



AWS 클라우드 마이그레이션을 위한 애플리케이션 포트폴리오 평가 가이드

# AWS 권장 가이드



---

# AWS 권장 가이드: AWS 클라우드 마이그레이션을 위한 애플리케이션 포트폴리오 평가 가이드

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

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

---

# Table of Contents

소개 .....	1
개요 .....	1
검색 가속화 및 초기 계획 .....	4
이 단계의 기본 결과 .....	4
초기 평가 데이터 요구 사항 이해 .....	4
데이터 소스 및 데이터 요구 사항 .....	4
검색 도구의 필요성 평가 .....	14
비즈니스 동인 및 기술 지침 원칙 .....	19
비즈니스 동인 .....	19
기술 지침 원칙 .....	20
데이터 수집 시작 .....	21
우선순위 지정 및 마이그레이션 전략 .....	23
애플리케이션 우선 순위 지정 .....	23
마이그레이션을 위한 R 유형 결정 .....	25
첨부 파일 .....	28
방향성 비즈니스 사례 생성 .....	28
방향성 비즈니스 사례의 범위 수정 .....	29
포커스 값 동인 .....	30
데이터 요구 사항 .....	30
인프라 TCO 비교 구축 .....	31
운영 비용 최적화 구축 .....	32
전체 방향성 비즈니스 사례로 확장 .....	34
마이그레이션 및 현대화 프로그램 설정 추정 .....	35
우선 순위가 지정된 애플리케이션 평가 .....	44
세부 평가 데이터 요구 사항 이해 .....	44
세부 애플리케이션 평가 .....	52
일반 .....	53
아키텍처 .....	53
운영 .....	54
성능 .....	54
소프트웨어 수명 주기 .....	54
마이그레이션 .....	55
복원력 .....	55
보안 및 규정 준수 .....	55

데이터베이스 수 .....	55
종속성 .....	56
AWS 애플리케이션 설계 및 마이그레이션 전략 .....	56
애플리케이션 미래 상태 .....	57
반복성 .....	58
요구 사항 .....	58
To-be 아키텍처 .....	58
아키텍처 결정 .....	61
소프트웨어 수명 주기 환경 .....	61
태그 지정 .....	61
마이그레이션 전략 .....	61
마이그레이션 패턴 및 도구 .....	62
서비스 관리 및 운영 .....	62
전환 고려 사항 .....	63
위험, 가정, 문제 및 종속성 .....	63
실행 비용 추정 .....	63
.....	64
전체 평가 데이터 요구 사항 이해 .....	64
애플리케이션 포트폴리오의 기준 설정 .....	73
우선 순위 지정 기준 반복 .....	74
6Rs 마이그레이션 전략 선택 반복 .....	77
웨이브 계획 .....	78
웨이브 플랜 생성 .....	79
변경 관리 .....	81
세부 비즈니스 사례 .....	82
사례에 필요한 시나리오 결정 .....	82
인프라 및 마이그레이션 비용 모델 검증 및 구체화 .....	83
IT 생산성 및 IT 운영 개선 및 효율성 가치 모델 지원 .....	84
복원력 가치 모델 개발 .....	90
비즈니스 민첩성 가치 모델 개발 .....	91
지속적인 평가 및 개선 .....	93
지속적 평가 데이터 요구 사항 이해 .....	93
세부 파도 평가 .....	94
최적화 및 현대화 평가 .....	94
웨이브 플랜 반복 .....	95
비즈니스 사례 개선 및 추적 .....	96

---

리소스 .....	97
문서 기록 .....	99
용어집 .....	100
# .....	100
A .....	101
B .....	103
C .....	105
D .....	108
E .....	112
F .....	114
G .....	115
H .....	116
I .....	118
L .....	120
M .....	121
O .....	125
P .....	127
Q .....	130
R .....	130
S .....	133
T .....	136
U .....	138
V .....	138
W .....	139
Z .....	140
.....	cxli

# AWS 클라우드 마이그레이션을 위한 애플리케이션 포트폴리오 평가 가이드

German Goncalves 및 Mark Berner, Amazon Web Services(AWS)

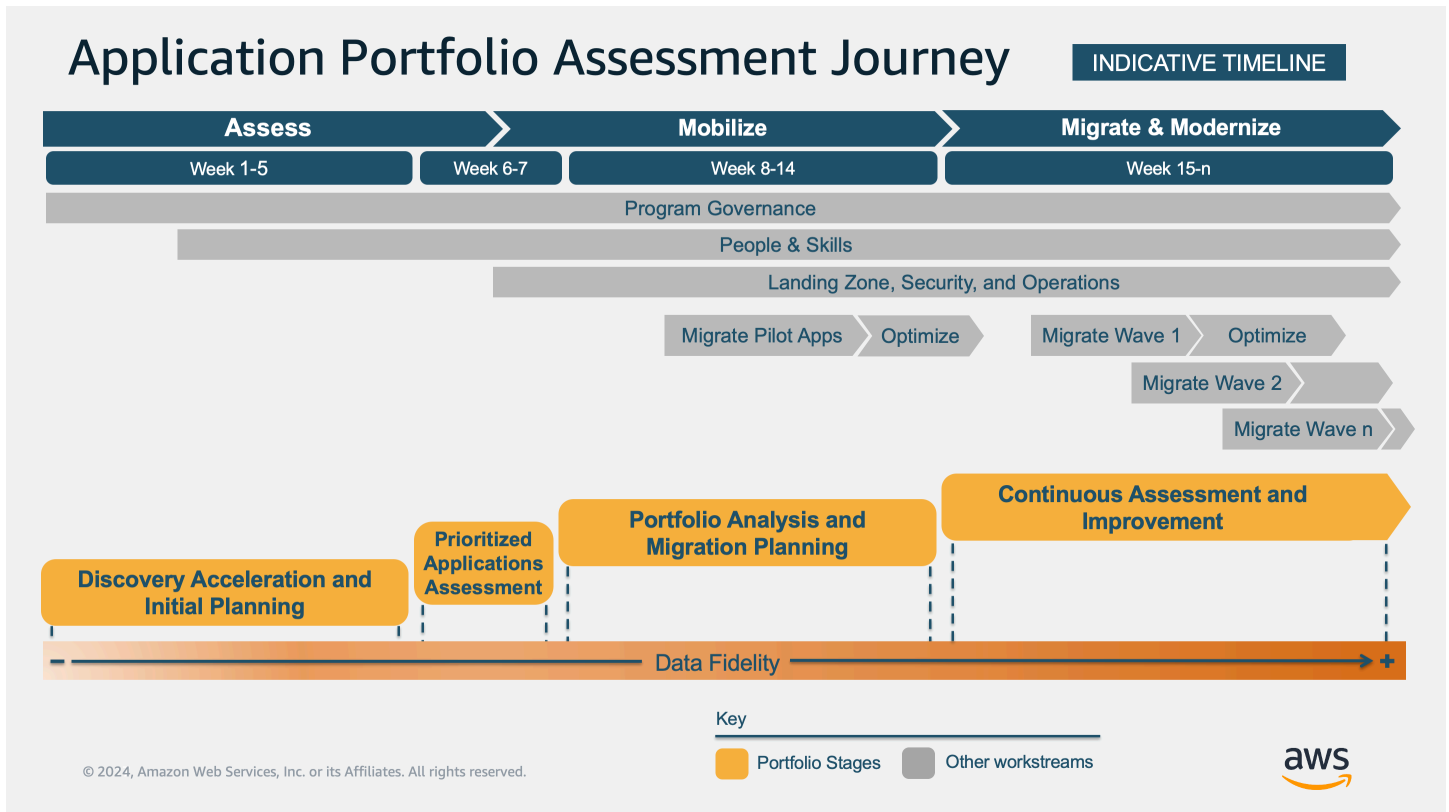
2024년 5월([문서 기록](#))

이 Amazon Web Services(AWS) 권장 가이드 문서는 [애플리케이션 포트폴리오 평가 전략](#)을 구현하는 방법을 자세히 설명합니다. 이 가이드를 사용하면 애플리케이션 포트폴리오 및 관련 인프라의 평가를 통해 시작하고 발전하는 데 도움이 될 수 있습니다. 평가에는 검색, 분석 및 계획이 포함됩니다. 인프라에는 컴퓨팅, 스토리지 및 네트워크가 포함됩니다.

## 개요

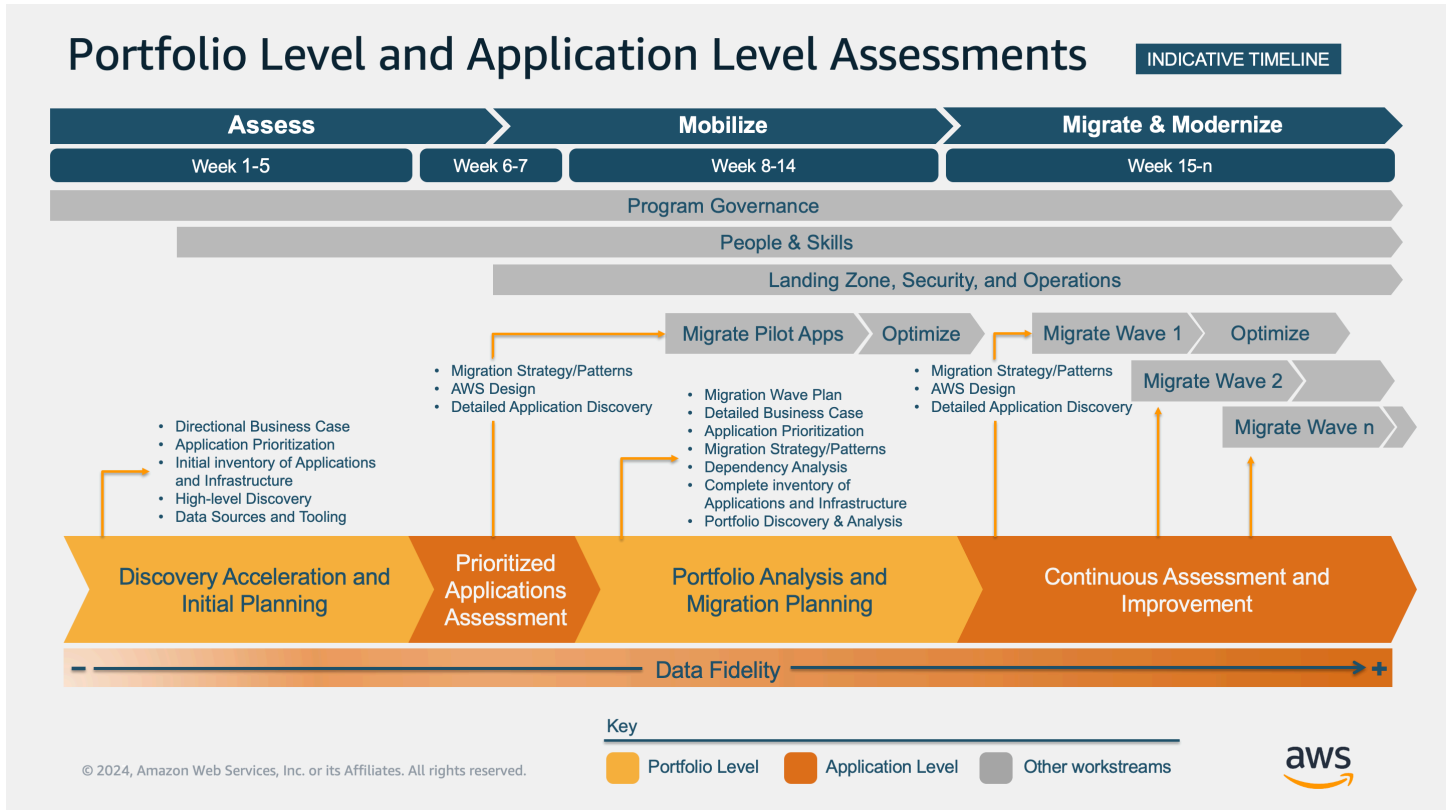
클라우드 마이그레이션 프로그램을 장기 실행하려면 프로그램 거버넌스, 랜딩 존(보안 제어 기능이 있는 운영 가능한 대상 환경), 마이그레이션 및 애플리케이션 포트폴리오와 같은 여러 워크스트림을 조정해야 합니다. 이러한 워크스트림의 이름은 마이그레이션 프로그램을 구성하는 방법에 따라 달라질 수 있습니다. 워크스트림으로서 애플리케이션 포트폴리오 평가는 이러한 프로그램의 전체 수명 주기에 걸친 기본 활동을 나타냅니다. 평가를 통해 얻은 포트폴리오를 이해하면 지속적인 애플리케이션 포트폴리오 평가로 인한 데이터 및 분석에 의존하는 다른 워크스트림에 대한 주요 입력이 제공됩니다.

