアプリケーションのアップグレードが推進すると予想される時間の節約について、現実的で保守的な評価を行います。
 - 影響を受けるユーザーベース全体の人の時間と労力の平均コストを見積もります。
 - メジャーリリース頻度とマイナーリリース頻度の増加の数値を使用し、ビジネスケースの期間中のメリットを計算します。

デベロッパーの生産性が向上し、起動までの時間が短縮されるため、追加のリソースは必要ないので、割引されたキャッシュフロー、NPV、ROI、MIRR、およびペイバック計算に含めるために、各ワークロードの純利益ラインをビジネスケースのキャッシュフローモデルに追加します。

継続的な評価と改善

この評価段階では、次の2つの側面に焦点を当てます。

- アプリケーションの種類ごとに、継続的に詳細なアプリケーション評価を行います
- ポートフォリオの継続的な進化と改善

最初の側面は、継続的な詳細なアプリケーション評価であり、アーキテクチャとテクノロジーレベルに至るまでの詳細な発見と分析に重点を置いて、特定の波における各アプリケーション、AWS提案された設計、および移行戦略を完全に理解します。この移行準備状況の評価は、特定の移行ウェーブを開始するための前提条件です。

2つ目の側面は、ポートフォリオの継続的な進化と改善であり、ポートフォリオ管理と、ビジネスケースの進化と追跡を含め、時間の経過とともにアプリケーションをどのように改善する予定であるかに重点を置いています。

この段階の主な移行結果は次のとおりです。

- 各ウェーブの検証済み移行範囲
- 特定の移行段階におけるアプリケーションのターゲットアーキテクチャと移行戦略を文書化
- 移行パターンとツールの特定と検証
- 各ウェーブにおける要件 (セキュリティ、AWSインフラストラクチャ、運用) と移行のカットオーバーに関する考慮事項を文書化

この段階の主な最適化結果は次のとおりです。

- ポートフォリオ合理化モデルとビジネス成果
- 提案されているアーキテクチャとテクノロジーの変更、およびそれらに期待されるメリット
- プラットフォーム要件 (セキュリティ、AWSインフラストラクチャ、運用)
- 実装計画

継続的評価データ要件の理解

アプリケーションポートフォリオの継続的な評価と改善のためのデータ要件は、前のセクションのデータ要件を組み合わせたものです。ポートフォリオの移行とその進化を継続的に管理するには、以下のセクションを参照してデータ要件を理解してください。

- ウェーブアセスメントとアプリケーションの最適化には、「[優先アプリケーションアセスメント](#)」セクションのデータ要件を使用してください。
- 継続的なポートフォリオ管理には、「[ポートフォリオ分析と移行計画](#)」セクションのデータ要件を使用してください。
- ウェーブプランの定義については、[ウェーブプランニングのセクション](#)を参照してください。

詳細なウェーブアセスメント

移行の波を先取りし、移行の鍵となるアプリケーションの詳細な評価には、[優先順位付けされたアプリケーション評価段階と同じ要件と推奨事項があります](#)。目標は、特定の波におけるアプリケーションの現状を詳細に理解し、運用面、ツール、特定の移行パターンなど、future アーキテクチャ設計と移行戦略を策定することです。

[優先順位付けされたアプリケーション評価](#)を、特定のウェーブ内のアプリケーショングループに適用します。移行計画の各段階の前にこのプロセスを繰り返してください。重要なのは、詳細な評価から波が始まるまでの間に十分な時間を確保することです。必要な時間は、Wave 要件を実装して移行を実行するプラットフォームチームと移行チームの要件によって決まります。それらのチームと協力して、詳細なウェーブアセスメントとウェーブのスケジュールを設定します。生産ラインをエミュレートする工場のようなモデルを実装することをお勧めします。

最適化と最新化のための評価

すでに移行されているワークロードの最適化とモダナイゼーションの評価プロセスは、AWS移行先のワークロードの評価に似ていますAWS。主な変更点は、評価を実施するためのデータのソースです。にはAWS、out-of-the-box 実行中のアプリケーションに関する詳細情報を取得するために使用できるツールとサービスがいくつかありますAWS。

アプリケーションを最適化および最新化する内容と方法は、お客様固有の推進要因と状況によって異なります。最適化は、現在のアーキテクチャとテクノロジーに変更を適用して、コストを削減し、パフォーマンス要件を調整し、学んだ教訓を取り入れることに重点を置いています。モダナイゼーショ

