



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

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

Table of Contents

序章	1
概要:	1
Discovery アクセラレーションと初期計画	4
このステージの主な結果	4
初期評価データ要件について	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	26
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
アタッチメント	31
方向性のあるビジネスケースの作成	31
方向性のあるビジネスケースの範囲の修正	32
フォーカス値ドライバー	33
データのニーズ	34
インフラストラクチャの TCO 比較の構築	34
運用コスト最適化の構築	35
完全な方向性のビジネスケースへの拡張	37
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	39
優先順位付けされたアプリケーション評価	49
詳細な評価データ要件について	49
詳細なアプリケーション評価	58
General	60
アーキテクチャ	60
オペレーション	61
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	62
回復性	62
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	63
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	66
アーキテクチャ上の決定事項	69
ソフトウェアライフサイクル環境	69
Tagging	69
移行戦略	69
移行パターンとツール	69
サービス管理とオペレーション	70
カットオーバーに関する考慮事項	71
リスク、前提、問題、依存関係	71
実行コストの見積もり	71
.....	72
完全な評価データ要件について	72
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	83
優先順位付け基準の反復	85
6 Rs 移行戦略の選択を繰り返す	87
ウェーブプランニング	88
ウェーブプランの作成	90
変更の管理	92
詳細なビジネスケース	92
ケースに必要なシナリオを決定する	93
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絞り込む	94
IT 生産性と IT 運用を改善し、効率バリューモデルをサポートする	95
レジリエンスバリューモデルを開発する	102
ビジネスの俊敏性バリューモデルを開発する	103
継続的な評価と改善	105
継続的な評価データ要件を理解する	106
詳細なウェーブ評価	106
最適化とモダナイゼーションの評価	106
ウェーブプランの反復	107
ビジネスケースの進化と追跡	108

リソース	109
ドキュメント履歴	111
用語集	112
#	112
A	113
B	115
C	117
D	120
E	124
F	127
G	128
H	129
I	131
L	133
M	134
O	138
P	141
Q	144
R	144
S	147
T	151
U	152
V	153
W	153
Z	154
.....	clv

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

German Goncalves and Mark Berner、Amazon Web Services (AWS)

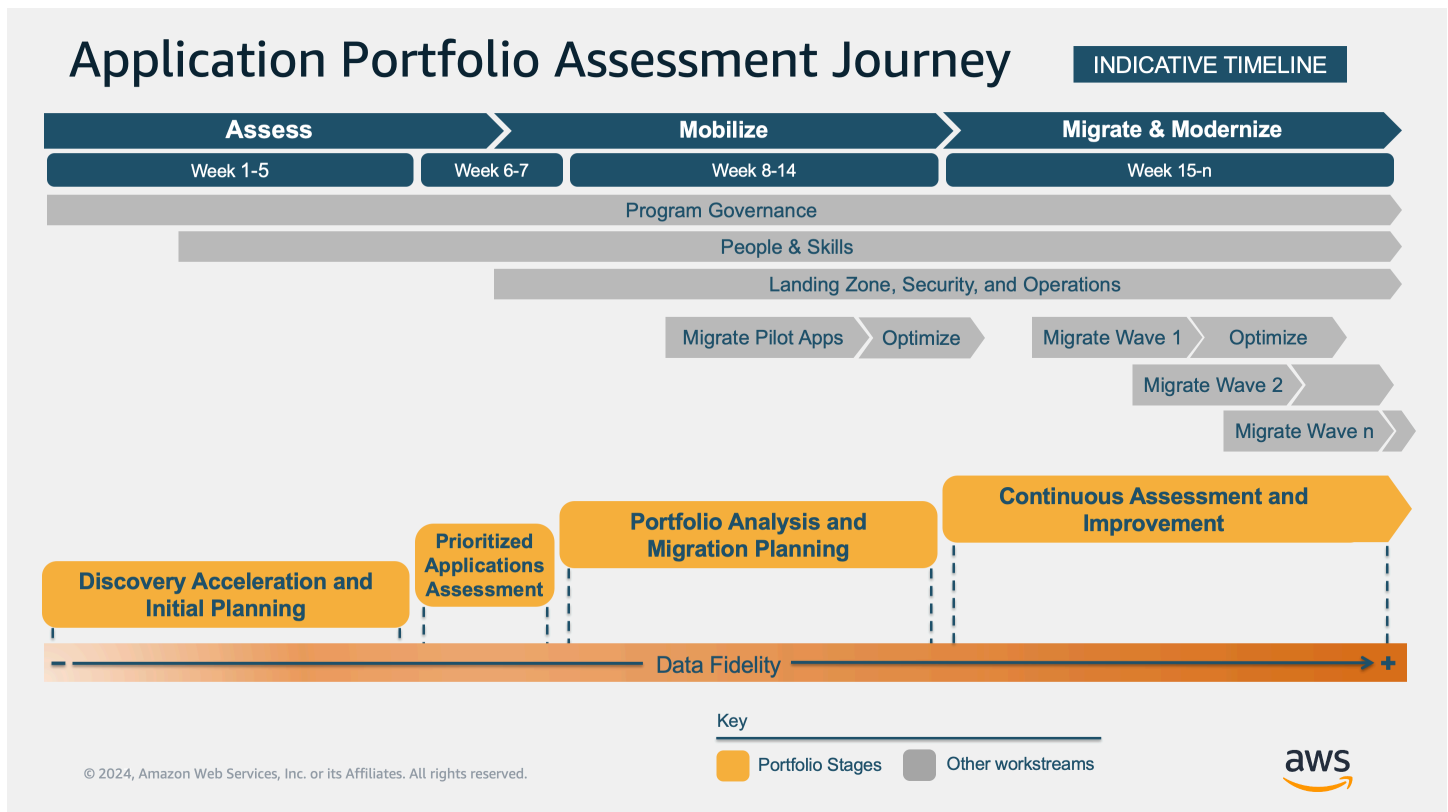
2024 年 5 月 ([ドキュメント履歴](#))

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、アプリケーションポートフォリオと関連するインフラストラクチャ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:

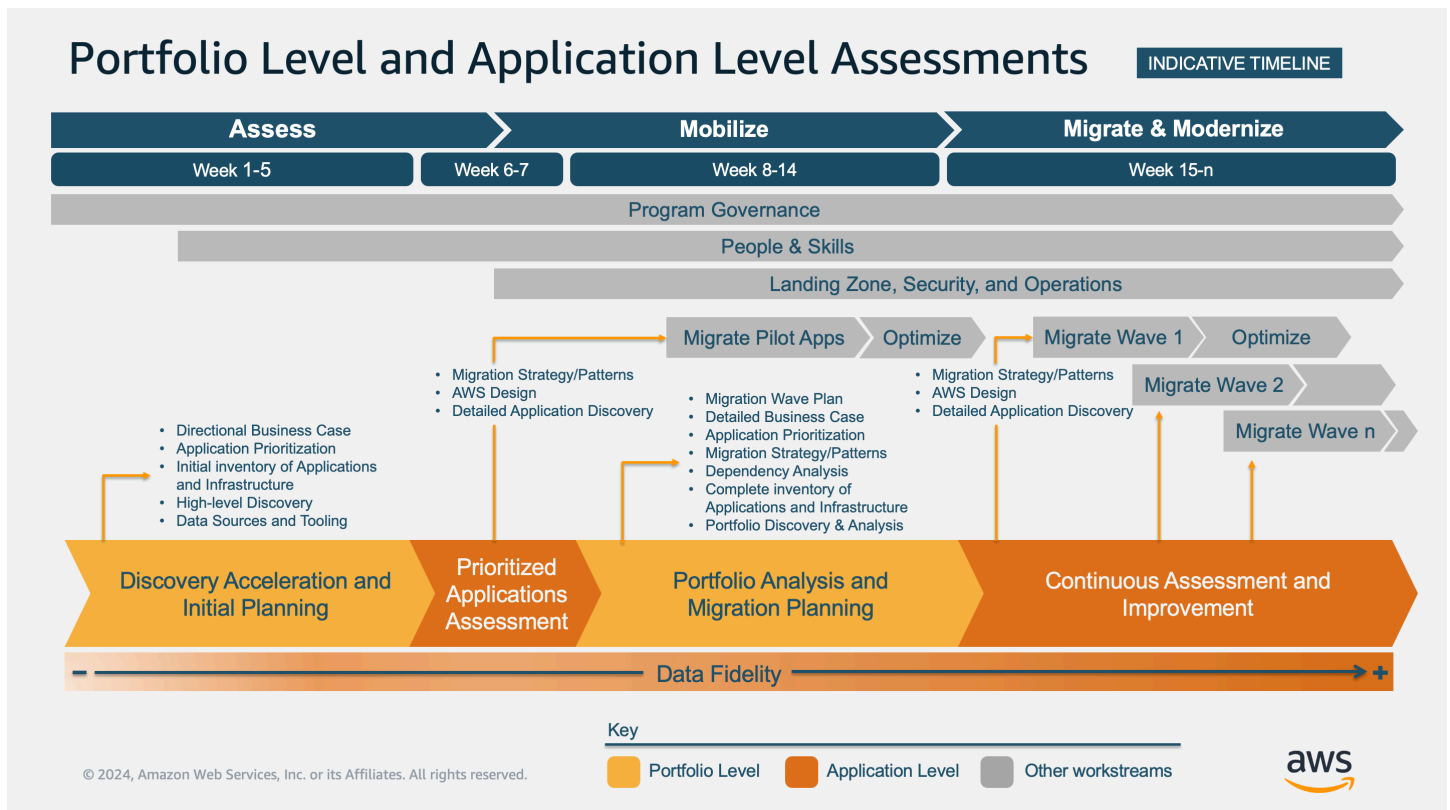
長時間実行されるクラウド移行プログラムでは、プログラムガバナンス、ランディングゾーン (セキュリティコントロールを備えた運用ターゲット環境)、移行、アプリケーションポートフォリオなど、複数のワークストリームを調整する必要があります。これらのワークストリームの名前は、移行プログラムの整理方法によって異なる場合があります。ワークストリームとして、アプリケーションポートフォリオ評価は、これらのプログラムのライフサイクル全体にわたる基本的なアクティビティを表します。評価を通じて得られるポートフォリオの理解は、継続的なアプリケーションポートフォリオ評価から得られるデータと分析に依存する他のワークストリー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