다음 다이어그램은 포트폴리오 평가 단계가 마이그레이션 및 기타 워크스트림의 단계에 어떻게 대응하는 AWS 지 보여줍니다. 포트폴리오 검색 및 초기 계획 단계는 일반적으로 처음 5주 동안 평가 단계에서 시작됩니다. 6주 및 7주 동안의 우선 순위가 지정된 애플리케이션 평가는 평가 및 동원 단계에 걸쳐 있습니다. 포트폴리오 분석 및 마이그레이션 계획 단계는 동원 단계에서 8~14주 후에 수행됩니다. 지속적 평가 및 개선 단계는 마이그레이션 및 현대화 단계에서 15주부터 마이그레이션 프로그램 종료까지 진행됩니다. 이 타임라인은 이를 나타냅니다. 실제 단계 기간은 전체 프로그램 조직에 따라 달라집니다. 포트폴리오 평가 단계는이 프레임워크 외부에서도 유효하며 모든 마이그레이션 프로그램 구조에 통합할 수 있습니다.



- 검색 가속화 및 초기 계획은 포트폴리오에 대한 현재 이해에 중점을 둡니다. 여기에는 방향성 비즈니스 사례 생성, 마이그레이션을 위한 기본 합리화 모델 설정, 초기 마이그레이션 후보 식별이 포함됩니다.
- 우선순위가 지정된 애플리케이션 평가는 상세한 평가, 대상 상태 아키텍처의 초기 설계, 단기적으로 이동할 수 있는 애플리케이션 식별을 통해 더 빠른 time-to-value 제공합니다. 애플리케이션 이동은 팀에 마이그레이션 경험을 신속하게 제공하고 초기 랜딩 존 및 기타 인프라 구성 요소와 같은 클라우드 기반을 구축합니다.
- 포트폴리오 분석 및 마이그레이션 계획은 애플리케이션 포트폴리오에 대한 완전한 up-to-date 보기를 구축하는 데 중점을 둡니다. 이 뷰는 포트폴리오 데이터셋을 반복적으로 보강하고, 데이터 격차를 해소하고, 비즈니스 사례를 발전시키고, 신뢰도가 높은 마이그레이션 웨이브 계획을 수립하여 구축됩니다.
- 지속적인 평가 및 개선은 각 마이그레이션 웨이브에 대한 상세한 애플리케이션 및 기술 평가를 지속적인 활동으로 생성하여 대규모 마이그레이션을 지원합니다. 이 단계에는 마이그레이션 웨이브 계획을 반복하고 최적화 및 현대화를 위해 마이그레이션된 워크로드에 대한 추가 분석을 수행하는 작업이 포함됩니다.

다음 다이어그램은 각 평가 단계의 주요 활동과 이러한 활동이 포트폴리오 수준 평가와 애플리케이션 수준 평가 사이에서 어떻게 피벗되는지 보여줍니다. 포트폴리오 수준 평가는 포트폴리오의 상위 수준 검색 및 전체 분석에 중점을 둡니다. 예를 들어 포트폴리오 데이터, 애플리케이션 및 인프라 인벤토리, 우선순위 지정 및 방향성 비즈니스 사례의 소스입니다. 애플리케이션 수준 평가는 하나 이상의 애플리케이션을 자세히 검색하는 데 중점을 둡니다. 예를 들어 애플리케이션의 아키텍처 및 기술 수준에서 상세한 애플리케이션 검색, 대상 AWS 설계 및 마이그레이션 전략이 있습니다. 포트폴리오 수준 및 애플리케이션 수준 평가는 필요한 정보의 폭과 깊이를 나타냅니다.



## 검색 가속화 및 초기 계획

포트폴리오 평가의 첫 단계는 포트폴리오 수준에서 데이터를 얻고 분석하는 초기 단계에 중점을 둡니다. 주요 목표는 비즈니스 동인을 식별하고 애플리케이션 및 인프라에서 일반 데이터를 수집하여 포트폴리오의 초기 보기를 얻는 것입니다. 이 데이터에는 [데이터 요구 사항](#) 섹션에 설명된 대로 애플리케이션 이름, 환경, 제품 버전, 중요도, 성능 값 등과 같은 상위 수준의 기술 및 비즈니스 속성이 포함됩니다. 이 단계를 완료하는 것은 프로젝트의 범위를 이해하고, 초기 마이그레이션 후보를 식별하고, 비즈니스 사례를 알리는 데 중요합니다.

### 이 단계의 기본 결과

- 문서화된 비즈니스 동인, 성과, 목표 및 기술 지침 원칙.
- 애플리케이션 및 인프라의 초기 인벤토리 및 식별된 데이터 격차. 이는 추가 단계에서 반복되고 개선될 포트폴리오의 초기 보기입니다.
- 방향성 비즈니스 사례 및 마이그레이션 예상 비용.
- 초기 마이그레이션 후보 목록(예: 3-5개 애플리케이션).
- 다음 단계를 정의했습니다.

### 초기 평가 데이터 요구 사항 이해

데이터 수집에는 상당한 시간이 소요될 수 있으며 필요한 데이터와 필요한 시간에 대한 명확한 설명이 없는 경우 쉽게 차단할 수 있습니다. 핵심은 이 단계의 결과에 비해 너무 적은 데이터와 너무 많은 데이터의 균형을 이해하는 것입니다. 포트폴리오 평가의 초기 단계에 필요한 데이터와 충실도 수준에 초점을 맞추려면 데이터 수집에 대한 반복적인 접근 방식을 채택합니다.

### 데이터 소스 및 데이터 요구 사항

첫 번째 단계는 데이터 소스를 식별하는 것입니다. 먼저 데이터 요구 사항을 충족할 수 있는 조직 내 주요 이해관계자를 식별합니다. 이들은 일반적으로 서비스 관리, 운영, 용량 계획, 모니터링 및 지원 팀, 애플리케이션 소유자의 구성원입니다. 이러한 그룹의 구성원과 작업 세션을 설정합니다. 데이터 요구 사항을 전달하고 데이터를 제공할 수 있는 도구 및 기존 설명서 목록을 가져옵니다.

이러한 대화를 안내하려면 다음 질문 세트를 사용합니다.

- 현재 인프라 및 애플리케이션 인벤토리는 얼마나 정확하고 최신 상태입니까? 예를 들어 회사 구성 관리 데이터베이스(CMDB)의 경우 격차가 있는 위치를 이미 알고 있습니까?

- CMDB(또는 이에 상응하는)를 최신 상태로 유지하는 활성 도구 및 프로세스가 있습니까? 그렇다면 얼마나 자주 업데이트되나요? 최신 새로 고침 날짜는 언제입니까?
- CMDB와 같은 현재 인벤토리에 application-to-infrastructure 매핑이 포함되어 있습니까? 각 인프라 자산이 애플리케이션에 연결되어 있습니까? 각 애플리케이션이 인프라에 매핑되나요?
- 인벤토리에 각 제품의 라이선스 및 라이선스 계약 카탈로그가 포함되어 있습니까?
- 인벤토리에 종속성 데이터가 포함되어 있습니까? 서버 대 서버, 애플리케이션 대 애플리케이션, 애플리케이션 또는 서버 대 데이터베이스와 같은 통신 데이터가 있는지 확인합니다.
- 환경에서 애플리케이션 및 인프라 정보를 제공할 수 있는 다른 도구는 무엇입니까? 데이터 소스로 사용할 수 있는 성능, 모니터링 및 관리 도구가 있는지 확인합니다.
- 애플리케이션 및 인프라를 호스팅하는 데이터 센터와 같은 다양한 위치는 무엇입니까?

이러한 질문에 답변한 후 식별된 데이터 소스를 나열합니다. 그런 다음 각각에 충실도 수준 또는 신뢰 수준을 할당합니다. 도구와 같은 활성 프로그래밍 방식 소스에서 최근(30일 이내) 검증된 데이터는 충실도가 가장 높습니다. 정적 데이터는 충실도가 낮고 신뢰도가 낮은 것으로 간주됩니다. 정적 데이터의 예로는 문서, 통합 문서, 수동으로 업데이트된 CMDBs 또는 프로그래밍 방식으로 유지 관리되지 않는 기타 데이터 세트 또는 마지막 새로 고침 날짜가 60일 이상인 데이터 세트가 있습니다.

다음 표의 데이터 충실도 수준은 예제로 제공됩니다. 조직의 요구 사항을 가정에 대한 최대 허용 범위 및 관련 위험 측면에서 평가하여 적절한 수준의 충실도를 결정하는 것이 좋습니다. 표에서 제도적 지식은 문서화되지 않은 애플리케이션 및 인프라에 대한 모든 정보를 나타냅니다.

데이터 소스	충실도 수준	포트폴리오 적용 범위	설명
실험 지식	낮음 - 정확한 데이터의 최대 25%, 가정된 값 또는 데이터의 75%가 150일을 초과합니다.	낮음	중요한 애플리케이션에 초점을 맞춘 Scarce
지식 기반	중간 낮음 - 정확한 데이터의 35~40%, 가정된 값 또는 데이터의 65~60%는 120~150일입니다.	중간	수동으로 유지 관리되고 일관성 없는 세부 정보 수준

데이터 소스	충실도 수준	포트폴리오 적용 범위	설명
CMDB	중간 - 정확한 데이터의 ~50%, ~50% 가해진 값 또는 데이터는 90~120일입니다.	중간	혼합 소스의 데이터, 여러 데이터 격차 포함
VMware vCenter 내보내기	중간-높음 - 정확한 데이터의 75~80%, 가해진 값 또는 데이터의 25~20%는 60~90일입니다.	높음	가상화 자산의 90%를 차지합니다.
애플리케이션 성능 모니터링	높음 - 대부분 정확한 데이터, ~5% 가해진 값 또는 데이터는 0~60일 경과되었습니다.	낮음	중요한 프로덕션 시스템으로 제한됨(애플리케이션 포트폴리오의 15% 포함)

다음 표에는 각 자산 클래스(애플리케이션, 인프라, 네트워크 및 마이그레이션)의 필수 및 선택적 데이터 속성, 특정 활동(인벤토리 또는 비즈니스 사례),이 평가 단계에 권장되는 데이터 충실도가 나와 있습니다. 테이블은 다음 약어를 사용합니다.