ンは、サーバーレスモデルやマイクロサービスアーキテクチャの採用など、アプリケーションを次のレベルに引き上げることに重点を置いています。

[優先順位付けされたアプリケーション評価のガイドラインに従ってください](#)。最適化と最新化の取り組みをさらに支援するには、次のリソースを参照してください。

- [AWSコスト最適化](#)は、ITの最適化とITコストの削減に関する情報を提供します。
- [AWS Compute Optimizer](#) は、機械学習を使用して過去の使用率メトリクスを分析することにより、AWSワークロードのリソースを推奨してコストを削減し、パフォーマンスを向上させます。
- [AWSコストと容量の最適化サービスとツール](#)は、コンピューティングリソースの管理に役立つため、コンピューティングコストの管理にかかる時間を短縮して、より多くの時間を構築に費やすことができます。
- [Amazon S3 Storage Lens](#) を使用すれば、オブジェクトストレージの使用状況とアクティビティについての傾向を、組織全体で確認できます。コスト効率を高め、データ保護のベストプラクティスを適用するための実用的な推奨事項を作成します。
- [Database Freedom](#)は、AWSデータベースおよび分析サービスへの移行を容易にします。
- [Amazon CodeGuru](#) は、コードの品質を向上させ、アプリケーションの最もコストのかかるコード行を特定するためのインテリジェントな推奨事項を提供する開発者ツールです。
- [AWSハイブリッドクラウドサービス](#)は、クラウドからオンプレミス、エッジまで、AWS必要な場所で一貫したエクスペリエンスを提供します。

その他のリソース

- [コストの最適化とイノベーション:アプリケーションモダナイゼーション入門](#) (ブログ記事)
- [サーバーレス Web アプリケーションのコストの最適化](#) (ブログ記事)
- [ウィンドウズオンAWS](#) (ブログ)
- [最新のアプリケーション](#)
- [アプリケーションのモダナイゼーション](#) (AWSre: Invent 2020)
- [AWSマイクロサービスガイド](#)

ウェーブプランの繰り返し

移行プログラムが進み、移行する波が増えるにつれて、教訓と変化するビジネス上の優先事項に基づいて移行ウェーブプランを進化させることが重要です。特に、長期にわたる移行プログラムでは、ビ

ビジネスの推進要因と組織の変化を再評価し、移行の波の計画が引き続き有効であることを確認することが重要です。

同様に、移行から学んだ教訓は、ウェーブプランの構成と各ウェーブの範囲に影響を与えます。何が起きているかを把握できなくなることを避けるため、[ウェーブプランを最新の状態に保ってください](#)。計画には、提供されている内容を反映して追跡し、移行範囲の変更を管理および評価する必要があります。

ビジネスケースの進化と追跡

移行が進むにつれ、特に長期にわたるプログラムの場合は、ビジネス上の圧力によって移行と近代化の優先順位が定期的に再検討されることは避けられません。

新しい情報が入手可能になったときにビジネスケースを進化させること、および詳細なビジネスケースに記録されている期待と照らし合わせて実際の商業パフォーマンスを追跡することの両方をお勧めします。これには以下が含まれます。

- 組織の新たな構造変化がビジネスの優先事項に影響を及ぼし、それに伴うIT戦略とアプリケーションポートフォリオにも影響を与える
- 移行と最新化が目指すアプリケーションポートフォリオの一部の商業的重要性の高まり、またはアプリケーションポートフォリオへの変更
- 移行後のアプリケーションに関する実際のリソース使用率データの可用性（サイズの調整、段階的なモダナイゼーションのケースの定量化と確認など）
- IT運用とサポート活動に費やされた労力に関するデータの入手可能性、および可能な運用改善と自動化の分析
- ソフトウェア開発とメンテナンスのサイクルタイムの変化、開発段階ごとのソフトウェア不具合、サービスの可用性情報を測定するデータの可用性、さらに改善の余地がある領域の根本原因分析

ビジネスケースと照らし合わせてパフォーマンスを追跡することで、移行開始後に簡単に評価および定量化できるさらなる改善点を含めるようにケースを進化させることができます。プログラムガバナンス組織は、変化するビジネス圧力に対応し、管理可能で許容可能なレベルのリスクで最大の価値を生み出す方向に変革を導くための準備がはるかに整っています。

これは、ケース内のIT生産性、耐障害性、およびビジネスアジリティのメリットにとって特に重要です。これらは通常、前もって評価するのが最も大きい要因であり、またより困難な要因でもあります。これらのドライバーのパフォーマンスを追跡することで、チームはメリットの実現を妨げている