次の図は、ポートフォリオ評価のステージが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 AWS 対応しているかを示しています。ポートフォリオ検出と初期計画段階は、通常は最初の 5 週間に評価段階で開始されます。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、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、任意の移行プログラム構造に組み込むことができます。



- Discovery 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、短期的に移動できるアプリケーションの特定を通じて、time-to-valueします。アプリケーションをすばやく移行することで、チームは移行の経験を持ち、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ー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生成することで、大規模な移行をサポートします。このステージには、移行ウェーブプランの反復と、最適化とモダナイゼーションのための移行されたワークロードのさらなる分析が含まれます。

次の図は、評価の各ステージ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ています。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。



Discovery アクセラレーションと初期計画

このポートフォリオ評価の最初の段階では、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の高レベルの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 (3 つのアプリケーションなど)。
- 次のステップを定義します。

初期評価データ要件について

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、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、サポートチームのメンバー、およびアプリケーション所有者です。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？たとえば、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) を更新するアクティブなツールと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間の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タログ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。
- 環境では、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他のどのようなツールを利用できますか？ 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、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、忠実度レベルまたは信頼レベルをそれぞれのレベルに割り当てます。ツールなどのアクティブなプログラムソースから最近 (30 日以内) 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、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 日を超えてい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。前提とそれに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、組織の知識とは、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
の専門知識	低 - 正確なデータの最大 25%、75% の想定値、またはデータが	低	重要アプリケーションに焦点を当てた希少

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
	150 日以上経過している。		
Knowledge base	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	手動で管理され、詳細レベルに一貫性がない
CMDB	中 - 正確なデータの約 50%、約 50% の想定値、またはデータが 90~120 日経過しています。	中	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	仮想化された資産の 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。

- R、必須
- (D)、ディレクティブビジネスケースの場合、総所有コスト (TCO) の比較とディレクティブビジネスケースに必要

- (F)、TCO 比較に必要な全方向性ビジネスケースと、移行とモダナイゼーションのコストを含む方向性ビジネスケースの場合
- O、オプション
- 該当なし、は該当なし

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	推奨される忠実度レベル (最小)
一意の識別子	たとえば、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	R (D)	高
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、市販の (COTS) off-the-shelf ベンダーと製品名を含めません。	R	R (D)	やや高い
COTS ですか？	はいまたはいいえ。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	推奨される忠実 度レベル (最小)
	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
説明	プライマリアプ リケーション関 数とコンテキス ト	R	O	中
緊急性	例えば、戦略ア 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	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
環境	例: 本番稼働前、 本番稼働前、開 発、テスト、サ ンドボックス	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	推奨される忠実度レベル (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
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	推奨される忠実度レベル (最小)
Cost	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

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	推奨される忠実度レベル (最小)
一意の識別子	たとえば、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
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
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	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
オペレーティングシステム	例: REHL 8、Windows Server 2019、AIX 6.1	R	R	やや高い
設定	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
は共有インフラストラクチャですか？	はいまたはいいえ。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します。	R	R (D)	中
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

Cost	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	推奨される忠実度レベル (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/秒冗長)	O	R	中
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
レイテンシー (ミリ秒)	接続された場所間の現在のレイテンシー。	O	O	中

Cost	1 か月あたりの 現在のコスト	該当なし	O	中
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	推奨される忠実 度レベル (最小)
リホスト	各ワークロー ド (人日) の顧客 とパートナーの 労力、1 日あた りの顧客とパー トナーのコスト 率、ツールコス ト、ワークロー ド数	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 の顧客とパート ナーの労力 (人 日)、1 日あた りの顧客とパー トナーのコスト 率、ワークロー ドの数	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 の顧客とパート ナーの労力 (人 日)、1 日あた りの顧客とパー トナーのコスト 率、ワークロー ドの数	該当なし	O	やや高い

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 (Y/N)、必要な AWS リージョンのリスト、コストの再利用	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
時間	対象範囲内のワークロード移行期間 (月)	O	R (F)	やや高い
並列コスト	移行中に現状の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信頼性の高い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、進捗状況に応じて、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における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、データの信頼性を高めることです。

- 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る
- 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
- 通常 CMDBs では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が向上
- 自動アプリケーション識別、TCO 分析、予測実行率、最適化レコメンデーションなどのインサイトの取得
- 信頼性の高い移行ウェブプランニング

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) リクエストに応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考慮し、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要因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。フィールドの経験上、ほとんどの停止した移行プロジェクトには、スコープ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。up-to-dateデータを取得することは、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。

検出ツールを選択する方法

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、どのような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 1 つ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早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および処理されますか？
- 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？
- データが組織の境界を離れる前に、データをスクリーニングできますか？

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、どのような異なる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、サポートされるプラットフォームをスコープに関する現在の情報と照らし合わせて確認します。
- ほとんどのオペレーティングシステムはデータ収集に対応していますか？ 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、各ターゲットシステム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

- 利用可能な機能は何ですか？
- 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を実行しますか？
- 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？ (たとえば、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？
- 料金は各サーバーの料金ですか？ 階層料金ですか？
- 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、検出ツールがない場合、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

- ベンダーは検出出力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 + サービスの割引やバンドルはありますか？

Tip

検出ツールを検索して評価するには、[検出、計画、推奨事項](#)のサイトを使用します。

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時間の経過に伴う組み合わせを回避するには、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た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必要があります。
- 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必要があります。
- 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が。
- 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 AWS コストの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 (たとえば、推奨されるターゲット 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 (たとえば、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ますか?)
- アーキテクチャビュー (たとえば、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 (推奨されるアプリケーションの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、ツールを組織にデプロイ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、チームが関与し、従うプロセスが指示される可能性があります。
- 検出ツールをデプロイするため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)。特にこれらのアセットが一元化されていない場合、各アセットに接続するための検出ツールの認証情報を取得するのは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) に関連する現在のサポートサービスレベルアグリーメント (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と、移行の理由が明確になります。理由が文書化されたら、移行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ステークホルダーを特定し、推進要因を文書化します。通常、組織内の CxOs、シニアマネージャー、主要なテクノロジーリーダー、および独自の顧客です。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、結果が達成されたかどうかを検証するために、移行ジャーニー全体で測定できるメトリクスにリンク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、会社のありたい場所に会話を集中させます。目標とビジネス成果を考慮します。また、クラウド導入の増加に伴う成功について考えてみましょう。

次に、各ドライバーの重要度レベルを確立します。優先順位は何ですか？ 予想される利点は何ですか？ メリットはビジネス目標と成果にどのように役立ちますか？ 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に優先順位を付け、技術的な指針を確立するのに役立ちます。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

テクニカルガイド原則は、ポートフォリオ評価の後半段階で移行戦略の選択に役立ちます。現在のステージでは、それらを特定することに重点を置いています。

指針の原則は、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の主な目標があり、望ましい成果は 6~12 か月で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。結果としての指針となる原則は、可能な限り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です。この場合、lift-and-shiftアプローチは短期的な移行成果を加速します。アプリケーションがオンプレミスデータセンターの外に移動した後、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、望ましい結果を達成します。

移行の計画と実行に関わる人々に指針を文書化し、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、ビジネスドライバーと技術ガイドの原則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た新機能、顧客満足度スコア、実験数	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ファクタリングし、新機能の俊敏性と市場投入までの時間を短縮します。
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 (使用した分の支払い)	時間の経過に伴う支出の変動	<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、インシデントの数	<ol style="list-style-type: none"> 1. アプリケーションを最新のサポートされているオペレーティングシステム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。

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
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。
オンプレミスのまま、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数、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、システムをモダナイズします。 のリホストまたはリプラットフォームを評価します AWS Outposts。

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を収集することに重点を置いています。インベントリは、方向性のあるビジネスケースを作成し、初期移行候補を特定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の更新に役立つ利点があります。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) 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。組織で移行の対象範囲と対象外の範囲が完全に明確である場合は、対象範囲内のシステムにデータ収集を制限することがあります。

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、顧客関係管理 (CRM) アプリケーションを識別し、テスト環境、開発環境、本番稼働環境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 名を使用しますが、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。たとえば、非本番環境を最初に移行できます。環境に従ってアプリケーションのインスタン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、どのアプリケーションまたはサーバーが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かが不確実な場合があります。このような場合、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。たとえば、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、初期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何を取得できるかという点で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より高い完全性レベルには、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。この分析は、方向性のあるビジネスケースを作成し、移行のアプリケーションに優先順位を付けるために実行する必要がある前提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。データの信頼性を高め、移行の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最新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。複数の記録システム、アプリケーション、インフラストラクチャのリストを保持しないでください。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、優先順位付け基準を確立することです。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、優先順位モデルを進化させるために反復的かつ段階的なアプローチを取ることです。

アプリケーションの優先順位付け

この評価ステージでは、初期基準を確立して、低リスクおよび低複雑さのワークロードを優先することに重点を置いています。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複雑さの低い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の推進要因に合わせ、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、非本番環境から実行される、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。たとえば、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です。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、最初と 2 番目の移行ウェブを定義するために有効です。