- R, 필수
- (D), 방향성 비즈니스 사례의 경우 총 소유 비용(TCO) 비교 및 방향성 비즈니스 사례에 필요
- (F) 전체 방향성 비즈니스 사례의 경우 TCO 비교에 필요하고 마이그레이션 및 현대화 비용을 포함하는 방향성 비즈니스 사례
- O, 선택 사항의 경우
- 해당 없음, 해당 없음

### 애플리케이션

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
고유한 식별자	애플리케이션 ID 를 예로 들 수 있습니다. 일반적으로 기존 CMDBs 또는 기타 내부 인벤토리 및 제어 시스템에서 사용할 수 있습니다. 조직에 고유 ID가 정의되지 않은 경우 고유 IDs를 생성하는 것이 좋습니다.	R	R(D)	높음
애플리케이션 이름	이 애플리케이션이 조직에 알려진 이름입니다. 해당하는 경우 상용 off-the-shelf(COTS) 공급업체 및 제품 이름을 포함합니다.	R	R(D)	중간-높음
COTS입니까?	예 또는 아니요. 상용 애플리케이션인지 아니면 내부 개발인지 여부	R	R(D)	중간-높음
COTS 제품 및 버전	상용 소프트웨어 제품 이름 및 버전	R	R(D)	중간

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
설명	기본 애플리케이션 함수 및 컨텍스트	R	O	중간
중요도	예: 전략적 또는 수익 창출 애플리케이션 또는 중요한 함수 지원	R	O	중간-높음
Type	예: 데이터베이스, 고객 관계 관리(CRM), 웹 애플리케이션, 멀티미디어, IT 공유 서비스	R	O	중간
환경	예: 프로덕션, 사전 프로덕션, 개발, 테스트, 샌드박스	R	R(D)	중간-높음
규정 준수 및 규제	워크로드에 적용되는 프레임워크(예: HIPAA, SOX, PCI-DSS, ISO, SOC, FedRAMP) 및 규제 요구 사항	R	R(D)	중간-높음

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
종속성	내부 및 외부 애플리케이션 또는 서비스에 대한 업스트림 및 다운스트림 종속성. 운영 요소와 같은 비기술적 종속성 (예: 유지 관리 주기)	O	O	중간-낮음
인프라 매핑	애플리케이션을 구성하는 물리적 및/또는 가상 자산에 매핑	O	O	중간
라이선스	상품 소프트웨어 라이선스 유형(예: Microsoft SQL Server Enterprise)	O	R	중간-높음
비용	소프트웨어 라이선스, 소프트웨어 운영 및 유지 관리 비용	해당 사항 없음	O	중간
<b>인프라</b>				
속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
고유한 식별자	서버 ID를 예로 들 수 있습니다. 일반적으로 기존	R	R	높음

	CMDBs 또는 기타 내부 인벤토리 및 제어 시스템에서 사용할 수 있습니다. 조직에 고유 ID가 정의되지 않은 경우 고유 IDs를 생성하는 것이 좋습니다.			
네트워크 이름	네트워크의 자산 이름(예: 호스트 이름)	R	O	중간-높음
DNS 이름(정규화된 도메인 이름 또는 FQDN)	DNS 이름	O	O	중간
IP 주소 및 넷마스크	내부 및/또는 퍼블릭 IP 주소	R	O	중간-높음
애셋 유형	물리적 또는 가상 서버, 하이퍼바이저, 컨테이너, 디바이스, 데이터베이스 인스턴스 등	R	R	중간-높음
제품 이름	상용 공급업체 및 제품 이름(예: VMware ESXi, IBM Power Systems, Exadata)	R	R	중간
운영 체제	예: REHL 8, Windows Server 2019, AIX 6.1	R	R	중간-높음

구성	할당된 CPU, 코어 수, 코어당 스퀘드 수, 총 메모리, 스토리지, 네트워크 카드	R	R	중간-높음
사용률	CPU, 메모리, 스토리지 피크 및 평균. 데이터베이스 인스턴스 처리량.	R	O	중간-높음
라이선스	상품 라이선스 유형(예: RHEL Standard)	R	R	중간
공유 인프라입니까?	인증 공급자, 모니터링 시스템, 백업 서비스 및 유사한 서비스와 같은 공유 서비스를 제공하는 인프라 서비스를 나타내는 예 또는 아니요	R	R(D)	중간
애플리케이션 매핑	이 인프라에서 실행되는 애플리케이션 또는 애플리케이션 구성 요소	O	O	중간

비용	하드웨어, 유지 관리, 운영, 스토리지(SAN, NAS, 객체), 운영 체제 라이선스, 랙스페이스 공유, 데이터 센터 오버헤드 등 베어 메탈 서버의 전체 로드 비용	해당 사항 없음	O	중간-높음
----	---	----------	---	-------

네트워크

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
파이프 크기(Mb/s), 중복(Y/N)	현재 WAN 링크 사양(예: 1000Mb/s 중복)	O	R	중간
링크 사용률	최대 및 평균 사용률, 아웃바운드 데이터 전송(GB/월)	O	R	중간
지연 시간(ms)	연결된 위치 간의 현재 지연 시간입니다.	O	O	중간
비용	현재 월별 비용	해당 사항 없음	O	중간

마이그레이션

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
-------	----	-----------------	---------	----------------

리호스팅	각 워크로드(인 일), 일일 고객 및 파트너 비용 비율, 도구 비용, 워크로드 수에 대한 고객 및 파트너 작업	해당 사항 없음	R(F)	중간-높음
리플랫폼	각 워크로드(인 일), 일일 고객 및 파트너 비용 비율, 워크로드 수에 대한 고객 및 파트너 노력	해당 사항 없음	R(F)	중간-높음
리팩터링	각 워크로드(인 일), 일일 고객 및 파트너 비용 비율, 워크로드 수에 대한 고객 및 파트너 노력	해당 사항 없음	O	중간-높음
만료	서버 수, 평균 폐기 비용	해당 사항 없음	O	중간-높음
랜딩 존	기존 재사용(Y/N), 필요한 AWS 리전 목록, 비용	해당 사항 없음	R(F)	중간-높음
사람 및 변화	클라우드 운영 및 개발에서 훈련할 직원 수, 1명당 훈련 비용, 1명당 훈련 시간 비용	해당 사항 없음	R(F)	중간-높음
지속 시간	범위 내 워크로드 마이그레이션 기간(개월)	O	R(F)	중간-높음

병렬 비용	마이그레이션 중에 있는 그대로 비용을 제거할 수 있는 기간 및 비율	해당 사항 없음	O	중간-높음
	마이그레이션 중에 AWS 제품 및 서비스와 기타 인프라 비용이 도입 되는 기간 및 속도	해당 사항 없음	O	중간-높음

## 검색 도구의 필요성 평가

조직에 검색 도구가 필요합니까? 포트폴리오 평가에는 애플리케이션 및 인프라에 대한 높은 신뢰도의 up-to-date 데이터가 필요합니다. 포트폴리오 평가의 초기 단계에서는 가정을 사용하여 데이터 격차를 해소할 수 있습니다.

그러나 진행 상황이 진행됨에 따라 충실도가 높은 데이터를 통해 성공적인 마이그레이션 계획을 수립하고 대상 인프라를 올바르게 추정하여 비용을 절감하고 이점을 극대화할 수 있습니다. 또한 종속성을 고려하고 마이그레이션 위험을 방지하는 구현을 활성화하여 위험을 줄입니다. 클라우드 마이그레이션 프로그램에서 검색 도구의 주요 사용 사례는 다음을 통해 위험을 줄이고 데이터의 신뢰도 수준을 높이는 것입니다.

- 자동 또는 프로그래밍 방식 데이터 수집으로 검증되고 신뢰도가 높은 데이터 생성
- 데이터 획득 속도 가속화, 프로젝트 속도 향상 및 비용 절감
- CMDBs에서 일반적으로 사용할 수 없는 통신 데이터 및 종속성을 포함하여 데이터 완전성 수준 향상
- 자동화된 애플리케이션 식별, TCO 분석, 예상 실행률 및 최적화 권장 사항과 같은 인사이트 확보
- 신뢰도가 높은 마이그레이션 웨이브 계획

시스템이 특정 위치에 있는지 확실하지 않은 경우 대부분의 검색 도구는 네트워크 서브넷을 스캔하고 ping 또는 Simple Network Management Protocol(SNMP) 요청에 응답하는 시스템을 검색할 수 있습니다. 모든 네트워크 또는 시스템 구성이 ping 또는 SNMP 트래픽을 허용하는 것은 아닙니다. 네트워크 및 기술 팀과 이러한 옵션에 대해 논의합니다.

애플리케이션 포트폴리오 평가 및 마이그레이션의 추가 단계는 정확한 종속성 매핑 정보에 크게 의존합니다. 종속성 매핑은 AWS 필요한 인프라 및 구성(예: 보안 그룹, 인스턴스 유형, 계정 배치 및 네트워크 라우팅)에 대한 이해를 제공합니다. 또한 동시에 이동해야 하는 애플리케이션(예: 짧은 지연 시간 네트워크를 통해 통신해야 하는 애플리케이션)을 그룹화하는 데 도움이 됩니다. 또한 종속성 매핑은 비즈니스 사례를 발전시키기 위한 정보를 제공합니다.

검색 도구를 결정할 때는 평가 프로세스의 모든 단계를 고려하고 데이터 요구 사항을 예측하는 것이 중요합니다. 데이터 격차는 차단 요소가 될 가능성이 있으므로 향후 데이터 요구 사항과 데이터 소스를 분석하여 이를 예측하는 것이 중요합니다. 필드에서의 경험에 따르면 대부분의 중단된 마이그레이션 프로젝트에는 범위 내 애플리케이션, 관련 인프라 및 종속성이 명확하게 식별되지 않는 제한된 데이터 세트가 있습니다. 이러한 식별 부족으로 인해 잘못된 지표, 결정 및 지연이 발생할 수 있습니다. up-to-date 데이터를 얻는 것이 성공적인 마이그레이션 프로젝트의 첫 번째 단계입니다.

검색 도구를 선택하려면 어떻게 해야 하나요?

시장 내 여러 검색 도구는 다양한 기능을 제공합니다. 요구 사항을 고려하세요. 그리고 조직에 가장 적합한 옵션을 결정합니다. 마이그레이션을 위한 검색 도구를 결정할 때 가장 일반적인 요소는 다음과 같습니다.

## 보안

- 도구 데이터 리포지토리 또는 분석 엔진에 액세스하는 인증 방법은 무엇입니까?
- 누가 데이터에 액세스할 수 있으며 도구에 액세스하기 위한 보안 제어는 무엇입니까?
- 도구는 어떻게 데이터를 수집하나요? 전용 자격 증명이 필요합니까?
- 도구를 사용하여 시스템에 액세스하고 데이터를 얻으려면 어떤 자격 증명과 액세스 수준이 필요합니까?
- 도구 구성 요소 간에 데이터가 어떻게 전송되나요?
- 도구가 저장 데이터 암호화 및 전송 중 데이터 암호화를 지원하나요?
- 내 환경 내부 또는 외부의 단일 구성 요소에 데이터가 중앙 집중화되어 있습니까?
- 네트워크 및 방화벽 요구 사항은 무엇입니까?

보안 팀이 검색 도구에 대한 초기 대화에 참여하고 있는지 확인합니다.

## 데이터 주권

- 데이터는 어디에 저장되고 처리되나요?
- 도구에서 서비스형 소프트웨어(SaaS) 모델을 사용하나요?

- 내 환경의 경계 내에 모든 데이터를 유지할 수 있나요?
- 내 조직의 경계를 벗어나기 전에 데이터를 검사할 수 있나요?

데이터 레지던시 요구 사항 측면에서 조직의 요구 사항을 고려합니다.

### 아키텍처

- 필요한 인프라와 다양한 구성 요소는 무엇입니까?
- 둘 이상의 아키텍처를 사용할 수 있나요?
- 도구가 공기 잠금 보안 영역에 구성 요소 설치를 지원하나요?

### 성능

- 데이터 수집이 시스템에 미치는 영향은 무엇인가요?

### 호환성 및 범위

- 도구가 내 제품 및 버전의 전체 또는 대부분을 지원하나요? 도구 설명서를 검토하여 범위에 대한 현재 정보를 기준으로 지원되는 플랫폼을 확인합니다.
- 대부분의 운영 체제가 데이터 수집을 지원하나요? 운영 체제 버전을 모르는 경우 검색 도구 목록을 지원하는 시스템의 범위가 더 넓은 것으로 좁히십시오.

### 수집 방법

- 도구에서 각 대상 시스템에 에이전트를 설치해야 합니까?
- 에이전트 없는 배포를 지원하나요?
- 에이전트와 에이전트가 없는 도 동일한 기능을 제공하나요?
- 수집 프로세스란 무엇입니까?

### Features

- 사용 가능한 기능은 무엇입니까?
- 총 소유 비용(TCO)과 예상 AWS 클라우드 실행률을 계산할 수 있나요?
- 마이그레이션 계획을 지원하나요?
- 성능을 측정하나요?

- 대상 AWS 인프라를 추천할 수 있나요?
- 종속성 매핑을 수행하나요?
- 어떤 수준의 종속성 매핑을 제공하나요?
- API 액세스를 제공하는가?(예: 데이터를 얻기 위해 프로그래밍 방식으로 액세스할 수 있는가?)

강력한 애플리케이션 및 인프라 종속성 매핑 함수가 있는 도구와 통신 패턴에서 애플리케이션을 유추할 수 있는 도구를 고려합니다.

## 비용

- 라이선스 모델이란 무엇입니까?
- 라이선스 비용은 얼마인가요?
- 각 서버의 요금인가요? 계층형 요금인가요?
- 온디맨드 방식으로 라이선스를 부여할 수 있는 기능이 제한된 옵션이 있나요?

검색 도구는 일반적으로 마이그레이션 프로젝트의 전체 수명 주기 동안 사용됩니다. 예산이 제한된 경우 최소 6개월을 고려하세요. 그러나 검색 도구가 없으면 일반적으로 수동 작업과 내부 비용이 증가합니다.