問題を深く掘り下げて解決することができます。または、ビジネスケースを調整して、最も継続的な財務パフォーマンスの最適化を実現するイニシアチブに優先順位を付けることもできます。

リソース

AWS参照

- [Amazon Builders' Library](#)
- [アプリケーションのモダナイゼーション](#) (AWSre: Invent 2020)
- [アプリケーションポートフォリオ評価戦略](#)
- [AWSアーキテクチャーセンター](#)
- [AWS Compute Optimizer](#)
- [AWSコストと容量の最適化サービスとツール](#)
- [AWSコスト最適化](#)
- [コストの最適化とイノベーション:アプリケーションモダナイゼーション入門](#) (ブログ記事)
- [AWS ドキュメント](#)
- [リソースセンターのご利用開始にあたって](#)
- [AWS Marketplace](#)
- [AWS Managed Services パートナー](#)
- [AWSマイクロサービスガイド](#)
- [AWS 移行コンピテンシーパートナー](#)
- [最新のアプリケーション](#)
- [サーバーレス Web アプリケーションのコストの最適化](#) (ブログ記事)
- [AWS Prescriptive Guidance](#)
- [AWSプロフェッショナルサービス](#)
- [AWSソリューションライブラリ](#)
- [ウィンドウズオンAWS](#) (ブログ)

AWS のサービス

- [AWSアプリ 2 コンテナ](#)
- [AWS Application Migration Service](#)
- [アマゾン CodeGuru](#)
- [AWS Control Tower](#)

- [データベースの自由](#)
- [AWS Database Migration Service](#)
- [AWS DataSync](#)
- [AWS Direct Connect](#)
- [Amazon ECS](#)
- [Amazon EKS](#)
- [AWS Fargate](#)
- [AWS Managed Services](#)
- [移行評価者](#)
- [AWS Migration Hub戦略に関する推奨事項](#)
- [AWSランディングゾーン](#)
- [AWS 料金計算機](#)
- [AWS Schema Conversion Tool](#)
- [Amazon S3 Storage Lens](#)
- [AWS Snowball](#)
- [AWS Snowcone](#)
- [AWS VPN](#)

その他のリソース

- [Amazon Web Services によるビジネス価値創出のためのビジネスおよび組織変革の促進](#)
- [IDC 2018 調査](#)

ドキュメント履歴

次の表に、この戦略の重要な変更点を示します。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
更新	ポートフォリオの発見と初期計画セクションの発見促進と初期計画の名前を変更し、決定木図を更新しました。	2024 年 5 月 20 日
二	初版発行	2021 年 11 月 12 日

AWS 規範ガイド用語集

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

数字

7 Rs

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

- リファクタリング/アーキテクチャの再設計 — クラウドネイティブ特徴を最大限に活用して、俊敏性、パフォーマンス、スケーラビリティを向上させ、アプリケーションを移動させ、アーキテクチャを変更します。これには、通常、オペレーティングシステムとデータベースの移植が含まれます。例: オンプレミスの Oracle データベースを Amazon Aurora PostgreSQL 互換エディションに移行します。
- 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ある程度の最適化を導入します。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: 顧客関係管理 (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 インスタンスで 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。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: を移行する Microsoft Hyper-V へのアプリケーション 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

[原子性、一貫性、分離、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。柔軟性がありますが、[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SQLします。集計関数の例には、SUMおよびMAXが含まれます。

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。

人工知能オペレーション (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。移行戦略で AWS がどのように AIOps 使用されるかの詳細については、「[オペレーション統合ガイド](#)」を参照してください。

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、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コントロール (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[ABAC AWS](#)「」の「」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

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安価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かつ効果的な計画を策定 AWS するのに役立ちます。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まとめています。ビジネス、人材、ガバナンスの観点では、ビジネススキルとプロセスに重点を置き、プラットフォーム、セキュリティ、オペレーションの視点は技術的なスキルとプロセスに焦点を当てています。例えば、人材の観点では、人事 (HR)、人材派遣機能、および人材管理を扱うステークホルダーを対象としています。この観点から、AWS CAFは、クラウド導入を成功させるための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に関する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

個人または組織に混乱または害を与えることを目的とした[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 で動作グラフを使用して、失敗したログオン試行、疑わしいAPI呼び出し、および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。 [「endianness」](#) 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