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、各値にスコアが割り当てられ、数値が大きいほど優先度が高くなります。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高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、最初のいくつか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)	重要性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)
	中	40	
	高	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再設計	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。たとえば、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と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、クラウドの早期導入者にな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、特に初期に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、上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め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がリストの先頭に表示されます。

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、最初のアプリケーション移行候補になります。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ことは、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。

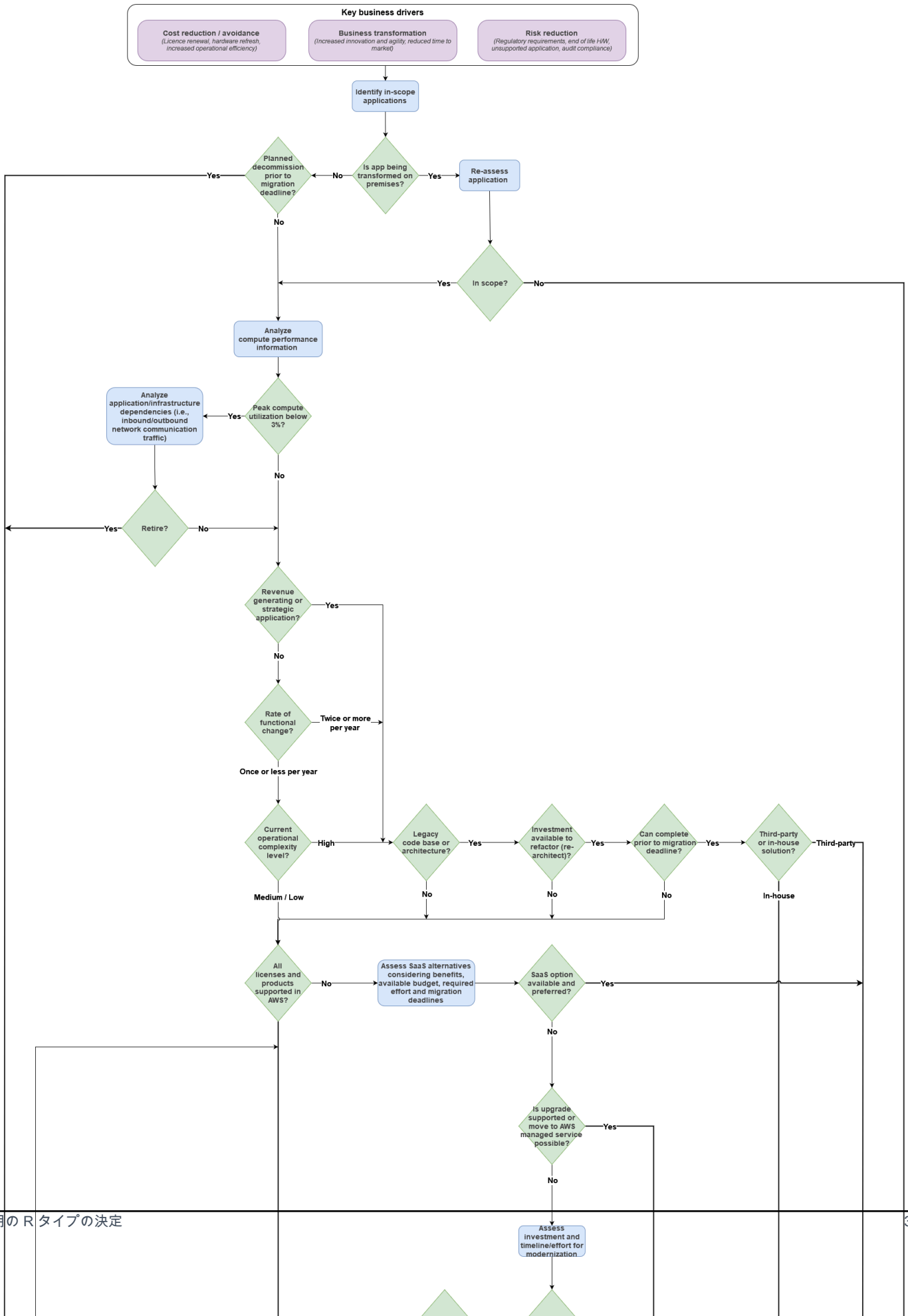
これらの要因によって、ビューが競合することがあります。例えば、移行の主な要因はイノベーションと俊敏性です。同時に、コストを迅速に削減する必要がある場合があります。範囲内のすべてのアプリケーションをモダナイズすると、長期的にコストを削減できますが、事前により大きな投資が必要になります。この場合、1つのアプローチは、リホストやリプラットフォームなど、労力の少ない戦略を使用してアプリケーションを移行することです。これにより、短期間で迅速な効率とコスト削減を実現できます。次に、コスト削減を後の段階でアプリケーションのモダナイズに再投資し、さらなるコスト削減を実現します。

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。デシジョンツリー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、ツリーの上部にあるビジネスドライバーを、組織が定義した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。たとえば、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。たとえば、新しいオペレーティングシステムを搭載した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用に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。6Rs 決定ツリーは、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上書きします。これにより、管理が困難になる可能性のある複数の決定木バージョンを防ぐことができます。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。残りは例外があります。この評価段階でのツリーロジックの調整は、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。

アタッチメント

[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のビジネスケースを理解し、受け入れる必要があります。

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達成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発展します。詳細なケースにより、精度が向上し、プログラムの価値をより完全に把握し、計画の優先順位を把握できます。組織が購入する計画されたビジネス成果を定義および定量化し、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 のサービス アーキテクチャの単純総所有コスト (TCO) の比較。比較は、特定のワークロードボリュームの予想される実行率の差を示しています。
- 移行コストを含む への移行と現状維持 のための純現在価値 (NPV)、投資収益率 (ROI)、ペイバック期間、変更された内部収益率 (MIRR) AWS、3~5 年間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

一般的に、ポートフォリオが大きいほど、開発の少ないケースが必要になります。これは、結果に大きな影響を与えることなく、より広範な仮定を行うことができるためです。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、詳細が必要になります。

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、運用から 3 年以内 AWS、または 5 年以内の 250 台のサーバーで、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派生した値など、この段階で他のビジネス価値の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部品 AWS のサービス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。これは先に進むのに十分です。

- コンピューティングの伸縮性 – 8x5 (24% の使用)、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、使用時にのみ請求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 – コストを最大 75% 削減するための適切な削減計画を使用して、本稼働サーバーやその他のサーバーを高い使用率 (36% 超) で調達する計画を立てます。オプションには、1 年契約と 3 年契約があり、割引を強化す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% 未満のサーバーを特定し、不要になったことを確認して、コスト分析から削除します。
- コンピューティングの適切なサイズ設定 – CPU とメモリの使用率の時系列データを使用して、各サーバー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、適合する 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) ライセンスの適切なサイズ設定 – データベースサーバーでコンピューティングの適切なサイズ設定を行った後、RDBMS ライセンスのニーズを再評価し、Bring Your Own License (BYOL) と Procuring license from を比較し AWS、Amazon Relational Database Service (Amazon RDS) の可能性を調べて削減額を増やします。
- ストレージ – 必要な合計ストレージボリューム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。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の構築に必要なデータと、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

インフラストラクチャの TCO 比較を構築するには、ツールが不可欠です。[AWS プロフェッショナルサービス](#)または [AWS パートナー](#)は、特により広範な移行プロセスを支援するためにそれらに関与させる予定がある場合は、あらゆる種類の方向性のケースを支援できます。

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生成します。

からの[移行評価者](#) 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](#)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ための専用サポート
- SaaS オペレーションの利便性がありますが、顧客ネットワーク内でデータ収集を完全に実行して、分析エンジンにロードする前にスクラブをサポートできます。
- 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のみを評価する (Windows および Linux)
- 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ションする限定オプション
- 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のサポートなし
- TCO 比較以外のビジネスケースの構築を直接サポートしていない

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 比較も行われます。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。市場をリードするツールの主な機能を確認して比較するには、[「検出、計画、推奨事項の移行ツール」](#)を参照してください。

運用コスト最適化の構築

IT 運用の生産性向上は、多くの場合、移行に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の促進と組織変革に関する国際データ株式会社 (IDC) のホ

ホワイトペーパーによると、移行後 AWS、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、これらの利点を方向性のあるケースにサイジングして含めるには、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](#)に適しています。この課題は、単純なベンチマークデータを使用してより簡単に評価およびサイズ設定できるが、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。

第 2 に、コスト削減のソースとして生産性に焦点を当てると、主要な顧客ステークホルダーとプログラムメンバーの間に懸念と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るか、それが影響を受ける人々にとって何を意味するかを明確にしてください。このような問題は、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、内部運用スタッフを開発して新しい役割に移行するためのトラックが含まれています。例えば、DevSecOps チームに参加してコードオートメーションとしてインフラストラクチャを構築し、チームの成長を促進するオートメーションをテストします。
- メリットは、オペレーションのアウトソーシング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、内部スタッフがより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変換に基づいて、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高め、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から AWS Managed Services (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。

最初の変換では、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見積もりを取得するには、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。運用作業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. 各フルタイム同等の (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、必要な人員配置を計算します。オンプレミスでは、この数は約 150 サーバーです。AWS上のサーバーは約 400 台です。
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 (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) による平均生産性の向上を大幅に上回らないことを確認します。

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 番目の変換では、対象範囲内のポートフォリオの現在の総オペレーション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 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) をアカウントマネージャーまたは [AWS Managed Services パートナー](#) に提供します。これにより、変換されたソリューションの AWS のサービス コンポーネントのマネージド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。

完全な方向性のビジネスケースへの拡張

一般的に、完全な方向性のビジネスケースを構築するには、IT 生産性要素の有無にかかわらず TCO 比較を構築し、すべての移行とモダナイゼーションのコストを見積もります。次に、migrate-and-modernize のシナリオのペアと t-migrate-and-modernize シナリオ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、移行 t-migrate-and-modernize 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る 1 組のシナリオの準備です。