## 지원 모델

- 기본적으로 제공되는 지원 수준은 무엇입니까?
- 지원 플랜을 사용할 수 있나요?
- 인시던트 대응 시간은 어떻게 됩니까?

## 전문 서비스

- 공급업체는 검색 결과를 분석하기 위한 전문 서비스를 제공하니까?
- 이 가이드의 요소를 다룰 수 있나요?
- 도구 + 서비스에 대한 할인 또는 번들이 있나요?

### Tip

검색 도구를 찾고 평가하려면 [검색, 계획 및 권장 사항](#) 사이트를 사용합니다.

## 검색 도구에 권장되는 기능

시간이 지남에 따라 여러 도구에서 데이터를 프로비저닝하고 결합하는 것을 방지하려면 검색 도구에 서 다음과 같은 최소 기능을 포함해야 합니다.

- 소프트웨어 - 검색 도구는 실행 중인 프로세스와 설치된 소프트웨어를 식별할 수 있어야 합니다.
- 종속성 매핑 - 네트워크 연결 정보를 수집하고 서버 및 실행 중인 애플리케이션의 인바운드 및 아웃바운드 종속성 맵을 구축할 수 있어야 합니다. 또한 검색 도구는 통신 패턴을 기반으로 인프라 그룹에서 애플리케이션을 추론할 수 있어야 합니다.
- 프로파일 및 구성 검색 - CPU 패밀리(예: x86, PowerPC), CPU 코어 수, 메모리 크기, 디스크 수 및 크기, 네트워크 인터페이스와 같은 인프라 프로파일을 보고할 수 있어야 합니다.
- 네트워크 스토리지 검색 - 네트워크 연결 스토리지(NAS)에서 네트워크 공유를 감지하고 프로파일링할 수 있어야 합니다.
- 성능 - CPU, 메모리, 디스크 및 네트워크의 최대 및 평균 사용률을 보고할 수 있어야 합니다.
- 격차 분석 - 데이터 수량 및 충실도에 대한 인사이트를 제공할 수 있어야 합니다.
- 네트워크 스캔 - 네트워크 서브넷을 스캔하고 알 수 없는 인프라 자산을 검색할 수 있어야 합니다.
- 보고 - 수집 및 분석 상태를 제공할 수 있어야 합니다.
- API 액세스 - 수집된 데이터에 액세스할 수 있는 프로그래밍 방식을 제공할 수 있어야 합니다.

## 고려해야 할 추가 기능

- 현재 온프레미스 비용과 예상 비용 간의 AWS 비용 비교를 제공하는 TCO 분석.
- 리호스팅 및 리플랫폼 시나리오에서 Microsoft SQL Server 및 Oracle 시스템에 대한 라이선스 분석 및 최적화 권장 사항입니다.
- 마이그레이션 전략 권장 사항(검색 도구가 현재 기술을 기반으로 기본 마이그레이션 R 유형 권장 사항을 만들 수 있습니까?)
- 인벤토리 내보내기(CSV 또는 유사한 형식으로)
- 적절한 크기 권장 사항(예: 권장 대상 AWS 인프라를 매핑할 수 있습니까?)
- 종속성 시각화(예: 종속성 매핑을 그래픽 모드로 시각화할 수 있습니까?)
- 아키텍처 보기(예: 아키텍처 다이어그램을 자동으로 생성할 수 있습니까?)
- 애플리케이션 우선 순위 지정( 마이그레이션을 위한 우선 순위 지정 기준을 생성하기 위해 애플리케이션 및 인프라 속성에 가중치 또는 관련성을 할당할 수 있습니까?)
- 웨이브 계획(예: 권장 애플리케이션 그룹 및 마이그레이션 웨이브 계획 생성 기능)
- 마이그레이션 비용 추정(마이그레이션 작업 추정)

## 배포 고려 사항

검색 도구를 선택하고 조달한 후에는 다음 질문을 고려하여 조직에 도구 배포를 담당하는 팀과 대화를 나누세요.

- 서버 또는 애플리케이션을 타사에서 운영하나요? 이로 인해 팀이 참여하고 따라야 할 프로세스가 필요할 수 있습니다.
- 검색 도구 배포 승인을 얻기 위한 상위 수준 프로세스는 무엇입니까?
- 서버, 컨테이너, 스토리지 및 데이터베이스와 같은 시스템에 액세스하기 위한 주요 인증 프로세스는 무엇입니까? 서버 자격 증명은 로컬입니까, 아니면 중앙 집중식입니까? 자격 증명을 얻는 프로세스는 무엇입니까? 시스템(예: 컨테이너, 가상 또는 물리적 서버, 하이퍼바이저 및 데이터베이스)에서 데이터를 수집하려면 자격 증명에 필요합니다. 각 자산에 연결하기 위해 검색 도구의 자격 증명을 얻는 것은 어려울 수 있습니다. 특히 이러한 자산이 중앙 집중화되지 않은 경우 더욱 그렇습니다.
- 네트워크 보안 영역 개요란 무엇입니까? 네트워크 다이어그램을 사용할 수 있나요?
- 데이터 센터에서 방화벽 규칙을 요청하는 프로세스는 무엇입니까?
- 데이터 센터 운영(검색 도구 설치, 방화벽 요청)과 관련된 현재 지원 서비스 수준 계약(SLAs)은 무엇입니까?

## 비즈니스 동인 및 기술 지침 원칙

### 비즈니스 동인

조직이 이미 클라우드로 전환하기로 결정했는지 또는 해당 결정에 근접했는지 여부에 관계없이 클라우드 마이그레이션을 위한 비즈니스 동인을 정의하고 문서화하면 마이그레이션 이유를 명확히 할 수 있습니다. 이유가 문서화된 후 마이그레이션할 대상과 마이그레이션 방법을 정의할 수 있습니다. 이 활동은 중요합니다. 다음 단계를 알리고 안내하려면 프로세스 초기에 수행하는 것이 좋습니다.

토론에 참여해야 하는 이해관계자를 식별하여 동인을 문서화합니다. 일반적으로 CxOs, 고위 관리자, 조직 내 주요 기술 리더 및 자체 고객. 고객이이 토론에 참여할 가능성은 낮지만 조직의 한 명 이상의 사람이 지정되어 고객의 관점과 목표를 나타내는 것이 좋습니다.

비즈니스 동인은 마이그레이션 여정 전체에서 측정할 수 있는 지표에 연결되어 결과가 달성되었는지 확인해야 합니다. 회사의 전략적 목표와 연간 보고서는 출발점 역할을 할 수 있습니다.

클라우드로 전환한 결과로 기존 지표와 예상 지표를 기반으로 회사가 원하는 위치에 대한 대화에 집중합니다. 목표와 비즈니스 성과를 고려합니다. 또한 클라우드 채택이 증가함에 따라 성공이 어떤 모습인지 생각해 보세요.

그런 다음 각 드라이버의 중요도 수준을 설정합니다. 우선 순위는 무엇입니까? 예상되는 이점은 무엇입니까? 이점이 비즈니스 목표와 성과를 어떻게 지원하나요? 애플리케이션 포트폴리오 평가의 맥락에서 답변은 마이그레이션을 위한 워크로드의 우선순위를 지정하고 기술 가이드 원칙을 설정하는 데 도움이 됩니다. 그러나 비즈니스 동인은 마이그레이션 프로그램 전체를 정의하고 영향을 미칩니다.

## 기술 지침 원칙

기술 지침 원칙은 포트폴리오 평가의 후반부 단계에서 마이그레이션 전략 선택을 안내합니다. 현재 단계에서는 이를 식별하는 데 중점을 둡니다.

기본 원칙은 비즈니스 목표 및 성과에서 파생된 일반적인 기술 관련 및 접근 방식 관련 결정으로 설정할 수 있습니다.

예를 들어 회사는 비용을 절감하는 주요 목표를 가지고 있으며, 원하는 성과는 6~12개월 내에 지정된 날짜까지 온프레미스 데이터 센터를 닫는 것입니다. 그에 따른 기본 원칙은 가능하면 리호스팅 또는 재배치 마이그레이션 전략을 사용하여 모든 애플리케이션을 클라우드로 리프트 앤드 시프트하는 것입니다. 이 경우 lift-and-shift 접근 방식은 단기 마이그레이션 결과를 가속화합니다. 애플리케이션이 온프레미스 데이터 센터 밖으로 이동한 후 회사는 마이그레이션된 워크로드를 최적화하거나 현대화하기 위해 주요 비즈니스 동인에 집중할 수 있습니다.

기술 지침 원칙을 설정하려면 먼저 비즈니스 동인을 분석합니다. 비즈니스 목표 및 성과를 달성할 수 있는 기술 및 기법 목록을 식별합니다. 그런 다음 목록을 구체화하고 적합성 또는 선호도에 따라 관련성 순서를 할당하여 원하는 결과를 달성합니다.

마이그레이션을 계획하고 수행하는 데 관련된 사람들과 지침 원칙을 문서화하고 전달합니다. 원칙과 실제 구현 간의 우려 사항 및 잠재적 충돌을 강조합니다.

다음 표에는 비즈니스 동인과 기술 지침 원칙의 예가 나와 있습니다.

비즈니스 드라이버	결과	Metrics	기술 지침 원칙
혁신을 가속화합니다.	경쟁력 향상, 비즈니스 민첩성 향상	일별 또는 월별 배포 수, 분기당 릴리스된 새로운 기능, 고객 만족도 점수, 실험 수	마이크로서비스와 DevOps 운영 모델을 사용하여 차별화된 애플리케이션을 리팩터링하여 새로운 기능의 민첩성과 출시 속도를 높입니다.

비즈니스 드라이버	결과	Metrics	기술 지침 원칙
운영 및 인프라 비용을 절감합니다.	공급 및 수요 일치, 탄력적 비용 기반(사용한 만큼 지불)	시간 경과에 따른 지출 변동	<ol style="list-style-type: none"> <li>1. 인프라 적정 크기로 애플리케이션을 리호스팅합니다.</li> <li>2. 사용률이 낮거나 없는 애플리케이션은 사용 중지합니다.</li> </ol>
운영 복원력을 높입니다.	가동 시간 개선, 평균 복구 시간 단축	SLAs, 인시던트 수	<ol style="list-style-type: none"> <li>1. 애플리케이션을 가장 잘 지원되는 최신 운영 체제 버전으로 리플랫폼합니다.</li> <li>2. 중요한 애플리케이션을 위한고가용성 아키텍처를 구현합니다.</li> </ol>
데이터 센터를 종료합니다.	6~12개월 이내 날짜별 데이터 센터 폐쇄	서버 마이그레이션 속도	Cloud Migration Factory 솔루션을 사용하여 애플리케이션을 리호스팅합니다.
온프레미스에서는 유지하되 민첩성과 복원력을 높입니다.	온프레미스에 남아 있는 동안 경쟁 및 가동 시간 개선	일별 또는 월별 배포 수, 분기당 새로운 기능 릴리스, SLAs, 인시던트 수	<ol style="list-style-type: none"> <li>1. 기능을 클라우드로 확장하여 시스템을 현대화합니다.</li> <li>2. 리호스팅 또는 리플랫폼을 평가합니다 AWS Outposts.</li> </ol>

## 데이터 수집 시작

데이터 수집은 애플리케이션 및 인프라에서 메타데이터를 수집하는 프로세스입니다. 프로세스는 평가의 모든 단계에서 반복적입니다. 각 단계에서 데이터 수량과 충실도가 증가합니다. 이 단계에서는 초기 인벤토리를 설정하는 데 도움이 될 수 있는 일반 데이터를 수집하는 데 중점을 둡니다. 인벤토리는 방향성 비즈니스 사례를 생성하고 초기 마이그레이션 후보를 식별하는 데 사용됩니다.

현재 데이터 소스를 식별한 후에는 가능한 한 많은 시스템에서 정보를 수집하는 것이 좋습니다. 자세한 내용은 이 단계의 [데이터 요구 사항을](#) 참조하세요.

이 접근 방식은 현재 포트폴리오 보기와 애플리케이션 및 서비스에 대한 조직의 지식을 업데이트하는 데 도움이 되는 이점이 있습니다. 또한 이동할 대상을 결정하는 데도 도움이 됩니다. 권장 접근 방식은 구성 관리 데이터베이스(CMDB) 출력 및 정보 기술 서비스 관리(ITSM) 시스템과 같은 기존 데이터를 검토하는 것입니다. 그런 다음 데이터 수집을 대상으로 하는 자산 목록을 구성합니다. 조직에서 마이그레이션 범위 내에 있고 범위를 벗어난 항목을 완전히 명확히 파악한 경우 데이터 수집을 범위 내에 있는 시스템으로 제한할 수 있습니다.