2 つの別々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は 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は他の環境 (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インタラクション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をインデックス化するウェブクロウラーなど、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、個人や組織に混乱や害を与え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呼ばれる 1 つ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、[「ブランチについて \(GitHub ドキュメント\)」](#)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アクセスするための簡単な手段を提供します。詳細については、「Well-Architected」ガイダンスの AWS [「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。

事業継続計画 (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 Cloud Adoption Framework](#) を参照してください。

Canary のデプロイ

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。自信が持てば、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。

データキャプチャの変更 (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。は、同期を維持するために、ターゲットシステムの変更を監査したりレプリケートしたりするなど、CDCさまざまな目的で使用できます。

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 ワークロードに負荷をかけ、そのレスポンスを評価する実験を実行できます。

CI/CD

[継続的統合と継続的配信](#) を参照してください。

分類

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

クライアント側の暗号化

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

Cloud Center of Excellence (CCoE)

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

クラウドコンピューティング

リモートデータストレージと IoT デバイス管理に通常使用されるクラウドテクノロジー。クラウドコンピューティングは、一般的に[エッジコンピューティング](#)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
- 基盤 — クラウド導入を拡大するための基盤投資 (ランディングゾーンの作成、 の定義CCoE、オペレーションモデルの確立など)
- 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

[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以下が含まれます。GitHub または 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、デジタル画像や動画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、はオンプレミスのカメラネットワークに CV を追加するデバイス AWS Panorama を提供し、Amazon SageMaker は CV 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、設定は想定された状態から変更されます。ワークロードが非準拠になる可能性があり、通常、段階的かつ意図的ではありません。

設定管理データベース (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、移行の CMDB ポートフォリオ検出および分析段階でのデータを使用します。

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、または組織全体に単一のエンティティとしてデプロイできます。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 is commonly described as a pipeline. CI/CD は、プロセスの自動化、生産性の向上、コード品質の向上、および迅速な提供に役立ちます。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

一元的な管理とガバナンスで分散された分散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 のみが期待されたネットワークから信頼されたリソースにアクセスしていることを確実にします。詳細については、「[「でデータ境界を構築する 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、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。

defense-in-depth

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 に採用する場合 AWS、リソースの保護に役立つように、AWS Organizations 構造のさまざまなレイヤーに複数のコントロールを追加します。例えば、アプローチでは defense-in-depth、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。

デプロイメント

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、リーンな製造プラクティス用に最初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ます。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番号です。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、「[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。AWS

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)。ストラングルラーの fig パターンでドメイン駆動型設計を使用する方法については、「[従来の Microsoft のモダナイズ](#)」を参照してくださいASP。NET (ASMX) コンテナと Amazon API Gateway を使用してウェブサービスを段階的に更新する「」。

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。例えば、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化を検出](#)したりできます。

DVSM

[「開発バリュー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。

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、レスポンスタイムを向上させることができます。

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) 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、インターフェイスエンドポイントを作成することでVPC、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、アイデンティティとアクセスの管理、検出コントロール、インフラストラクチャのセキュリティ、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、サマリー統計を計算し、データ視覚化を作成することで実行されます。

F

ファクトテーブル

[星スキーマの中央テーブル](#)。ビジネスオペレーションに関する定量的なデータを保存します。通常、ファクトテーブルには 2 つのタイプの列が含まれます。つまり、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です。

フェイルファースト

開発ライフサイクルを短縮するために頻繁で増分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。

障害分離境界

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

機能ブランチ

[ブランチ](#) を参照してください。

特徴量

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

特徴量重要度

モデルの予測に対する特徴量の重要性。これは通常、Shapley Additive Explanations (SHAP) や統合勾配など、さまざまな手法で計算できる数値スコアとして表されます。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。

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コントロール (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。

G

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)

Amazon では CloudFront、特定の国のユーザーがコンテンツディストリビューションにアクセスできないようにするオプションです。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分散の制限](#)」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](#)はモダンで望ましいアプローチです。

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

組織全体のリソース、ポリシー、コンプライアンスを管理するのに役立つ大まかなルール (OUs)。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、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。](#)

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。

同種データベースの移行

ソースデータベースを、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に移行する (Microsoft SQL Server から Amazon RDS for SQL Server など)。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。緊急性のため、通常、ホットフィックスは一般的な DevOps リリースワークフローの外部で行われます。

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 日間の平均使用量 CPU とメモリ使用量が 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、本番稼働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、予測性に優れています。詳細については、AWS 「Well-Architected Framework」の [「イミュータブルインフラストラクチャのベストプラクティスを使用したデプロイ」](#) を参照してください。