- 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
- ストレージ要件の着実な少量の増加
- 既存のシステム機能に一致するQuality-of-service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目的に合致します。これは、前進する権限を得るために十分な価値を迅速に示します。

小規模なポートフォリオの場合、migrate-and-modernizeのペアを追加し、クラウド移行の付加価値の他の側面を示すシナリオt-migrate-and-modernize方が有益です。次に例を示します。

- 中程度と大容量の成長要件がワークロード間で混在し、その成長が見込まれる
- 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、新しい仕様に合わせて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、特に移行コンピテンシーを持つAWSコンサルティングパートナーで、migrate-and-modernizeのシナリオと移行t-migrate-and-modernizeのシナリオの両方をサポートできる場合、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。

- 移行およびt-migrate-and-modernizeシナリオのコスト。最も基本的なケースには、以下が含まれます。
 - 現在のインフラストラクチャ設定のビジネスケース期間における総所有コスト
 - 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
- migrate-and-modernizeのコスト。シナリオには以下が含まれます。
 - プログラムのセットアップには、詳細な検出、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、まだ導入されていない場合のランディングゾーン的确立、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が含まれます。
 - 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 Edge](#) 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) を含む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。
- ストランド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

プログラムを成功させるためにセットアップするには、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事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のセクション](#)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、使用した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技術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 の見積もりを提供できます。

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示すには、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保ちます。

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設計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、初期ペイバックを加速し、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しやすくなります。[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 (再設計) 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、パートナー AWS 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見積もりをシードします。

ディレクショナルケースの構築に役立つように、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 の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、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	Do nothing, with no cost, no benefits, and no reduction in technology debt.	–	–	–
Retire	Estimate decommissioning the hardware	–	–	–

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
	equipment used, if any.			
Relocate	Estimate copying the workload within VMware using VMware tools. This includes copying the data, smoke testing to verify, and any hardware decommissioning. The effort to relocate VMs is typically less than for low-complexity rehost patterns.	–	–	–

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
Rehost	Estimate copying the workload and data with an image copy, smoke testing, high availability (HA) and disaster recovery (DR) testing where appropriate for production servers, and any hardware decommissioning. The best practice is to use tools such as AWS Application Migration Service . Divide workloads into low, medium, and high complexity, based on factors such as whether a database or other infrastructure software is running, database complexity, whether clustered,	Effort per app per server Low Medium High	Migration 10–14 16–24 26–38	HA/DR test 3–5 4–6 8–12

移行戦略	見積りプロセス integration complexity, and data volumes.	[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
Replatform	For replatform migration s that include upgrades to operating system or RDBMS version, take the estimate for a rehost, and add time to run a rebuild and smoke test on the new platform. If the replatform includes changing the technology of the platform, estimate additional time for the use of the conversion tools, such as AWS Schema Conversion Tool and AWS Database Migration Service , and a more complete application test. An example of changing the technology is	Effort per app per server	Version up	Technology change
		Low	Add 1–3	Add 10–15
		Medium	Add 2–5	Add 20–30
		High	Add 4–8	Add 40–60

移行戦略	見積りプロセス	[Elements] (要素)	人時	人時
	migrating away from a proprietary commercial database to an open source replacement.			
Repurchase	Estimate data extraction, transformation, and uploading into the newly purchased SaaS service replacement, and any hardware decommissioning.	–	–	–

移行インフラストラクチャのコストの見積もり

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予算 AWS のサービス (特にコンピューティングとストレージ)
- 各移行ウェーブの完了に伴う AWS ユーティリティコストの増加
- 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](#) または [Site-to-Site VPN](#) から WAN 上のポイント AWS へのリンクを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、費用対効果の高いソリューションです。そうでない場合、[AWS Snowball Edge](#)やなどの AWS メディア交換デバイスは、ほとんどのソリューション [AWS Snowball Edge](#)を提供します AWS リージョン。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、いつコストが発生するかを正確に判断するためのウェーブプランがある可能性は低くなります。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。
- 廃止予定の既存のインフラストラクチャのコストを同じ期間にわたって一定の割合で削減する。

既存のインフラストラクチャがランプダウンする 1~2 か月前に、AWS コストのランプアップを開始します。これにより、各ウェーブの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、テストの時間と、置き換えられたインフラストラクチャのコストの発生を停止するために必要な廃止作業を完了する追加の時間が含まれます。

廃止コストの見積もり

再デプロイできない機器を廃止し、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。

完全な方向性の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、それぞれに割引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

スペ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、各シナリオの現状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移行する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)

次の例のように、ケースの最初の反復を使用して、予想される財務パフォーマンスが絞り込む必要があるかどうかを判断します。

- 支払いが遅すぎる場合は、次のような移行のコストを高速化して削減する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。
 - 技術的に可能な場合は、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 (リアーキテクト) 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、より強力な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、最初に商用性が低い特定のワークロードを除外できますか？たとえば、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条件が異

なるため、サードパーティーのソフトウェアライセンスのコストが高く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予想される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フェーズ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ためのものです。
- 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ときは、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の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 つは、[ポートフォリオ検出と初期計画](#)であり、詳細な評価のために[アプリケーションのサブセットを優先](#)することでした。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。評価と今後のアーキテクチャ設計のプロセスは、潜在的なブロック要因を明らかにし、より大きな範囲の移行に取って代わ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ときでもあります。

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
- 移行候補の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、オペレーション)
- 移行計画に関する文書化されたカットオーバーの考慮事項
- 推定 AWS 実行レート

詳細な評価データ要件について

次の表は、移行内のアプリケーションとその関連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。

- R、必須
- 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	検出、設計、移行戦略	推定実行レート	推奨される忠実度レベル (最小)
一意の識別子	たとえば、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
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、市販の (COTS) off-the-shelf ベンダーと製品名を含めません。	R	R	高
COTS ですか?	はいまたはいいえ。商用アプリケーションか内部開発か	R	R	高
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

属性名	説明	検出、設計、移行戦略	推定実行レート	推奨される忠実度レベル (最小)
説明	プライマリアプリケーション関数とコンテキスト	R	O	高
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	R	O	高
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
環境	例: 本番稼働前、本番稼働前、開発、テスト、サンドボックス	R	R	高
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

属性名	説明	検出、設計、移行戦略	推定実行レート	推奨される忠実度レベル (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
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
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
Cost	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
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

属性名	説明	検出、設計、移行戦略	推定実行レート	推奨される忠実度レベル (最小)
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
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
収益を生み出すアプリケーションが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か、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
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/外部ユーザー/顧客など	R	R	やや高い

属性名	説明	検出、設計、移行戦略	推定実行レート	推奨される忠実度レベル (最小)
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
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	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をモニタリングするために使用される製品	R	O	やや高い

属性名	説明	検出、設計、移行戦略	推定実行レート	推奨される忠実度レベル (最小)
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、アカウントプレースメント、ネットワーク、セキュリティなど	R	R	高

インフラストラクチャ

属性名	説明	検出、設計、移行戦略	推定実行レート	推奨される忠実度レベル (最小)
一意の識別子	たとえば、サーバー ID などで。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	R	O	高

	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
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
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
製品名	商用ベンダーと製品名 (VMware ESXi、IBM Power Systems、Exadata など)	R	R	高
オペレーティングシステム	例: REHL 8、Windows Server 2019、AIX 6.1	R	R	高

設定	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
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
ライセンス	商品ライセンスタイプ (RHEL Standard など)	R	R	高
は共有インフラストラクチャですか？	はいまたはいいえ。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します。	R	O	高

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
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
移行戦略、パターン、ツール	例えば、移行用の 6R の 1 つ、特定の技術パターン、移行ツール	R	O	高
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。たとえば、組織がアプリケーションを完全に理解していることを前提とするのが一般的です。これは自然であり、多くの場合当てはまります。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、検出プロセスの負担が軽減されま

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せん。

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、移行するときは、継続的な移行ウェーブにサーバーを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回避して、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ステークホルダー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し、インフラストラクチャまで行きます。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、アプリケーションまたはサービスとそのユーザーにまで至ります。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、application-to-infrastructure間のマッピングが不明な場合に役立ちます。一般的に、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、クラウド上の小さな変更で迅速に対処できる非効率性を特定できます。

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

を検証します。このツールは、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、内部と外部の両方の他のアプリケーションと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。たとえば、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、オープン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。ただし、より長い間隔 (通常は少数の割合) で発生する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場合がありますが、それでもエラーや欠落データが発生しやすくなります。

次に、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を取得します。次に、次のアンケートを使用して、詳細な評価プロセスについて説明します。特定されたステークホルダー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。

General

- 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？ ビジネス戦略アプリケーションか、サポートビジネスアプリケーションか。他のシステムによって共有され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？
- これは内部向けか外部向けか。

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 これらは内部チームまたは 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ですか？ 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？ アプリケーションへのアクセス方法 同時ユーザー数はいくつですか？
- テクノロジーを最後に更新したのはいつですか？ 更新は今後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、中程度および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) を教えてください。運用タイムゾーンは何ですか？
- 変更凍結期間とは何ですか？
- 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、日付を教えてください。
- このアプリケーションの一部である、または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

- 現在の変化率はどのくらいですか (毎週、毎月、四半期ごと、または毎年)？
- 開発ライフサイクル (テスト、開発、QA、UAT、本番前、本番など) は何ですか？
- 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？
- 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、知識と意見を対比します。

回復性

- 現在のバックアップ方法は何ですか？ バックアップに使用される製品 バックアップスケジュールとは何ですか？ 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、DR シミュレ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 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は何ですか？

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ますか？

- データベースへの同時接続の一般的な数を教えてください。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、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、コンポーネントと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散していますか？ これらの場所 (WAN、VPN など) 間の接続は何ですか？
- 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、移行の重要な成功要因です。シンプルでも複雑でも、あらゆる種類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、アーキテクチャの変更が予想されない場合でも、潜在的なブロック要因、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な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ンス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、数十の技術的およびビジネス上の問題について AWS、によって検証されたクラウドベースのソリューションのコレクションを提供します。これには、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 Documentation](#)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、基礎を学習するための実践的なチュートリアルと詳細な演習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あるかに応じて、AWS 基盤が既に存在している可能性があります。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。
- 基本 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 AWS 基盤がまだ確立されていない可能性があります。基盤が確立されていないと、アプリケーション設計の範囲が制限されたり、定義のためのさらなる作業が必要になる可能性があります。この場合、ランディングゾーンの基本的な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、ネットワーキング、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、クラウドオペレーション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ための最も簡単な方法を提供します。は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 AWS Organizations、何千人ものお客様とクラウドに移行する際に、AWS ベストプラクティスベースのエクスペリエンスを継続的に管理、ガバナンス、実装できます。

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、潜在的な制限が明らかになる可能性があるため、初期と見なされます。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。たとえば、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は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ます)?