포트폴리오를 구축할 때 애플리케이션과 해당 환경 또는 소프트웨어 릴리스 수명 주기를 고려하세요. 예를 들어 고객 관계 관리(CRM) 애플리케이션을 식별하고 테스트, 개발 및 생산 환경이 있음을 지정하는 대신 세 가지 애플리케이션(예: CRM-Test, CRM-Dev, CRM-Prod)을 나열합니다. 또는 CRM 이름을 사용하지만 각 환경에 고유한 ID를 할당하고 데이터 리포지토리에 별도의 레코드로 제공합니다. 이렇게 하면 이러한 환경의 마이그레이션을 개별적으로 계획하고 추적하는 데 도움이 됩니다. 예를 들어 비프로덕션 환경을 먼저 마이그레이션할 수 있습니다. 환경에 따라 애플리케이션 인스턴스를 나열하면 전환을 명확하게 관리하고 관리할 수 있습니다.

데이터 수집 중에 특정 데이터 센터 또는 소스 위치에 있는 애플리케이션 또는 서버에 대한 불확실성이 있을 수 있습니다. 이러한 경우 기존 관리 도구에서 베어 메탈 및 하이퍼바이저 목록을 가져오는 것이 좋습니다. 예를 들어 하이퍼바이저에 연결하여 데이터 수집 대상 가상 머신 목록을 가져올 수 있습니다.

기존 데이터 소스를 결합할 때 초기 출력이 불완전할 수 있습니다. 핵심은 이 단계의 [데이터 요구 사항](#)과 기존 소스에서 얻을 수 있는 내용 측면에서 격차 분석을 수행하는 것입니다. 완전성의 백분율을 데이터 충실도 수준과 대조하는 것이 중요합니다. 충실도가 낮은 소스의 완전성 수준이 높을수록 분석 결합으로 이어질 수 있는 몇 가지 가정이 포함됩니다. 이 평가 단계에서는 최대 데이터 충실도가 필요하지 않지만 데이터 소스는 최소한 중간에서 중간까지 충실도가 높은 것이 좋습니다. 이러한 수치와 데이터 격차를 줄이기 위한 가정 사용을 포함하여 조직의 위험 허용 오차를 비교합니다.

격차 분석은 작업 중인 데이터의 양과 품질을 이해하는 데 도움이 됩니다. 또한 분석을 통해 방향성 비즈니스 사례를 생성하고 마이그레이션을 위한 애플리케이션의 우선순위를 정하기 위해 수행해야 하는 가정 수준을 설정할 수 있습니다. 검색 도구는 격차를 해소하고 충실도가 높은 데이터를 수집하는 데 도움이 될 수 있습니다. 데이터의 신뢰도 수준을 높이고 마이그레이션 결과를 가속화하려면 최대한 빨리 검색 도구를 배포하는 것이 좋습니다. 새 도구의 내부 조달, 보안 및 구현 프로세스를 완료하는 데 몇 주 또는 몇 달이 걸릴 수 있으므로 초기 조치도 중요합니다.

이 단계에서는 커뮤니케이션 계획 또는 주기와 범위 변경 제어 메커니즘을 설정하는 것이 좋습니다. 이를 통해 이해관계자가 미리 계획을 세우고 위험을 완화할 수 있도록 이해관계자에게 최신 정보를 제공

할 수 있습니다. 명확한 커뮤니케이션의 핵심 요소는 애플리케이션 포트폴리오 및 관련 인프라에 대한 단일 정보 소스를 정의하는 것입니다. 여러 개의 레코드 시스템, 애플리케이션 및 인프라 목록을 보관하지 마세요. 버전 관리 및 온라인 공동 작업을 지원하는 데이터(예: 데이터베이스, 도구 또는 스프레드 시트)를 한 곳에 보관하고 소유자를 할당합니다.

## 우선순위 지정 및 마이그레이션 전략

마이그레이션 계획의 핵심 요소는 우선순위 기준을 설정하는 것입니다. 이 연습의 요점은 애플리케이션을 마이그레이션할 순서를 이해하는 것입니다. 전략은 우선순위 지정 모델을 발전시키기 위해 반복적이고 점진적인 접근 방식을 취하는 것입니다.

### 애플리케이션 우선 순위 지정

이 평가 단계는 저위험 및 저복잡성 워크로드의 우선 순위를 정하기 위한 초기 기준을 설정하는 데 중점을 둡니다. 이러한 워크로드는 파일럿 애플리케이션에 적합합니다. 초기 마이그레이션에서 위험이 낮고 복잡성이 낮은 워크로드를 사용하면 위험이 줄어들고 팀이 경험을 얻을 수 있는 기회가 제공됩니다. 이러한 기준은 마이그레이션 웨이브 플랜을 생성할 때 비즈니스 동인에 우선순위를 맞추기 위해 평가 단계에서 발전할 것입니다.

초기 기준은 클라우드 지원 인프라와 비프로덕션 환경에서 실행되는 소수의 종속성이 있는 애플리케이션의 우선 순위를 지정해야 합니다. 예를 들어 개발 또는 테스트 환경에서 있는 그대로 리호스팅할 준비가 된 종속성이 0~3개인 애플리케이션이 있습니다. 이러한 기준은 클라우드 채택 성숙도 및 신뢰도 수준에 따라 파일럿 애플리케이션과 잠재적으로 첫 번째 및 두 번째 마이그레이션 웨이브를 정의하는 데 유효합니다.

#### 사용할 초기 기준 결정

첫 번째 워크로드의 우선 순위를 지정하는 데 사용할 데이터 포인트를 2~10개 선택합니다. 이러한 데이터 포인트는 초기 애플리케이션 및 인프라 인벤토리에서 가져옵니다([데이터 수집](#) 섹션 참조).

그런 다음 각 데이터 포인트의 가능한 각 값에 대한 점수 또는 가중치를 정의합니다. 예를 들어 환경 속성을 선택하고 가능한 값이 프로덕션, 개발 및 테스트인 경우 각 값에 점수가 할당되며, 숫자가 클수록 우선 순위가 높습니다. 선택 사항이지만 중요도 또는 관련성을 각 데이터 포인트에 곱하는 요소를 할당하는 것이 좋습니다. 이 선택적 단계는 더 중요한 것이 무엇인지 강조하는 상위 수준의 차별화 요소를 제공하므로 값에 점수를 반복적으로 할당할 때 기준을 일치시키는 데 도움이 됩니다.

다음 표에는 처음 몇 번의 마이그레이션 웨이브에 대해 위험도가 낮고 간단한 애플리케이션의 우선 순위를 지정하는 전략에 따라 속성 선택 예제와 해당 값 할당이 나와 있습니다.

속성(데이터 포인트)	가능한 값	점수(0~99)	중요도 또는 관련성 곱하기 계수
환경	테스트	60	높음(1x)
	개발	40	
	프로덕션	20	
비즈니스 중요도	낮음	60	높음(1x)
	중간	40	
	높음	20	
규제 또는 규정 준수 프레임워크	없음	60	높음(1x)
	FedRAMP	10	
운영 체제 지원	클라우드 지원	60	중간-높음(0.8x)
	클라우드에서 지원되지 않음	10	
컴퓨팅 인스턴스 수	1-3	60	중간-높음(0.8x)
	4-10	40	
	11개 이상	20	
마이그레이션 전략	리호스팅	70	중간(0.6x)
	리플랫폼	30	
	리팩터링 또는 리아키텍트	10	

애플리케이션 간의 주요 차별화 요소 역할을 할 수 있는 속성을 선택해야 합니다. 그렇지 않으면 기준이 동일한 우선 순위를 공유하는 워크로드가 많습니다. 모델을 적용한 후에는 결과 순위의 상단과 하단을 보고 동의하는지 확인하는 것이 좋습니다. 일반적으로 동의하지 않는 경우 워크로드 점수를 매기는데 사용한 기준을 다시 살펴볼 수 있습니다.

순위를 얻은 후 전체 포트폴리오의 점수 분포를 살펴봅니다. 점수 자체는 중요하지 않습니다. 중요한 것은 점수 간의 차이입니다. 예를 들어 최고 총 점수는 8,000이고 최저 점수는 800입니다. 분포가 좋은지 확인할 수 있도록 결과 점수를 히스토그램으로 도표화하는 것이 좋습니다. 이상적인 분포는 몇 개의 매우 높은 우선 순위 워크로드와 몇 개의 매우 낮은 우선 순위 워크로드가 있는 표준 종형 곡선과 같습니다. 대부분의 애플리케이션은 중간에 있습니다.

초기 우선순위 지정의 또 다른 주요 측면은 클라우드를 조기에 채택하는 데 관심을 보이는 내부 팀 또는 사업부를 포함하는 것입니다. 이는 특히 초기에 특정 애플리케이션을 마이그레이션하기 위한 비즈니스 지원을 받는 데 상당한 도움이 될 수 있습니다. 조직의 경우 위 표에 사업부 속성을 포함시킵니다. 애플리케이션을 제공할 의향이 있는 사업부에 높은 점수를 할당합니다. 사업부 속성을 사용하면 해당 애플리케이션을 목록 맨 위로 가져오는 데 도움이 됩니다.

결과 순위에 동의한 후 상위 5~10개 애플리케이션을 선택합니다. 이들은 초기 애플리케이션 마이그레이션 후보가 됩니다. 3~5개의 애플리케이션을 확인하도록 목록을 구체화합니다. 이를 통해 상세한 애플리케이션 평가를 수행할 때 대상 접근 방식을 취할 수 있습니다. 자세한 내용은 [우선 순위가 지정된 애플리케이션 평가를](#) 참조하세요.

## 마이그레이션을 위한 R 유형 결정

각 애플리케이션 및 관련 인프라에 대한 마이그레이션 전략을 결정하면 마이그레이션 속도, 비용 및 이점 수준에 영향을 미칩니다. 비즈니스 동인, 기술 지침 원칙, 우선순위 기준, 비즈니스 전략 등 요소의 균형 잡힌 조합을 기반으로 전략을 결정하는 것이 중요합니다.

이러한 요인으로 인해 보기가 충돌하는 경우가 있습니다. 예를 들어 마이그레이션의 주요 동인은 혁신과 민첩성일 수 있습니다. 동시에 비용을 빠르게 줄여야 할 수도 있습니다. 범위 내에서 모든 애플리케이션을 현대화하면 장기적으로 비용을 절감할 수 있지만 더 많은 투자를 선결제해야 합니다. 이 경우 한 가지 접근 방식은 리호스팅 또는 리플랫폼과 같이 적은 노력이 필요한 전략을 사용하여 애플리케이션을 마이그레이션하는 것입니다. 이는 단기적으로 빠른 효율성과 비용 절감을 제공할 수 있습니다. 그런 다음 나중에 애플리케이션을 현대화하는 데 비용 절감을 다시 투자하고 추가 비용을 절감합니다.