インバウンド (インGRESS) VPC

AWS マルチアカウントアーキテクチャでは、VPC が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、ルーティングします。[AWS セキュリティリファレンスアーキテクチャ](#) 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インターフェイス VPCs を保護するために、インバウンド、アウトバウンド、および検査でネットワークアカウントを設定することをお勧めします。

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

接続、リアルタイムデータ、オートメーション、分析、AI/ML の進歩を通じて、製造プロセスのモダナイゼーションを指すために 2016 年に [Klaus Schwab](#) によって導入された用語。

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。

産業用モノのインターネット (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、[「産業用モノのインターネット \(IIoT\) 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。

検査 VPC

AWS マルチアカウントアーキテクチャでは、VPCs (同一または異なる 内の 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VPCを管理する一元化されたです。[AWS セキュリティリファレンスアーキテクチャ](#)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インターフェイスVPCsを保護するために、インバウンド、アウトバウンド、および検査でネットワークアカウントを設定することをお勧めします。

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 の基盤を提供しますITSM。

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

ラベルベースのアクセスコントロール (LBAC)

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ている必須のアクセスコントロール (MAC) の実装。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「[最小権限の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。[「endianness」](#) も参照してください。

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を混乱させたり、機密情報を漏洩したり、不正アクセスを受けたりす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、プラットフォームを で AWS 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。これらは抽象サービス と呼ばれます。

製造実行システム (MES)

原材料を作業現場の最終製品に変換する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。

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、ツールの採用を推進し、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

の組織の一部である管理アカウント AWS アカウント を除くすべての 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](#) デバイス向けの、machine-to-machine [パブリッシュ/サブスクライブ](#) パターンに基づく軽量 (M2M) 通信プロトコル。

マイクロサービス

明確に定義された上で通信APIsし、通常、小規模な自己完結型チームが所有する、小規模で独立したサービス。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APIs。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

びスケールリングできます。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。

移行促進プログラム (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、従来の移行を系統的な方法で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して高速化するための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、オペレーション、ビジネスアナリストと所有者、移行エンジニア、デベロッパー、スプリントに携わる 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には、ターゲットサブネット、セキュリティグループ、AWS アカウントが含まれます。

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 EC2を使用して Amazon への移行をリホストします。

移行ポートフォリオ評価 (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 (サーバーの適正サイズ、料金、TCO比較、移行コス

ト分析)と移行計画(アプリケーションデータ分析とデータ収集、アプリケーショングループ化、移行の優先順位付け、ウェーブプランニング)を提供します。[MPA ツール](#) (ロゲインが必要)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)

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 AWS CAF。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 移行戦略の最初のフェーズ](#) です。

移行戦略

ワークロードを に移行するために使用されるアプローチ AWS クラウド。詳細については、この用語集の「[7 Rs エントリ](#)」および「[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「[アプリケーションのモダナイズ戦略 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

本稼働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 Framework AWS 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](#)をベストプラクティスとして使用することを推奨しています。

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。

Open Process Communications - 統合アーキテクチャ (OPC-UA)

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

運用レベルの契約 (OLA)

サービスレベルの契約 (SLA) をサポートするために、IT グループが相互に提供することを約束する機能的な IT グループを明確にする契約SLA。

運用準備状況のレビュー (ORR)

インシデントや潜在的な障害の範囲を理解、評価、防止、または軽減するの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)

物理環境と連携して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](#) 変換の主要な焦点です。

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、ドキュメントの「[組織の証跡の作成](#)」を参照してください。CloudTrail

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、変化の採用を加速し、移行に伴う問題に対処し、文化的および組織的な変化を推進することで、組織が新しいシステムや戦略の準備と移行を支援します。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から、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)

では CloudFront、Amazon Simple Storage Service (Amazon S3) コンテンツを保護するためのアクセスを制限するための拡張オプションです。OAC は、すべての S3 バケット AWS リージョン、AWS KMS (SSE-KMS) によるサーバー側の暗号化、および 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)

では CloudFront、Amazon S3 コンテンツを保護するためにアクセスを制限するオプションがあります。を使用する場合 OAI、は Amazon S3 が認証できるプリンシパル CloudFront を作成します。認証されたプリンシパルは、特定の CloudFront ディストリビューションを介してのみ S3 バケット内のコンテンツにアクセスできます。も参照してください。これにより [OAC](#)、より詳細で拡張されたアクセスコントロールが提供されます。

ORR

[「運用準備状況の確認」](#)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 (出力) 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VPCを処理するです。[AWS セキュリティリファレンスアーキテクチャ](#)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インターフェイスVPCsを保護するために、インバウンド、アウトバウンド、および検査でネットワークアカウントを設定することをお勧めします。