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、変更なしでアプリケーションを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セットに存在する場合があります。たとえば、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。こ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ク要因はありますか?
- 現在の欠点は何ですか?
- スコープ内とスコープ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こ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中に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、アカウント AWS リージョン、VPCs、アベイラビリティーゾーンなど) の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とサービス (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精査された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
Application Load Balancer	バックエンドサービス間でトラフィックを分散します	オンプレミス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。
アプリケーションのセキュリティ	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。

名前	説明	詳細
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)
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

名前	説明	詳細
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	AWS Elastic Disaster Recovery

コンポーネントを特定したら、任意のツールを使用してそれらを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？
- 設計を進化させることはできますか？
- 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決定事項

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。たとえば、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。

Tagging

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されるタグ付けと、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前提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、異なるアプリケーションコンポーネントでは、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でのコンポーネントの移動など、移行に対する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。たとえば、リホスト戦略は、などの移行ツールで実装できます [AWS Application Migration Service](#)。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 \(AWS DMS\)](#)、[AWS Schema Conversion Tool \(AWS SCT\)](#)、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、特定の製品またはの Software as a Service (SaaS) ソリューションに AMI を使用できます [AWS Marketplace](#)。

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、および移行の運用準備を検討してください。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。

サービス管理とオペレーション

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で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？
- 予想されるサービスレベルアグリーメント (SLA) とは何ですか？
- 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はできていますか？
- 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ます。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、または検証の要件を概説します。

リスク、前提、問題、依存関係

未解決の未解決のリスク、前提、潜在的な問題を文書化します。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

この評価ステージでは、ポートフォリオ検出と[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検出と分析を完了することに重点を置いています。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のベースラインを反復して確立することです。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要が含まれます。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成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について

次の表は、移行内のアプリケーションとその関連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。

- R、必須
- O、オプション
- 該当なし、は該当なし

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	推奨される忠実度レベル (最小)
一意の識別子	たとえば、アプリケーション ID などです。	R	R	高

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	推奨される忠実度レベル (最小)
	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。			
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、市販の (COTS) off-the-shelf ベンダーと製品名を含めます。	R	R	高
COTS ですか?	はいまたはいいえ。商用アプリケーションか内部開発か	R	R	高
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

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	推奨される忠実度レベル (最小)
説明	プライマリアプリケーション関数とコンテキスト	R	R	高
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
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
環境	例: 本番稼働前、本番稼働前、開発、テスト、サンドボックス	R	R	高
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

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	推奨される忠実度レベル (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
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
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
Cost	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	推奨される忠実度レベル (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
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
DR 情報	ディザスタリカバリコンポーネント	R	R	高
移行戦略	例えば、に移行するための 6 R のいずれか AWS	R	R	高
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

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	推奨される忠実度レベル (最小)
一意の識別子	たとえば、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
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
DNS 名 (完全修飾ドメイン名、または FQDN)	[DNS 名]	R	O	高
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
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

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	推奨される忠実度レベル (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
オペレーティングシステム	例: RHEL 8、Windows Server 2019、AIX 6.1	R	R	高
設定	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
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
ライセンス	商品ライセンスタイプ (RHEL Standard など)	R	R	高

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	推奨される忠実度レベル (最小)
は共有インフラストラクチャですか？	はいまたはいいえ。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します。	R	R	高
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

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	推奨される忠実度レベル (最小)
Cost	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
データ転送の推定量 (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セットごと	O	R	中

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	推奨される忠実度レベル (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)	R	R	やや高い
リンク使用率	ピーク使用率と平均使用率、	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	推奨される忠実度レベル (最小)
	アウトバウンドデータ転送 (GB/月)			
レイテンシー (ミリ秒)	接続された場所間の現在のレイテンシー。	R	O	高
Cost	1 か月あたりの現在のコスト	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	推奨される忠実度レベル (最小)
リホスト	各ワークロードの顧客とパートナーの労力 (人日)、1 日あたりの顧客とパートナーのコスト率、ツールコスト、ワークロード数	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) の顧客とパートナーの労力、1 日あたりの顧客およびパートナーのコスト率、ワークロードの数	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	推奨される忠実度レベル (最小)
リファクタリング	各ワークロード (人日) の顧客とパートナーの労力、1日あたりの顧客およびパートナーのコスト率、ワークロードの数	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 (Y/N)、AWS リージョン 必要な のリスト、コストを再利用する	該当なし	R	やや高い
人材と変化	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
時間	対象範囲内のワークロード移行期間 (月)	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	推奨される忠実度レベル (最小)
並列コスト	移行中に現状のままの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と、[「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明確にします。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。

- Application-to-infrastructure の依存関係は、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。たとえば、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示します。アプリケーションコンポーネ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。

- Application-to-application 依存関係は、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。
- Application-to-infrastructure サービスの依存関係は、インフラストラクチャサービス自体がアプリケーションであるため、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グループを含めることができます。重要なのは、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プールへの依存関係です。一方、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 への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。false 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を個別にアプローチし、クラウド環境でなどのサービスを有効にして依存関係に対処する AWS Directory Service for Microsoft Active Directory 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、各アプリケーションコンポーネントの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 つです ([「6 R の移行戦略の反復」](#) セクションを参照してください)。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など、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。

移行の範囲と利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に注意してください。

優先順位付け基準の反復

移行ウェー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ーブプランニングに移行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、学んだ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始め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、ウェー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ガイドの原則](#)」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)	重要性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	
	Server	50	
	Linux	20	
ビジネスの重要性	高	60	高 (1x)
	中	40	
	低	20	

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
アーキテクチャ	密結合	60	高 (1x)
	疎結合	20	
運用モデル	従来の - CI/CD なし	60	中高 (0.8x)
	基本的な CI/CD	40	
	完全な 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合わせ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)
	Microsoft SQL	70	
	その他	20	

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
オペレーティングシステム	Server	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 Rs 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、デフォルトの決定ツリーが導入されました。ツリーを改訂し、初期パイロットアプリケーションの移行全体の学習を考慮し、ビジネスドライバー、優先順位基準、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でディシジョンツリーを検証し、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、それに応じて口

ジックを更新します。結果のツリーは、アプリケーション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のが重要です。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すが、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 Rs 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。たとえば、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、スキルを推定するとき、および移行ウェーブプランを作成するとき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く、トラフィック量が多く、クエリが複雑な場合、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、カットオーバープランニング、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、ウェーブの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、技術的な指針などの他の要因を組み合わせることもできます。たとえば、ウェーブは環境 (開発、テスト、本番環境など) 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、移行戦略の完全なビューを取得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。たとえば、アプリケーション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は、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の関係、またはアセットから制約への関係であり、コンポーネントの場所に依存しません。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、もう一方を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。

ハード依存関係は、2 つ以上のアセット間の関係、またはアセットから制約への関係であり、場所によって異なります。たとえば、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、強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が発生し、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的な制約、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。プロセス全体でサポートするスペシャリストを提供す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 AWS 連携すること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動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強い依存関係を持つ可能性があるためです。

移行戦略は、ウェー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、準備を必要とす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、後続の波に配置される可能性があります。

ウェーブプランの作成

アプリケーションのウェーブ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、ウェーブ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、ウェー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。

ウェーブの所有権とガバナンスを確立することは、ウェーブ作業、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ために重要です。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。

ウェーブプランの概要を表示するには、デフォルトのウェーブコンストラクトから始めます。ウェーブ内で何が起こりますか？初期入力が定義されたら、ウェーブを開始できます。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、他の内部チームや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を概説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、何をする必要がありますか？
3. ターゲットインフラストラクチャを準備します。たとえば、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. 移行ツールを操作します。たとえば、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施します。参加しているすべてのチームメンバーをグループ化し、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. 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、コンプライアンスと規制要件 (移行前と移行後のワークロード検証など) 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 運用上の引き渡しとレポート用のメトリクスの取得を確認して、ウェーブクローズを実行します。

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、波の容量、関係者、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を教えてください。
- 期限は何ですか？
- 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サイクルは何ですか？
- 移行を避ける日付 (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) はありますか？

これらの考慮事項を考慮して、ウェーブをプランにプロットします。移行プロセスを高速化するには、可能な限り波を重ね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、運用の引き渡しに焦点を当てます。つまり、異なるチームがプロセスの各半分に参与し、ある程度の効率を得ることができません。例えば、ターゲットインフラストラクチャの準備に参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のようなアプロー

子を容易にするために、ほとんどの波の長さや構造が似ていることをお勧めします。ただし、ウェーブプランニングプロセスでは、特定のウェーブのサイズを拡張して、依存関係や運用要件を満たすことができます。