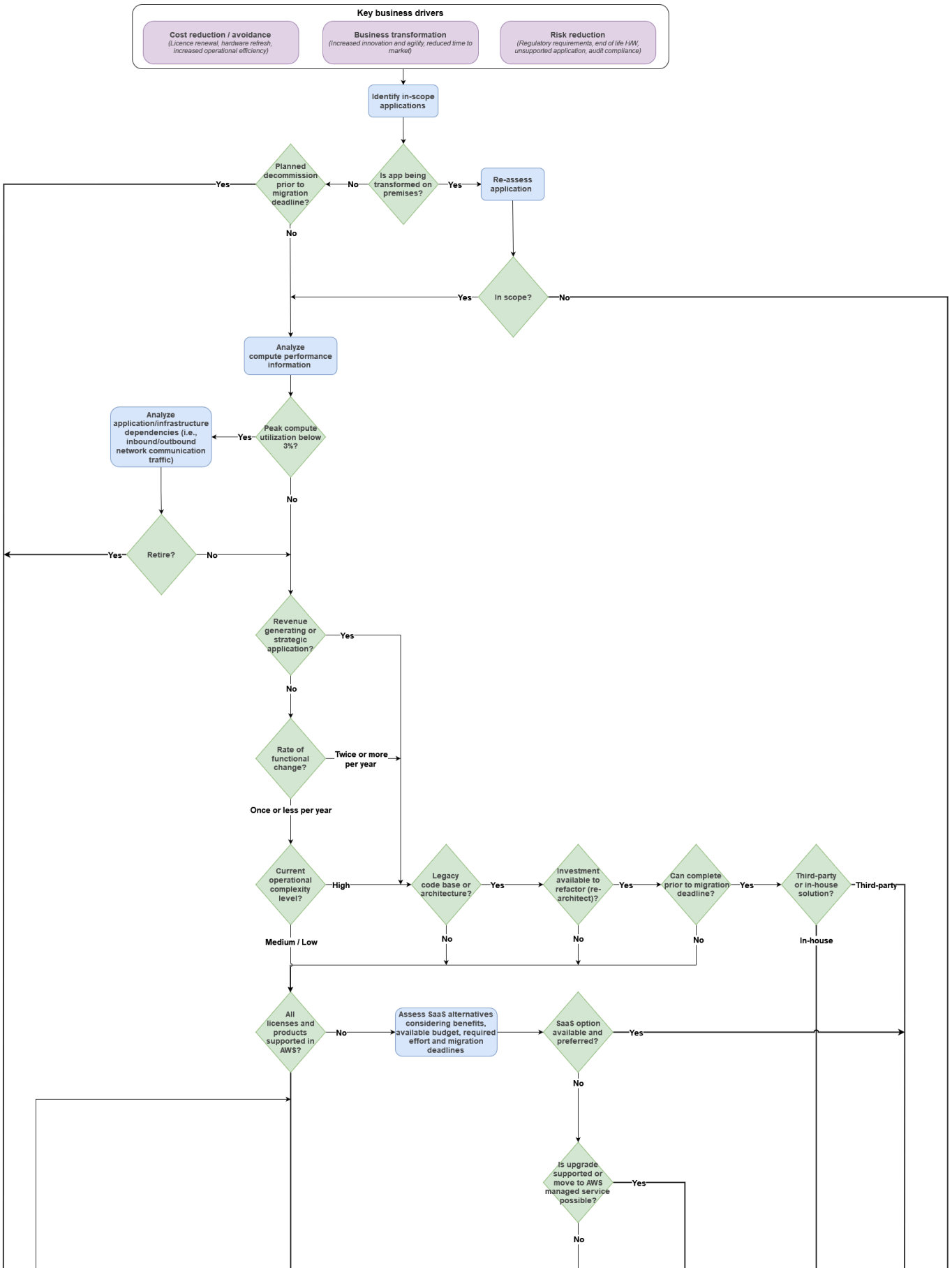
그러나 모든 애플리케이션의 완전한 리호스팅부터 시작하면 현대화의 더 큰 이점이 지연됩니다. 핵심은 마이그레이션 전략 간의 균형을 찾아 비즈니스 전략 애플리케이션이 현대화를 우선시하는 반면 다른 애플리케이션은 먼저 리호스팅하거나 리플랫폼한 다음 현대화할 수 있도록 하는 것입니다.

애플리케이션의 마이그레이션 전략을 어떻게 결정하나요?

이 평가 단계에서는 마이그레이션 전략 선택을 안내하기 위한 초기 모델을 통합하는 데 중점을 둡니다. 초기 애플리케이션의 마이그레이션 전략을 검증하려면 비즈니스 동인 및 우선순위 기준과 함께 모델을 사용합니다. 결정 트리의 기본 로직은 범위에 대한 초기 처리를 결정하는 데 도움이 됩니다. 트리에

---

서는 리팩터링 또는 리아키텍트와 같은 가장 복잡한 접근 방식이 전략적 워크로드용으로 예약되어 있습니다.



이 다이어그램의 사용자 지정 가능한 [draw.io](#) 버전은 [첨부](#) 파일 섹션에서 확인할 수 있습니다.

초기 모델의 첫 번째 단계는 트리 상단의 비즈니스 동인을 조직에서 정의한 동인으로 업데이트하는 것입니다. 다음으로 애플리케이션 전체가 아닌 애플리케이션 구성 요소에 트리를 적용합니다. 예를 들어 세 가지 구성 요소(프론트 엔드, 애플리케이션 계층 및 데이터베이스)가 있는 3계층 애플리케이션의 경우 각 구성 요소는 트리를 독립적으로 전송하고 특정 전략 및 패턴을 할당받아야 합니다. 이는 경우에 따라 특정 티어를 리호스팅하거나 리플랫폼하고 다른 티어를 리팩터링(리아키텍트)할 수 있기 때문입니다.

독립 구성 요소 할당을 통해 연결된 인프라에 대한 마이그레이션 전략을 정의할 수 있습니다. 인프라 전략은 지원하는 애플리케이션 구성 요소와 동일한 전략이거나 다를 수 있습니다. 예를 들어 최신 운영 체제를 사용하는 새 가상 머신으로 리플랫폼되는 애플리케이션 구성 요소는 리플랫폼 전략을 따르는 반면 해당 구성 요소를 호스팅하는 현재 가상 머신은 사용 중지됩니다. 인프라에 대한 마이그레이션 전략은 애플리케이션 구성 요소에 대해 선택한 전략을 기반으로 계산됩니다.

의사 결정 트리를 사용하여 마이그레이션 전략을 설정하기 전에 몇 가지 애플리케이션으로 로직을 테스트하고 일반적으로 결과에 동의하는지 확인합니다. 6Rs 의사 결정 트리는 정확성을 결정하는 데 필요한 분석을 대체하지 않는 가이드입니다. 트리 로직은 특정 사례에 적용되지 않을 수 있습니다. 이러한 사례를 예외로 처리하고 트리 로직을 변경하는 대신 재정의의 근거를 문서화하여 트리가 주도하는 결정을 재정의합니다. 이렇게 하면 관리가 어려울 수 있는 여러 의사 결정 트리 버전을 방지할 수 있습니다. 일반적인 지침은 트리가 워크로드의 70~80% 이상에서 유효해야 한다는 것입니다. 나머지는 예외가 있습니다. 이 평가 단계에서 트리 로직을 조정하려면 초기 모델을 설정하는 데 집중해야 합니다. [포트폴리오 분석 및 마이그레이션 계획과 같은 추가 반복 및](#) 개선은 이후 단계에서 이루어집니다.

## 첨부 파일

[attachment.zip](#)

## 방향성 비즈니스 사례 생성

비즈니스 전반의 이해관계자는 각 단계에서 혁신을 위해 비즈니스 사례를 이해하고 구매해야 합니다.

초기 단계에서는 프로그램을 계획하고 설정하는 데 필요한 리소스를 보호할 수 있도록 마이그레이션 프로그램의 잠재적 가치를 충분히 빠르게 보여주는 것이 중요합니다. 방향성 비즈니스 사례는 조기에 수집할 수 있는 제한된 데이터로 매력적인 비즈니스 가치를 달성하는 데 합리적인 혁신을 제공하도록 설계되었습니다.

프로그램이 설정되면 비즈니스 사례가 추가로 개발됩니다. 자세한 사례는 정확도 향상, 프로그램 가치에 대한 보다 완전한 그림, 계획 우선 순위에 대한 인사이트를 제공합니다. 조직이 구매하는 계획된 비

즈니스 성과를 정의하고 정량화하며, 프로그램 거버넌스 사무소가 프로그램을 주도하고 성과를 측정할 수 있는 기준을 설정합니다.

## 방향성 비즈니스 사례의 범위 수정

방향성 비즈니스 사례는 일반적으로 2~4주 이내에 빠르게 수집됩니다. 리소스를 보호하여 핵심 팀을 설정하고, 필요한 경우 AWS 파트너를 참여시키고, 최소한 [우선 순위가 지정된 애플리케이션 평가](#), [포트폴리오 분석 및 마이그레이션 계획](#) 단계를 완료할 수 있도록 충분한 신뢰도를 생성해야 합니다.

일반적으로 포트폴리오 마이그레이션을 지원하는 방향성 비즈니스 사례는 다음 중 하나로 생성됩니다.

- 있는 그대로 인프라 환경과 마이그레이션 후 AWS 서비스 아키텍처 간의 간단한 총 소유 비용(TCO) 비교입니다. 비교는 지정된 워크로드 볼륨에 대한 예상 실행 속도의 차이를 보여줍니다.
- 마이그레이션 비용과 그대로 유지를 AWS 포함하여 마이그레이션하기 위한 순 현재 가치(NPV), 투자 수익률(ROI), 회수 기간, 수정된 내부 수익률(MIRR) 및 3~5년 현금 흐름 분석을 보여주는 비즈니스 사례입니다.

방향성 비즈니스 사례 범위는 일반적으로 다음 중 하나로 제한됩니다.

- 인프라 기술 비용 비교
- 인프라 기술 및 운영 비용 비교

일반적으로 포트폴리오가 클수록 개발해야 하는 사례가 줄어듭니다. 이는 결과에 큰 영향을 주지 않고 더 광범위한 가정을 할 수 있기 때문입니다. 더 작은 포트폴리오의 경우 변경 사항이 더 큰 영향을 미치므로 더 자세한 정보가 필요합니다.

먼저 기본 인프라 비용 비교를 구축합니다. 그런 다음 계속하기 전에 비교가 충분히 매력적인지 확인합니다. 일반적으로 400개 이상의 서버 포트폴리오는 운영 후 3년 이내에 인프라 비용 절감만으로 AWS 또는 5년 이내에 250개의 서버로 구성된 긍정적인 비즈니스 사례를 보여주지만 이는 다를 수 있습니다. 더 작은 포트폴리오의 경우 더 자세한 정보가 필요할 수 있습니다.

반대로 총 마이그레이션 범위가 약 5개의 워크로드 또는 50개의 서버 미만이 아닌 한 복원력 향상 또는 비즈니스 민첩성에서 파생된 값과 같은 다른 비즈니스 가치 구성 요소들이 단계에서 검사하는 것은 거의 유용하지 않습니다.

## 포커스 값 동인

인프라 기술 TCO 비교는 있는 그대로 인프라 비용의 모델을 동등한 성능과 가용성으로 워크로드를 실행하는 데 필요한 재료 AWS 서비스 표의 기본 모델과 비교합니다. 많은 최적화를 수행할 수 있습니다. 그러나 이 단계에서는 평가가 더 쉽고 일반적으로 약 30%의 TCO 절감을 달성하므로 다음 목록에 중점을 둡니다.

- 컴퓨팅 탄력성 - 8x5(24% 사용량), 10x5(30%) 또는 10x6(36%)을 실행하는 개발 또는 UAT 서버, 2%로 실행되는 재해 복구(DR) 서버와 같이 사용량이 100%가 아닌 서버를 사용 시에만 요금이 청구되는 온디맨드 서비스에 매핑합니다.
- 절감형 플랜으로 조달 - 비용을 최대 75% 절감할 수 있는 적절한 절감형 플랜으로 사용량이 많은 (36% 초과) 프로덕션 서버 및 기타 서버를 조달할 계획입니다. 옵션에는 1년 약정과 3년 약정이 포함되며, 더 큰 할인을 보장하기 위해 선결제 금액이 다릅니다.
- 좀비 제거 - CPU 사용률이 2% 미만인 서버를 식별하여 더 이상 필요하지 않음을 확인하고 비용 분석에서 제거합니다.
- 올바른 컴퓨팅 크기 조정 - CPU 및 메모리 사용률 시계열 데이터를 사용하여 각 서버에 필요한 컴퓨팅 성능과 메모리를 평가합니다. 그런 다음 적합한 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스를 선택합니다.
- 관계형 데이터베이스 관리 시스템(RDBMS) 라이선스 적정 크기 조정 - 데이터베이스 서버의 컴퓨팅 적정 크기 조정 후 RDBMS 라이선스 요구 사항을 재평가하고, 기존 보유 라이선스 사용(BYOL)과에서 라이선스 조달을 비교하고 AWS, Amazon Relational Database Service(RDS)가 비용 절감 효과를 높일 수 있는 가능성을 살펴봅니다.
- 스토리지 - 필요한 총 스토리지 볼륨의 크기를 조정하고 포트폴리오 전반의 초당 입/출력 작업(IOPS) 요구 사항을 식별합니다. SLAs와 비용이 서로 다른 객체 스토리지로 이동할 수 있는 양을 결정합니다.