P

アクセス許可の境界

ユーザーまたはロールが持つことができる最大アクセス許可を設定するために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。例としてPIIは、名前、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

アクセス許可の定義 ([アイデンティティベースのポリシー](#) を参照)、アクセス条件の指定 ([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の定義 AWS Organizations ([サービスコントロールポリシー](#) を参照) が可能なオブジェクト。

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。一般的には false 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 ロール AWS アカウント、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「ロールの用語と概念」を参照してください。https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html#id_roles_terms-and-concepts

プライバシーバイデザイン

エンジニアリングプロセス全体を通してプライバシーを考慮に入れた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

Amazon Route 53 が 1 つ以上の 内のドメインとそのサブドメインのDNSクエリに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VPCs。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

非準拠の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、プロビジョニングされる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「[の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、減少と削除に至るまで、ライフサイクル全体にわたる製品のデータとプロセスの管理。

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、信頼性が高く適応性の高いコンピュータです。

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。

publish/subscribe (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で、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](#)、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ンネルにイベントメッセージを公開できます。システムは、パブリッシュ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す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、情報 \(RACI\)](#) 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。

再設計

[7 Rs](#) を参照してください。

復旧ポイントの目的 (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。耐障害性、安定性、耐障害性を提供するために、それぞれ AWS リージョン が分離され、他のものとは独立しています。詳細については、[AWS リージョン 「を使用できるアカウントを指定する」](#) 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、で障害耐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。

責任、説明責任、相談、情報 (RACI) 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、行列はRASCI行列と呼ばれ、除外すると行RACI列と呼ばれます。

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、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。

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。

行と列のアクセスコントロール (RCAC)

アクセスルールが定義されている基本的で柔軟なSQL式の使用。RCAC は、行のアクセス許可と列マスクで構成されます。

RPO

[「復旧ポイントの目的」](#)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 (IdPs) が使用す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) AWS Management Console が有効になるため、ユーザーは にログインしたり、AWS API組織内のすべてのユーザーIAMに対して でユーザーを作成したりすることなく 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

シヨンの詳細については、IAMドキュメントの [「2.0 SAML 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

では AWS Secrets Manager、暗号化された形式で保存するパスワードやユーザー認証情報などの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。

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 ???](#)、[およびプロアクティブ](#) 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。

セキュリティ情報とイベント管理 (SIEM) システム

セキュリティ情報管理 (SIM) システムとセキュリティイベント管理 (SEM) 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、その他の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、アラートを生成します。

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正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、認証情報のローテーションなどがあります。

サーバー側の暗号化

送信先で、それ AWS のサービスを受信する 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はガードレールを定義するか、管理者がユーザーまたはロールに委任できるアクションの制限を設定します。を許可リストまたは拒否リストSCPとしてを使用して、許可または禁止されるサービスまたはアクションを指定できます。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URLの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。

サービスレベル契約 (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[レベルインジケータ](#)によって測定される、サービスの正常性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について共有する責任を説明するモデル。クラウドのセキュリティAWSはクラウドのセキュリティに責任があり、クラウドのセキュリティはユーザーの責任です。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)

システムを混乱させる可能性のある、アプリケーションの単一の重要なコンポーネントの障害。

SLA

[「サービスレベル契約」](#)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。

split-and-seed モデル

モダナイゼーションプロジェクトのスケール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[「アプリケーションのモダナイズに対する段階的なアプローチ 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ため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。このパターンを適用する方法の例については、「[従来の Microsoft のモダナイズ](#)」を参照してください。ASP.NET (ASMX) コンテナと Amazon API Gateway を使用してウェブサービスを段階的に更新する「」を参照してください。

サブネット

内の IP アドレスの範囲VPC。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。

監視コントロールとデータ取得 (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です。

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

ユーザーインタラクション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。

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

VPCs とオンプレミスの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トランジットハブ。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

組織内でタスクを実行するために指定したサービスに、ユーザーに代わってそのアカウント AWS Organizations で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と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つのピザを食べることができる小さな 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VPCsできる 2 つの間の接続。詳細については、Amazon VPCドキュメントの[VPC「ピアリングとは」](#)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、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

「書き込み」を1回参照し、多くのを読み取ります。

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。

1回書き込み、多数読み取り (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。認定ユーザーは、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。

ゾンビアプリケーション

平均使用量CPUとメモリ使用量が 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。