次に、特定された依存関係グループに基づいて、含めることができる依存関係グループの数の観点からウェーブの最大サイズを決定します。波のサイズは通常、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、早期計画中は、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を確認したら、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、波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。長時間実行される移行プログラムは、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。サーバーは追加または削除され、新しいインフラストラクチャは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を変更する必要があ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、移行プログラムのガバナンスで定義されているスコープ、または .csv ファイル、スプレッドシート、データベースを管理するためのツールである場合があります。関係者がアクションを実行できるように、変更を特定し、影響を分析し、変更を関係者に伝える必要があります。ウェーブプランは結果として反復されます。

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、変換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性を提供するよう設計されています。

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、作業を段階的に優先順位付けする方法に関する意思決定に役立つ財務分析を提供する
- 詳細な再検討により、元の方向性財務ケースを検証、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 生産性と外部委託された運用効率
 - 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、スケールの推定、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目標ベース
- プログラムの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うことができます。
- 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、仮定。これにより、プログラムからの主なメリットを実現するためのベースラインが形成されます。

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 – 財務パフォーマンスの最小期待値を評価するには、ステータスクォーへの最小予想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ちます。このシナリオでは、可用性やレジリエンスなど、他の quality-of-service のニーズに対して予想される最小限のキャパシティ増加度と最小限の変更をモデル化します。最小の変更により、現在の運用モデルのコストとリソースの非効率が最も低くなります。

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通知するには、ビジネスが期待することを反映する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、ビジネスからの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前提をまだ作成する必要がある場合は、前提が当てはまるシナリオとそうでないシナリオの両方を作成します。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 個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、維持が困難になります。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絞り込む

ポートフォリオ分析を完了し、ターゲットの設計とサイズ設定を準備したら AWS のサービス、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ハードウェアの更新、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。
- COM データセンターとコロケーション施設のコストには、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、成長に合わせてサイズ設定され、容量を満たすように指定、シナリオの高可用性とディザスタリカバリ (DR) レベルが含まれます。
- 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は、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対して契約料金を使用して計算されます。
- 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いて、必要に応じて技術サポートと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設定、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ー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再利用できるアセットと切り替えることができるアセットの検証、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 運用を改善し、効率バリュー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および開発するには、主に 2 つのアプローチがあります。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 運用とサポートが社内で管理されている場合、ビジネスケースの焦点は次の通りです。

- 移行と対象範囲に含まれる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、通常より価値の高い他の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、チームに進歩の機会とより大きな報酬を与え、組織により多くの価値をもたらす

チーム内の各ロール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 オペレーションの大部分を消費し、チーム内のさまざまなロールの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を示しています。この表には、生産性の実現方法の説明が含まれています。

Note

リストされているアクティビティは通常、複数の異なるロール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ロール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) 別に編成された IT 運用チームでは、設備投資の計画と予算編成が各タワーのタワーリードに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	設計が簡素化され、考慮するパラメータが少なくなります。
設備投資の計画と予算編成	高	OPEX 中心の Elastic サービスは、実質的にすべての予算と計画の問題を削除します。
購入	高	の確立後、調達 AWS アカウントが大幅に簡素化されます。
容量プランニング	中～非常に高い	ネットワーキング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チューニングは必要なく、他のサービスには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。
サーバーの可用性と通信のモニタリング	高	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
セキュリティ管理	中	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、施設の セキュリティ責任 AWS を持つことで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	クラウドでの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。
ラックとスタッキング – ハードウェアの物流	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	バックアップは、AWS ツール、柔軟なストレージシステム、自動化によって広範囲に簡素化されます。
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。

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	に移行したエステートのハードウェアセットアップのアクティビティは通常、VPN VPNs または AWS データセンターへの接続を確立するための WAN AWS Direct Connect 接続デバイスを除き、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	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	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ます。また、AWS が提供するメカニズムにより、安全なアーキテクチャでの評価が簡素化されます。
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	インフラストラクチャサービスの拡張、更新、または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りますが、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は排除されます。

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
データセンター施設管理	中～非常に高い	<p>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、一部の作業は、WAN リンクネットワークデバイスと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施設を提供するためのものです。</p>
アプリケーションアーキテクチャ、開発、管理、テスト	低	<p>アジャイル開発ツールチェーンを、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と組み合わせて使用すると、アプリケーション開発のリードタイムが短縮され、多くの手動テストステップが不要になります。</p>

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
アプリケーションソフトウェアのインストールと設定	中	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、ランディングゾーンを使用して簡素化されます。ランディングゾーンは、を使用して簡単に設定できます AWS Control Tower。
IT サポート	中	L1 および L2 サポートの削減は、セルフサービスのプロビジョニングに Service Catalog 機能を使用することで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ことで実現されます (停止の削減と自動スケーリングとエッジコンピューティングの設定)。
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で提供されます。
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	Minimal	
ドキュメント	Minimal	
アプリケーションとパフォーマンスのモニタリング	Minimal	

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
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	Minimal	
アプリケーションソフトウェアのインストールと設定	Minimal	
アプリケーション L3 のサポート (予算編成と長距離キャパシティプランニングを除く)	Minimal	

次の表は、ワークロード削減の各レベルで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	60% ~ 90%
中	30% ~ 70%
低	10% ~ 35%
Minimal	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。範囲の中間と下限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。

プログラムが進むにつれて、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重要です。このデータは、運用を推定するための改善されたベースを構築し、新しいプロジェクトと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 (FOM) AWS のコスト配分を準備することができます。 [AWS Managed Services](#) 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あります。

残りの IT オペレーション、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、契約上のペナルティ (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、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。消費ベースの料金とは、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 つの要因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 ccustomers 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、参加している顧客が年間停止回数を 73% 削減し、平均復旧時間 (MTTR) を 58% 削減し、生産性の低下を 94% 削減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、さらなる回復力が達成されます。テストオートメーションを備えた CI/CD パイプラインを導入してビジネスの俊敏性を高めると、ソフトウェア欠陥が開発サイクルの早い段階で発見され、ソフトウェア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、パフォーマンスの減速、計画されたバッチおよびメンテナンスウィンドウのオーバーラン、主要な関数のバグ、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可能性があるトランザクションの数
 - 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時間の追加コストと、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と損失時間のコストへの影響

次に、予想されると、サービス中断にかかる時間のより控えめな削減を評価し、耐障害性を高める必要があります。たとえば、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て停止回数と MTTR を減らし、目標復旧時間 (RTO) と目標復旧時点 (RPO) を改善
- 自動スケーリングなどの機能を使用した、スローダウンの削減、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。

これらを移行およびモダナイズするアプリケーションのポートフォリオにまとめ、ケースの毎年の予想およびより保守的なビジネス価値の数値を計算します。メリットは移行スケジュールに沿って拡大し、寄与するアプリケーションの使用量の増加の期待に沿ってボリュームを拡大する必要があります。

ビジネスの俊敏性バリューモデルを開発する

ビジネスの俊敏性は、AWS 顧客が移行する主な理由です AWS。 [IDC 2018](#) AWS の顧客調査では、ビジネス俊敏性のメリットは、測定された総メリットの 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 倍以上を占めていると回答しました。

あらゆる変革から生じる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の源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るにつれて、ビジネスの俊敏性バリューモデルを段階的に改善し、拡張します。これにより、ビジネスケースが関連付けられるため、プログラムを運営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を最大に向上させる機会があるワークロードを選択します。
 - 収益を生成するワークロード
 - 効率の向上を推進し、ビジネスからコストを排除するスコープを持つ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
- 収益と効率を生成するワークロードについては、以下を実行します。
 - 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増加または運用効率を現実的かつ控えめに評価します。
 - アプリケーション開発速度を高め、インフラストラクチャのデプロイ時間を短縮できる 1 年あたりのメジャーリリースとマイナーリリースの数 AWS の増加を見積もります。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、以下を実行します。
 - メジャーなアプリケーションアップグレードとマイナーなアプリケーションアップグレードが推進できることが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々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、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て、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必要があります。

継続的な評価と改善

この評価ステージでは、次の 2 つの側面に焦点を当てます。

- アプリケーションの波ごとに進行中の詳細なアプリケーション評価
- ポートフォリオの継続的な進化と改善

最初の側面である継続的な詳細なアプリケーション評価では、特定の波の各アプリケーション、提案された AWS 設計、移行戦略を完全に理解するために、アーキテクチャとテクノロジーレベルまで詳細な検出と分析に焦点を当てています。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。

ポートフォリオの継続的な進化と改善という 2 つ目の側面は、ポートフォリオ管理と、ビジネスケースの進化と追跡など、時間の経過とともにアプリケーションを改善する計画を立てる方法に焦点を当てています。

このステージの主な移行結果は次のとおりです。

- ウェーブごとに検証された移行範囲
- 特定の移行ウェーブにおけるアプリケーションの文書化されたターゲットアーキテクチャと移行戦略
- 識別および検証された移行パターンとツール
- ウェーブごとの文書化された要件 (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

このステージ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
- アーキテクチャとテクノロジーの変更案とその期待される利点
- プラットフォームの要件 (セキュリティ、AWS インフラストラクチャ、オペレーション)
- 実装計画

継続的な評価データ要件を理解する

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の組み合わせです。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。

- ウェーブ評価とアプリケーションの最適化には、[「優先順位付けされたアプリケーション評価」](#)セクションのデータ要件を使用します。
- 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ます。
- 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」](#)セクションを参照してください。

詳細なウェーブ評価

アプリケーションの詳細な評価は、移行ウェーブに先立って、移行の主要なイネーブラーとして、[優先順位付けされたアプリケーション評価](#)ステージと同じ要件と推奨事項があります。目標は、特定のウェーブにおけるアプリケーションの現在の状態を詳細に理解し、運用面、ツール、特定の移行パターンなど、将来のアーキテクチャ設計と移行戦略を作成することです。