## 데이터 요구 사항

[초기 평가 데이터 요구 사항 이해](#)의 표에는 방향성 비즈니스 사례의 각 부분을 구축하는 데 필요한 데이터와 필수인지 아니면 선택 사항인지가 나와 있습니다.

사례를 구축하려면 초기 계획 데이터의 인프라 하위 집합과 비용 데이터가 필요합니다. 포함할 인프라를 식별하는 방법은 비즈니스 목표에 따라 결정됩니다.

- 프로그램의 목표가 특정 애플리케이션을 마이그레이션하고 현대화하는 경우 공유된 인프라를 고려하여 애플리케이션에 필요한 것을 기반으로 인프라 포트폴리오를 구축합니다.

- 임대가 만료될 예정인 데이터 센터에서 마이그레이션하는 등 프로그램의 목표가 인프라 중심인 경우 인프라 TCO 비교에 애플리케이션 매핑이 필요하지 않습니다.

선택 사항으로 표시된 데이터(예: 서버의 CPU 및 메모리 피크 사용률)는 일반적으로 표준 벤치마크 값으로 대체할 수 있습니다. AWS 파트너 또는 AWS 전문 서비스와 이에 대해 논의할 수 있습니다. 또는 포트폴리오의 일부(예: 하이퍼바이저에서 수집한 데이터)에서 사용할 수 있는 데이터 포인트의 값을 추정할 수 있습니다. 포트폴리오가 클수록 더 정확합니다.

## 인프라 TCO 비교 구축

도구는 인프라 TCO 비교를 구성하는 데 매우 중요합니다. [AWS 전문 서비스](#) 또는 [AWS 파트너는](#) 특히 광범위한 마이그레이션 프로세스를 지원하기 위해 참여하려는 경우 모든 유형의 방향성 사례에 도움을 제공할 수 있습니다.

다음은 수행하는 데 사용할 수 있는 도구가 있습니다.

- 인벤토리 데이터를 수집합니다.
- 사용률 데이터를 수집합니다.
- 정확한 그대로 인프라 비용 벤치마킹 데이터를 제공합니다.
- 좀비 식별 및 제거.
- 적절한 규모의 평가를 수행합니다.
- 구매 옵션을 권장합니다.
- 소프트웨어 라이선스 옵션을 비교합니다.
- 간단한 그래픽 현금 흐름 분석을 생성합니다.

의 [마이그레이션 평가자](#)는 한 가지 옵션 AWS입니다. 이 모든 기능을 무료 관리형 서비스로 제공합니다. AWS 계정 관리자 또는 마이그레이션 역량 파트너를 통해 또는 온라인으로 요청을 제출하여 AWS 마이그레이션 평가자를 요청할 수 있습니다. <https://pages.awscloud.com/Migration-Evaluator-request.html> 마이그레이션 평가자는 인프라 기술 TCO 비교를 신속하게 생성하기 위한 포인트 솔루션으로 특별히 설계되었습니다.

주요 이점:

- 무료
- 도구 기반 검색이 제한된 인벤토리 데이터의 에이전트 없는 검색 또는 수동 구성
- 배포, 구성, 데이터 수집 및 기본 사례 또는 방향성 비즈니스 사례 구축을 지원하는 전용 지원

- SaaS 작업의 편리성, 하지만 분석 엔진에 로드하기 전에 스크리빙을 지원하기 위해 고객 네트워크 내에서 데이터 수집을 완전히 실행할 수 있음
- Microsoft 라이선스 적정 크기 조정에 대한 강력한 지원
- 전체 데이터 내보내기 기능

#### 주요 제한 사항:

- x86 아키텍처 서버(Windows 및 Linux)만 평가
- 벤치마크를 있는 그대로 비용 데이터를 구성하거나 보정하는 제한된 옵션
- 운영 비용 최적화 모델링을 지원하지 않음
- 마이그레이션 비용 모델링을 지원하지 않음
- TCO 비교를 넘어 비즈니스 사례 구축을 직접 지원하지 않음

애플리케이션 스택 및 상호 종속성 검색과 같은 포트폴리오 검색 및 분석 기능에 상용 검색 도구를 사용하기로 결정한 경우 일반적으로 인프라 TCO 비교도 제공합니다. 포트폴리오 검색 및 평가를 위한 도구 사용에 [대한 지침은 검색 도구의 필요성 평가를 참조하세요](#). 시장 최고의 도구의 주요 기능을 검토하고 비교하려면 [Discovery, Planning 및 Recommendation 마이그레이션 도구](#)를 참조하세요.

## 운영 비용 최적화 구축

IT 운영 생산성 개선은 마이그레이션의 중요한 가치 기여자인 경우가 많습니다. [Amazon Web Services를 통한 비즈니스 가치 창출을 위한 비즈니스 및 조직 혁신 촉진](#) 백서에 따르면 마이그레이션 후 평균 AWS적으로 IT 운영 직원 생산성은 마이그레이션을 통해 62% 증가합니다. 그러나 크기 조정과 방향성 사례에 이러한 이점 포함에는 두 가지 문제가 있습니다.

첫째, 생산성 향상의 전체 범위를 평가하려면 광범위한 데이터 수집이 필요하며 [자세한 비즈니스 사례](#)에 더 적합합니다. 이 문제는 간단한 벤치마크 데이터로 보다 쉽게 평가되고 크기가 조정되지만 여전히 상당한 이점을 보이는 몇 가지 요소에 집중하여 해결할 수 있습니다.

둘째, 비용 절감의 원인으로 생산성에 집중하면 주요 고객 이해관계자와 프로그램 구성원 간에 우려와 부정성이 발생할 수 있습니다. 혜택이 실현되는 방식과 영향을 받는 사람에게 어떤 의미가 있는지 명확하게 설명해야 합니다. 이러한 문제는 팀의 역할만 개선한다는 점을 명확히 하여 피할 수 있습니다.

- 마이그레이션 프로그램에는 내부 운영 직원을 개발하고 코드 자동화로 인프라를 구축하는 DevSecOps 팀에 합류하고 팀의 성장을 주도할 자동화를 테스트하는 등 새로운 역할로 이동하는 트랙이 포함되어 있습니다.

- 내부 직원이 고부가가치 활동에 집중할 수 있도록 운영 아웃소싱 계약의 크기를 조정하고 크기를 조정하여 이점을 실현할 수 있습니다.

고려하려는 운영 혁신을 기반으로 비즈니스 사례 요소를 구성하는 접근 방식:

- 기존 사내 운영 팀이 있는 경우 팀원의 역량을 높이고 예상되는 생산성 개선을 보여줍니다.
- 또는 현재 운영 솔루션에서 AWS Managed Services (AMS) 또는 AWS 파트너의 대체 관리형 서비스로 마이그레이션합니다.

첫 번째 변환의 경우 사례에 포함될 수 있는 생산성 향상에 대한 보수적인 재무 추정치를 얻으려면 다음을 수행하는 것이 좋습니다.

1. 특히 서버 관리 운영 생산성에 중점을 둡니다. 운영 작업의 상당 부분이고, 더 쉽게 평가할 수 있으며, 나중에 더 쉽게 확인되는 경향이 있습니다.
2. 각 정규직(FTE) 직원이 관리할 수 있는 서버 수에 대한 벤치마크를 기반으로 필요한 인력 배치를 계산합니다. 온프레미스에서 이 수는 약 150개의 서버입니다. 약 400 AWS개의 서버가 있습니다.
3. 이러한 지표를 EC2 인스턴스 수와 비교하여 온프레미스 서버 수에 적용합니다.
4. 전체 운영 팀에 대한 혼합 비용 효율로 절약된 시간을 곱합니다.

그런 다음 결과가 다음 표(IDC 백서 [비즈니스 촉진 및 Amazon Web Services를 통한 비즈니스 가치 창출을 위한 조직 변환](#)에서 가져온 데이터)에 제공된 역할별 평균 생산성 향상을 크게 초과하지 않는지 확인하여 두 방법 중 하나로 결과를 확인할 수 있습니다.

역할	효율성 향상
IT 인프라 관리	62%
IT 지원	59%
애플리케이션 관리	43%
데이터베이스 관리	19%
애플리케이션 개발	25%

두 번째 변환의 경우 범위 내 포트폴리오의 현재 총 운영 및 지원 비용을 고려 중인 관리형 서비스의 비용과 직접 비교하여 운영 비용 절감을 추가할 수 있습니다.

관리형 서비스의 비용을 얻으려면 AWS 계정 관리자 또는 [AWS Managed Services 파트너에게](#) 제안된 자체 AWS 명세서, 서비스 수준 선택(Plus 또는 Premium) 및 AMS 패키지(AMS Accelerate 또는 AMS Advanced)를 제공하세요. 이렇게 하면 변환된 솔루션의 구성 요소인 AWS 서비스에 대한 관리형 서비스의 총 비용이 제공됩니다. 마찬가지로 자체 파라미터를 기반으로 자체 관리형 서비스 패키지를 제공하는 AWS 파트너로부터 요금을 받을 수 있습니다.

## 전체 방향성 비즈니스 사례로 확장

일반적으로 전체 방향성 비즈니스 사례를 수집하려면 IT 생산성 요소를 사용하거나 사용하지 않고 TCO 비교를 구축하고 모든 마이그레이션 및 현대화 비용을 추정합니다. 그런 다음 migrate-and-modernize 시나리오 쌍을 다루고 t-migrate-and-modernize하지 않는 현금 흐름을 생성합니다.

가장 기본적인 사례는 단일 시나리오 쌍을 준비하는 것입니다. 여기서 t-migrate-and-modernize 않는 시나리오는 현재 상황이고 migrate-and-modernize 시나리오는 다음과 같은 특성을 갖습니다.

- 트랜잭션 볼륨, 컴퓨팅 또는 네트워킹 용량의 증가 또는 감소 없음
- 스토리지 요구 사항의 안정적인 소량 증가
- 기존 시스템의 기능과 일치하는 Quality-of-service 기능(예: 가용성, 내구성, 처리량 및 성능)

매우 작은 포트폴리오를 제외한 모든 포트폴리오의 경우 방향성 사례 모음을 구축하는 목표에 적합합니다. 앞으로 나아갈 수 있는 권한을 얻기에 충분한 가치를 빠르게 보여줍니다.

소규모 포트폴리오의 경우 다음과 같이 클라우드 migrate-and-modernize와 마이그레이션 t-migrate-and-modernize 금지 시나리오 쌍을 추가하는 것이 유용할 수 있습니다.

- 이러한 성장이 예상되는 워크로드 전반의 중간 및 대용량 성장 요구 사항 혼합
- 고가용성, DR 및 내결함성과 같은 향상된 복원력 포함
- 엣지 컴퓨팅, 콘텐츠 전송 네트워크(CDN), 다중 리전 데이터베이스 복제를 통해 글로벌 성능을 개선했습니다.
- 프로그램의 비즈니스 우선 순위를 설정한 기타 특정 개선 서비스 품질

이러한 시나리오의 경우 현재 비클라우드 인프라 아키텍처를 새 사양과 일치하도록 업그레이드할 때 발생하는 비용 및 현금 흐름 영향을 정확하게 추정해야 합니다. 이 추정치를 얻는 가장 직접적인

방법은 시스템 통합자에게 인용을 요청하는 것일 수 있습니다. 특히 마이그레이션 및 migrate-and-modernize 시나리오와 마이그레이션 t-migrate-and-modernize 시나리오를 모두 지원할 수 있는 마이그레이션 역량을 갖춘 AWS 컨설팅 파트너인 경우 더욱 그렇습니다.

각 시나리오 쌍에 대해 다음으로 구성된 사례를 수집합니다.

- 마이그레이션 t-migrate-and-modernize 애플리케이션 시나리오의 비용입니다. 가장 기본적인 경우에는 다음이 포함됩니다.
  - 현재 인프라 구성의 비즈니스 사례 기간 동안 총 소유 비용
  - 컴퓨팅, 스토리지 및 네트워크 트래픽 소비의 주기적 증가
- 다음을 포함한 migrate-and-modernize 시나리오의 비용:
  - 세부 검색, 마이그레이션 계획, 세부 비즈니스 사례 개발, 핵심 팀 설정 및 기술 향상, 아직 없는 경우 랜딩 존 설정, 마이그레이션된 워크로드에 대한 보안 관리 및 운영 통합 설정 등 프로그램 설정
  - 워크로드 마이그레이션 및 현대화 비용
  - 네트워크 연결, [AWS Snowball Edge](#) 및와 같은 데이터 마이그레이션 서비스를 포함한 마이그레이션 인프라 비용 [AWS DataSync](#), 마이그레이션 프로세스 자체에 필요한 아키텍처의 AWS 유틸리티 비용(예: 테스트용)
  - 웨이브가 가동됨에 따라 마이그레이션 과정에서 AWS 유틸리티 비용의 증가 및 AWS 기반 서비스로 대체되고 폐기될 때 기존 인프라 비용의 감소
- 모든 부실 자산에 대한 폐기 비용 및 상각

## 마이그레이션 및 현대화 프로그램 설정 추정

성공을 위한 프로그램을 설정하려면 일련의 기본 활동을 실행하여 기존 기능을 구축하고 이전에 수행하지 않은 경우 세부 계획을 수립해야 할 수 있습니다. 이러한 기본 활동에는 다음이 포함됩니다.