[優先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のグループに適用します。移行計画の各ウェーブの前にこのプロセスを繰り返します。重要なのは、詳細な評価とウェーブの開始の間に十分な時間をスケジュールすることです。必要な時間は、ウェーブ要件を実装し、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。これらのチームと協力して、詳細なウェーブ評価とウェーブをスケジュールします。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。

最適化とモダナイゼーションの評価

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、移行するワークロードの評価と似ています AWS。主な変更点は、評価を実行するデータソースです。には AWS、で実行されている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 out-of-the-box 使えるツールや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モダナイズする内容と方法は、独自の推進要因と状況に基づきます。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適応させ、学んだ教訓を組み込む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導入など、アプリケーションを次のレベルに進め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。最適化とモダナイゼーション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 最適化と IT コストの節約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で、コストを削減し、パフォーマンスを向上させるためにワークロードの AWS リソースを推奨します。
- [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構築により多くの時間を費やし、管理時間を短縮できます。
- [Amazon S3 Storage Lens](#) は、オブジェクトストレージの使用状況とアクティビティの傾向を組織全体で可視化します。これにより、コスト効率を向上させ、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レコメンデーションが作成されます。
- [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、コード品質を向上させ、アプリケーションの最も高価なコード行を特定するためのインテリジェントなレコメンデーション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、必要な場所で一貫した AWS エクスペリエンスを提供します。

追加リソース

- [コストの最適化とイノベーション: アプリケーションのモダナイゼーション入門](#) (ブログ記事)
- [サーバーレスウェブアプリケーションのコストの最適化](#) (ブログ記事)
- [Windows on 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](#) (AWS re:Invent 2020)
- [AWS マイクロサービスガイド](#)

ウェーブプランの反復

移行プログラムが進み、より多くのウェーブが移行されるにつれて、学んだ教訓と変化するビジネスの優先順位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、ビジネスドライバーと組織の変化を再評価し、移行ウェーブプランがまだ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します。何が起きているかの可視性が失われないように、[ウェーブプラン](#)を最新の状態に保つ必要があります。計画では、何が配信されているかを反映および追跡し、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

移行が進むにつれて、特に長時間実行されるプログラムでは、ビジネス上のプレッシャーによって移行とモダナイゼーション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。

新しい情報が利用可能になるとビジネスケースを進化させ、詳細なビジネスケースに記載されている期待に照らして実際の商用パフォーマンスを追跡することをお勧めします。これらの推奨事項には以下が含まれます。

- ビジネスの優先順位に影響を与え、IT 戦略とそれに伴うアプリケーションポートフォリオに影響を与える組織における新しい構造変更
- アプリケーションポートフォリオの一部の商業上の重要性の高まり、または移行とモダナイゼーションが達成の対象となる変更
- サイズ調整、数値化、段階的なモダナイゼーションのケースの確認など、移行されたアプリケーションの実際のリソース使用率データの可用性
- 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可用性、および運用上の改善と自動化の可能性の分析
- ソフトウェア開発とメンテナンスのサイクル時間の変化、開発ステージとサービスの可用性情報別のソフトウェア不具合、さらに改善が見込める領域の根本原因分析を測定するデータの可用性

ビジネスケースに対するパフォーマンスを追跡することで、移行開始後により簡単に評価および定量化できるさらなる改善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もたらす方向に変革を誘導するはるかに優れた能力を備えています。

これは、ケース内の IT 生産性、レジリエンス、ビジネスの俊敏性にとって特に重要です。これらは通常、事前に評価することが難しい大きな要因と難しい要因の両方です。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る問題を深く掘り下げて解決できます。または、ビジネスケースを調整して、最も進行中の財務パフォーマンスの最適化を実現するイン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 リファレンス

- [Amazon Builders' Library](#)
- [アプリケーションのモダナイゼーション](#) (AWS re: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: アプリケーションのモダナイゼーション入門](#) (ブログ記事)
- 「[検出、計画、推奨の移行ツール](#)」
- [AWS Documentation](#)
- [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](#)
- [サーバーレスウェブアプリケーションのコストの最適化](#) (ブログ記事)
- [AWS 規範ガイド](#)
- [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](#)
- [Windows on AWS](#) (ブログ)

AWS のサービス

- [AWS App2Container](#)
- [AWS Application Migration Service](#)
- [Amazon 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](#)
- 「[Migration Evaluator](#)」
- [AWS Migration Hub 戦略の推奨事項](#)
- [AWS ランディングゾーン](#)
- [AWS 料金見積りツール](#)
- [AWS Schema Conversion Tool](#)
- [Amazon S3 Storage Lens](#)
- [AWS Snowball Edge](#)
- [AWS Snowball Edge](#)
- [Site-to-Site VPN](#)

その他のリソース

- [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の促進](#)
- [IDC 2018 アンケート](#)

ドキュメント履歴

次の表に、この戦略の重要な変更を示します。今後の更新に関する通知を受け取る場合は、[RSS フィード](#) をサブスクライブできます。

変更	説明	日付
更新	ポートフォリオ検出と初期計画のセクション「検出の加速と初期計画」の名前を変更し、ディビジョンツリー図を更新しました。	2024 年 5 月 20 日
二	初版発行	2021 年 11 月 12 日

AWS 規範ガイドの用語集

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

数字

7 Rs

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

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

A

ABAC

「[属性ベースのアクセス制御](#)」をご覧ください。

抽象化されたサービス

「[マネージドユーザー](#)」をご覧ください。

ACID

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

アクティブ/アクティブ移行

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

アクティブ/パッシブ移行

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

集計関数

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

AI

「[人工知能](#)」をご覧ください。

AIOps

「[AI オペレーション](#)」をご覧ください。

匿名化

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

アンチパターン

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

アプリケーション制御

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

アプリケーションポートフォリオ

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

人工知能 (AI)

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

AI オペレーション (AIOps)

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

非対称暗号化

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

原子性、一貫性、分離性、耐久性 (ACID)

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

属性ベースのアクセス制御 (ABAC)

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

信頼できるデータソース

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

アベイラビリティゾーン (AZ)

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

AWS クラウド導入フレームワーク (AWS CAF)

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

AWS ワークロード認定フレームワーク (AWS WQF)

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

B

不正なボット

個人や組織に混乱や損害を与えることを目的とした [ボット](#)。

BCP

「[ビジネス継続性計画 \(BCP\)](#)」をご覧ください。

動作グラフ

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

ビッグエンディアンシステム

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

二項分類

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

ブルームフィルター

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

ブルー/グリーンデプロイ

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

ボット

インターネット経由で自動タスクを実行し、人間のアクティビティややり取りをシミュレートするソフトウェアアプリケーション。インターネット上の情報のインデックスを作成するウェブクロウラーなど、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、個人や組織を混乱させたり、損害を与えたりすることを意図したものもあります。

ボットネット

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

ブランチ

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

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

ブレイクグラスアクセス

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

ブラウнフィールド戦略

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

バッファキャッシュ

アクセス頻度が最も高いデータが保存されるメモリ領域。

ビジネス能力

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

ビジネス継続性計画 (BCP)

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

C

CAF

「[AWS クラウド導入フレームワーク](#)」を参照してください

カナリアデプロイ

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

CCoE

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

CDC

「[変更データキャプチャ](#)」を参照してください。

変更データキャプチャ (CDC)

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

カオスエンジニアリング

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

CI/CD

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

分類

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

クライアント側の暗号化

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

Cloud Center of Excellence (CCoE)

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

クラウドコンピューティング

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

クラウド運用モデル

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

導入のクラウドステージ

組織が、AWS クラウドへの移行時に通常実行する 4 つの段階。

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

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

CMDB

「[構成管理データベース \(CMDB\)](#)」を参照してください。

コードリポジトリ

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

コールドキャッシュ

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

コールドデータ

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

コンピュータビジョン (CV)

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

設定ドリフト

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

構成管理データベース (CMDB)

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

コンフォーマンスパック

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

継続的インテグレーションと継続的デリバリー (CI/CD)

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

CV

[「コンピュータビジョン」](#) を参照してください。

D

保管中のデータ

ストレージ内にあるデータなど、常に自社のネットワーク内にあるデータ。

データ分類

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

データドリフト

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

転送中のデータ

ネットワーク内 (ネットワークリソース間など) を活発に移動するデータ。

データメッシュ

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

データ最小化

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

データ境界

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

データの前処理

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

データ出所

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

データ件名

データを収集、処理している個人。

データウェアハウス

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

データベース定義言語 (DDL)

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

データベース操作言語 (DML)

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

DDL

「[データベース定義言語](#)」を参照してください。

ディープアンサンブル

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

深層学習

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

多層防御

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

委任管理者

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

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

トラブルシューティング

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

開発環境

「[環境](#)」を参照してください。

検出管理

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

開発バリューストリームマッピング (DVSM)

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

デジタルツイン

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

ディメンションテーブル

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

デザスタ

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

ディザスタリカバリ (DR)

[ディザスタ](#)によるダウンタイムとデータ損失を最小限に抑えるための戦略とプロセス。詳細については、AWS Well-Architected フレームワークの「[でのワークロードのディザスタリカバリ](#)」[AWS: クラウドでのリカバリ](#)」を参照してください。

DML

「[データベース操作言語](#)」を参照してください。

ドメイン駆動型設計

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

DR

「[ディザスタリカバリ](#)」を参照してください。

ドリフト検出

ベースライン設定からの偏差を追跡します。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、ガバナンス要件への準拠に影響する[ランディングゾーンの変更を検出](#)したりできます。

DVSM

「[開発バリューSTREAMマッピング](#)」を参照してください。

E

EDA

「[探索的データ分析](#)」を参照してください。

EDI

「[電子データ交換](#)」を参照してください。

エッジコンピューティング

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

電子データ交換 (EDI)

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

暗号化

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

暗号化キー

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

エンディアン

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

エンドポイント

[「サービスエンドポイント」](#)を参照してください。

エンドポイントサービス

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

エンタープライズリソースプランニング (ERP)

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

エンベロープ暗号化

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

環境

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

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

エピック

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

ERP

「[エンタープライズリソース計画](#)」を参照してください。

探索的データ分析 (EDA)

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

F

ファクトテーブル

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

フェイルファスト

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

障害分離境界

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

機能ブランチ

「[ブランチ](#)」を参照してください。

特徴量

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

特徴量重要度

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

機能変換

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

数ショットプロンプト

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

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

FGAC

「[きめ細かなアクセス制御](#)」を参照してください。

きめ細かなアクセス制御 (FGAC)

複数の条件を使用してアクセス要求を許可または拒否すること。

フラッシュカット移行

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

FM

「[基盤モデル](#)」を参照してください。

基盤モデル (FM)

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

G

生成 AI

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

ジオブロッキング

「[地理的制限](#)」を参照してください。

地理的制限 (ジオブロッキング)

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

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

Gitflow ワークフロー

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

ゴールデンイメージ

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

グリーンフィールド戦略

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

ガードレール

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

H

HA

「[高可用性](#)」を参照してください。

異種混在データベースの移行

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

高可用性 (HA)

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

ヒストリアンのモダナイゼーション

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

ホールドアウトデータ

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

同種データベースの移行

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

ホットデータ

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

ホットフィックス

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

ハイパーケア期間

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

|

IaC

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

ID ベースのポリシー

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

アイドル状態のアプリケーション

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

IIoT

「[インダストリアル IoT](#)」を参照してください。

イミュータブルインフラストラクチャ

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

インバウンド (受信) VPC

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

|

増分移行

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

インダストリー 4.0

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

インフラストラクチャ

アプリケーションの環境に含まれるすべてのリソースとアセット。

Infrastructure as Code (IaC)

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

インダストリアル IoT (IIoT)

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

インスペクション VPC

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

IoT

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

解釈可能性

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

IoT

[「IoT」](#)を参照してください。

IT 情報ライブラリ (ITIL)

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

IT サービス管理 (ITSM)

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

ITIL

[「IT 情報ライブラリ」](#)を参照してください。

ITSM

[「IT サービス管理」](#)を参照してください。

L

ラベルベースアクセス制御 (LBAC)

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

ランディングゾーン

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

大規模言語モデル (LLM)

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

大規模な移行

300 台以上のサーバの移行。

LBAC

「[ラベルベースアクセス制御](#)」を参照してください。

最小特権

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

リフトアンドシフト

「[7 Rs](#)」を参照してください。

リトルエンディアンシステム

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

LLM

「[大規模言語モデル](#)」を参照してください。

下位環境

「[環境](#)」を参照してください。

M

機械学習 (ML)

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

メインブランチ

「[ブランチ](#)」を参照してください。

マルウェア

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

マネージドサービス

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

製造実行システム (MES)

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

MAP

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

メカニズム

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

メンバーアカウント

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

MES

[「製造実行システム」](#) を参照してください。

Message Queuing Telemetry Transport (MQTT)

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

マイクロサービス

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

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

マイクロサービスアーキテクチャ

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

Migration Acceleration Program (MAP)

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

大規模な移行

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

移行ファクトリー

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

移行メタデータ

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

移行パターン

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

Migration Portfolio Assessment (MPA)

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

移行準備状況評価 (MRA)

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

移行戦略

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

ML

「[機械学習](#)」を参照してください。

モダナイゼーション

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

モダナイゼーション準備状況評価

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

モノリシックアプリケーション (モノリス)

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

MPA

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

MQTT

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

多クラス分類

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

ミュータブルなインフラストラクチャ

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

O

OAC

「[オリジンアクセス制御](#)」を参照してください。

OAI

「[オリジンアクセスアイデンティティ](#)」を参照してください。

OCM

「[組織変更管理](#)」を参照してください。

オフライン移行

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

OI

「[オペレーション統合](#)」を参照してください。

Ola

「[オペレーショナルレベルアグリーメント](#)」を参照してください。

オンライン移行

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

OPC-UA

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

Open Process Communications - Unified Architecture (OPC-UA)

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

オペレーショナルレベルアグリーメント (OLA)

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

運用準備状況レビュー (ORR)

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

運用テクノロジー (OT)

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

オペレーション統合 (OI)

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

組織の証跡

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

組織変更管理 (OCM)

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

オリジンアクセス制御 (OAC)

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

オリジンアクセスアイデンティティ (OAI)

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

ORR

「[運用準備状況レビュー](#)」を参照してください。

OT

「[運用テクノロジー](#)」を参照してください。

アウトバウンド (送信) VPC

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

P

アクセス許可の境界

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

個人を特定できる情報 (PII)

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

PII

「[個人を特定できる情報](#)」を参照してください。

プレイブック

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

PLC

「[プログラマブルロジックコントローラー](#)」を参照してください。

PLM

「[製品ライフサイクル管理](#)」を参照してください。

ポリシー

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

多言語の永続性

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

ポートフォリオ評価

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

述語

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

述語プッシュダウン

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

予防的コントロール

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

プリンシパル

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

プライバシーバイデザイン

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

プライベートホストゾーン

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

プロアクティブコントロール

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

製品ライフサイクル管理 (PLM)

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

本番環境

「[環境](#)」を参照してください。

プログラマブルロジックコントローラー (PLC)

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

プロンプトチェイニング

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

仮名化

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

発行/サブスクライブ (pub/sub)

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

Q

クエリプラン

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

クエリプランのリグレッション

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

R

RACI マトリックス

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

RAG

「[検索拡張生成](#)」を参照してください。

ランサムウェア

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

RASCI マトリックス

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

RCAC

「[行と列のアクセス制御](#)」を参照してください。

リードレプリカ

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

リアーキテクト

「[7 Rs](#)」を参照してください。

目標復旧時点 (RPO)

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

目標復旧時間 (RTO)

サービスが中断から復旧までの最大許容遅延時間。

リファクタリング

「[7 Rs](#)」を参照してください。

リージョン

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

リグレッション

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

リホスト

「[7 Rs](#)」を参照してください。

リリース

デプロイプロセスで、変更を本番環境に昇格させること。

再配置

「[7 Rs](#)」を参照してください。

リプラットフォーム

「[7 Rs](#)」を参照してください。

再購入

「[7 Rs](#)」を参照してください。

回復性

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

リソースベースのポリシー

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

実行責任者、説明責任者、協業先、報告先 (RACI) に基づくマトリックス

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

レスポンスコントロール

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

保持

「[7 Rs](#)」を参照してください。

廃止

「[7 Rs](#)」を参照してください。

検索拡張生成 (RAG)

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

ローテーション

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

行と列のアクセス制御 (RCAC)

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

RPO

「[目標復旧時点](#)」を参照してください。

RTO

「[目標復旧時間](#)」を参照してください。

ランブック

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

S

SAML 2.0

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

SCADA

「[監視制御とデータ取得](#)」を参照してください。

SCP

「[サービスコントロールポリシー](#)」を参照してください。

シークレット

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

セキュリティバイデザイン

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

セキュリティコントロール

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

セキュリティ強化

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

Security Information and Event Management (SIEM) システム

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

セキュリティレスポンスの自動化

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

サーバー側の暗号化

送信先で、それ AWS のサービスを受け取る によるデータの暗号化。

サービスコントロールポリシー (SCP)

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

サービスエンドポイント

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

サービスレベルアグリーメント (SLA)

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

サービスレベルインジケータ (SLI)

エラー率、可用性、スループットといった、サービスパフォーマンス面の指標。

サービスレベル目標 (SLO)

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

責任共有モデル

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

SIEM

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

単一障害点 (SPOF)

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

SLA

「[サービスレベルアグリーメント](#)」を参照してください。

SLI

「[サービスレベルインジケータ](#)」を参照してください。

SLO

「[サービスレベルの目標](#)」を参照してください。

スプリットアンドシードモデル

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

SPOF

「[単一障害点](#)」を参照してください。

スタースキーマ

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

strangler fig パターン

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

サブネット

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

監視制御とデータ取得 (SCADA)

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

対称暗号化

データの暗号化と復号に同じキーを使用する暗号化のアルゴリズム。

合成テスト

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

システムプロンプト

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

T

タグ

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

ターゲット変数

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

タスクリスト

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

テスト環境

「[環境](#)」を参照してください。

トレーニング

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

トランジットゲートウェイ

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

トランクベースのワークフロー

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

信頼されたアクセス

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

チューニング

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

ツーピザチーム

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

U

不確実性

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

未分化なタスク

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

上位環境

「[環境](#)」を参照してください。

V

バキューミング

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

バージョンコントロール

リポジトリ内のソースコードへの変更など、変更を追跡するプロセスとツール。

VPC ピアリング

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

脆弱性

システムのセキュリティを脅かすソフトウェアまたはハードウェアの欠陥。

W

ウォームキャッシュ

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

ウォームデータ

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

ウィンドウ関数

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

ワークロード

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

ワークストリーム

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

WORM

「[Write-Once-Read-Many](#)」を参照してください。

WQF

「[AWS ワークロード資格フレームワーク](#)」を参照してください

Write-Once-Read-Many (WORM)

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

Z

ゼロデイ 익스プロイト

[ゼロデイ脆弱性](#)を悪用した攻撃 (一般的にマルウェアによる)。

ゼロデイ脆弱性

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

ゼロショットプロンプト

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

ゾンビアプリケーション

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

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