1. 포트폴리오 [분석 및 마이그레이션 계획](#) 섹션에 설명된 대로 세부 포트폴리오 검색, 마이그레이션 계획 및 세부 비즈니스 사례 개발을 수행하고 사용된 검색 도구의 비용을 문서화합니다.
2. 교육 및 채용을 통해 클라우드 비즈니스 및 기술 핵심 팀을 구축하고 사내 기술을 개발합니다. 교육이 필요한 IT 조직의 구성원을 식별하고 각 사람에게 교육 예산을 할당합니다.
3. 필요한 비용, 운영 및 보안 거버넌스 기능을 지원하도록 [랜딩 존](#)을 설정하고 구성합니다.

AWS 컨설팅 파트너는 항목 1 및 3에 대한 견적을 제공하는 데 도움을 줄 수 있습니다.

### 마이그레이션 및 현대화 비용 추정

방향성 비즈니스 사례의 목표를 충족하고 다음 단계로 진행하기에 충분한 상업적 잠재력을 입증하려면 마이그레이션 및 현대화 비용 추정을 가능한 한 기본으로 유지합니다.

이를 위해 다음 마이그레이션 전략에 속하는 애플리케이션에 집중하여 방향성 비즈니스 사례를 준비하는 것이 좋습니다.

- 만료
- 보관
- 재배포하다
- 리호스팅
- 리플랫폼
- 재구매

일반적으로 워크로드의 약 70%는 리호스팅, 재배포 또는 리플랫폼할 수 있으며, 또 다른 5%는 사용 중지할 수 있습니다. 마이그레이션 전략을 통해 애플리케이션을 평가하면 일반적으로 비용 절감 사례의 핵심이 해결됩니다.

리팩터링 또는 리아키텍팅 비용을 추정하는 것은 복잡할 수 있습니다. 방향성 비즈니스 사례를 준비하는 데 주어진 기간 내에 이를 시도하는 것은 실용적이지 않습니다. 앞서 [마이그레이션을 위한 R 유형 결정](#)에서 설명한 대로 마이그레이션 및 현대화의 첫 번째 단계에서 리호스팅, 재배포 또는 리플랫폼 전략을 사용하는 것이 좋습니다. 이러한 R 전략은 초기 페이백을 가속화하고, 구현 위험을 줄이고, 단기적으로 비즈니스 사례를 개선할 가능성이 높습니다. 또한 애플리케이션 팀이 AWS 환경 내에서 실행 중인 애플리케이션을 그렇지 않은 애플리케이션보다 현대화하는 것이 훨씬 쉽습니다. [세부 비즈니스 사례가](#) 준비되면 특정 애플리케이션을 리팩터링(리아키텍처링)하기 위한 추정치를 추가하는 것이 가장 좋습니다.

### 전략별 마이그레이션 작업 추정

마이그레이션마다 다릅니다. 예산 또는 계획을 약정하기 전에 프로젝트를 책임질 팀의 마이그레이션 활동에 대한 워크로드 추정치를 시드하세요. 이는 사내 애플리케이션 팀, AWS 전문 서비스 또는 AWS 파트너 조직입니다.

방향성 사례를 구축하는 데 도움이 되도록 다음 표에는 다양한 처리에 대한 표시 작업 범위가 나와 있습니다. 이러한 범위는 medium-to-large 포트폴리오가 마이그레이션되고 있으며 마이그레이션 팀이 훈련되고 경험이 있다고 가정합니다. 소규모 포트폴리오의 경우 방향성 사례에도 마이그레이션을 담당하는 팀이 견적을 준비하도록 하는 것이 가장 좋습니다.

마이그레이션 전략	추정 프로세스	요소	사람 시간	사람 시간
Retain	Do nothing, with no cost, no benefits, and no reduction in technology debt.	-	-	-
Retire	Estimate decommissioning the hardware 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.	-	-	-
Rehost	Estimate copying the workload and data with an image copy,	Effort per app per server  Low	Migration  10-14	HA/DR test  3-5

마이그레이션 전략	추정 프로세스	요소	사람 시간	사람 시간
	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 <a href="#">AWS Application Migration Service</a> . 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, integration complexity, and data volumes.	Medium  High	16–24  26–38	4–6  8–12

마이그레이션 전략	추정 프로세스	요소	사람 시간	사람 시간
Replatform	For replatform migration 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 <a href="#">AWS Schema Conversion Tool</a> and <a href="#">AWS Database Migration Service</a> , and a more complete application test. An example of changing the	<p>Effort per app per server</p> <p>Low</p> <p>Medium</p> <p>High</p>	<p>Version up</p> <p>Add 1–3</p> <p>Add 2–5</p> <p>Add 4–8</p>	<p>Technology change</p> <p>Add 10–15</p> <p>Add 20–30</p> <p>Add 40–60</p>

마이그레이션 전략	추정 프로세스	요소	사람 시간	사람 시간
	technology is 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.	-	-	-

### 마이그레이션 인프라 비용 추정

마이그레이션 과정에서 사용할 인프라에 대한 추정치를 포함합니다. 일반적으로 이러한 추정치는 다음과 같습니다.

- 현재 환경에서 로 워크로드 및 데이터 마이그레이션을 위한 연결 및 데이터 교환 서비스 예산 AWS
- 마이그레이션, 테스트 및 전환 프로세스 중에 마이그레이션된 워크로드를 호스팅하는 데 필요한 AWS 서비스 (특히 컴퓨팅 및 스토리지) 예산
- 각 마이그레이션 웨이브가 완료될 때 AWS 유틸리티 비용 증가
- 마이그레이션된 워크로드를 더 이상 실행하지 않을 기존 인프라의 폐기 비용

데이터 교환의 경우 총 데이터 볼륨을 검사하고 네트워킹 사용 가능성을 평가합니다. 마이그레이션 후 운영 용도로 WAN의 특정 지점 [Site-to-Site VPN](#) AWS 으로 또는 [AWS Direct Connect](#) 링크를 미리 프로비저닝한 경우 해당 리소스를 서비스 할당량까지 사용할 수 있습니다.

네트워크 용량이 충분하지 않은 경우 가상 프라이빗 네트워크(VPN)를 사용하여 인터넷 대역폭을 단기간으로 늘리는 것이 비용 효율성이 높은 솔루션인 경우가 많습니다. 그렇지 않은 경우 [AWS Snowball Edge](#) 및와 같은 AWS 미디어 교환 디바이스는 대부분 솔루션을 [AWS Snowball Edge](#) 제공합니다 AWS 리전. 또한 대용량 데이터 마이그레이션의 경우에 대한 예산을 포함 [AWS DataSync](#) 시켜 안정성을 개선하고 사용된 미디어에 관계없이 전송을 가속화할 수 있습니다.

비즈니스 사례의 현금 흐름 분석 요소에는 AWS 서비스 증가 및 기존 인프라 감소 모델링이 중요합니다. 이 단계에서는 비용이 발생하는 시기를 정확히 결정할 웨이브 플랜이 없을 것입니다. 다음과 같이 하는 것이 좋습니다:

- 마이그레이션 기간 동안 AWS 의 비용을 일정한 속도로 늘립니다.
- 동일한 기간 동안 일정한 속도로 폐기하려는 기존 인프라의 비용을 줄입니다.

기존 인프라가 축소되기 1~2개월 전에 AWS 비용이 증가합니다. 이렇게 하면 각 웨이브에 대해 마이그레이션을 수행할 수 있는 1개월의 AWS 유틸리티 사용량이 제공됩니다. 여기에는 테스트 시간과 교체된 인프라에서 비용 발생을 중지하는 데 필요한 폐기 작업을 완료하는 추가 시간이 포함됩니다.

## 폐기 비용 추정

재배포할 수 없는 장비를 폐기하고 합법적이고 환경 친화적인 방식으로 폐기하면 약간의 비용이 발생할 수 있습니다. 그러나 방향성 비즈니스 사례의 경우 일반적으로 교체된 자산의 나머지 책 가치를 상각하는 비용이 유일한 금액입니다.

방향성 비즈니스 사례의 경우 다음을 수행하는 것이 좋습니다.

- 자산 목록을 검토합니다.
- 폐기될 항목을 식별합니다.
- 상각을 줄이려면 디바이스를 전환할 기회를 검토하여 목록에 있는 최신 디바이스를 사용하여 더 오래되고 사용 중단된 자산을 교체할 수 있도록 합니다.
- 해당 시점에 폐기될 자산의 향후 책 가치를 평가합니다.
- 이를 폐기의 마이그레이션 비용으로 포함합니다.

## 전체 방향성 비즈니스 사례 수집 및 조정

각 시나리오 쌍에 대한 전체 비용 세트를 준비한 후 각 시나리오에 대해 할인된 현금 흐름 명세서를 작성하고 그래프로 표시합니다. 하드웨어 새로 고침 주기와 동일한 기간 동안 방향성 비즈니스 사례를 구축하는 것이 좋습니다. 서버, 스토리지 및 네트워크 디바이스의 경우 일반적으로 5년입니다. 하드웨어 새로 고침 주기와 동일한 기간을 사용하는 경우 정확히 한 번의 새로 고침 비용이 각 시나리오의 있는 그대로 비용에 포함됩니다.

그런 다음 프로그램의 다음 단계로 진행하기 위한 승인을 받는 데 필요한 주요 재무 지표를 계산합니다. 일반적으로 다음을 포함합니다.

- 평가된 비용 절감 및 생산성 향상의 절대값을 측정하기 위한 순 현재 가치(NPV)
- 반환이 충분히 빠르지 확인하기 위한 월별페이백 기간
- 프로세스가 기간 동안 충분한 비용을 소비하고 있는지 확인하기 위한 최종 실행률 비교
- 투자 수익률(ROI) 및 수정된 투자 수익률(MIRR)로, 조직이 우선시할 수 있는 다른 자본 수요에 비해 프로그램의 상대적 재무 성과를 평가합니다.

사례의 첫 번째 반복을 사용하여 다음 예제와 같이 예상 재무 성과가 개선이 이루어져야 함을 의미하는 지 확인합니다.

- 회수 속도가 너무 느리면 다음과 같이 마이그레이션 비용을 가속화하고 줄일 수 있는 옵션을 고려하세요.
  - AWS 파트너 또는 AWS 전문 서비스를 사용하여 사용 가능한 리소스를 확장하고 더 기본적인 패턴으로 워크로드 마이그레이션을 추가로 병렬화할 수 있습니다.
  - VMware에서 실행되는 워크로드의 경우 최소한 초기 단계에서 재배치 전략을 리호스팅 또는 리플랫폼 전략과 비교합니다. 재배치 전략을 사용하면 마이그레이션 비용을 줄이고 마이그레이션 속도를 높일 수 있습니다.
  - 기술적으로 가능한 경우 더 복잡한 리플랫폼 또는 리팩터링(리아키텍트) 전략이 필요한 워크로드를 초기 비즈니스 사례 범위를 벗어나 미래 단계로 푸시합니다.
- ROI와 MIRR이 너무 낮으면 다음을 고려하세요.
  - 고려 중인 시나리오가 너무 보수적입니까? 가장 가능성이 높은 용량 증가 및 탄력성 요구 사항을 반영하는 시나리오가 있습니까? 목표 내에서 서비스 품질 증가를 포함한 비용을 비교하는 시나리오가 있습니까?
  - 첫 번째 단계에서 마이그레이션할 애플리케이션 포트폴리오의 범위를 구체화하여 현재 사용률이 낮거나 비용이 많이 드는 재해 복구(DR) 요구 사항이 있는 워크로드와 같이 더 강력한 수익을 창출할 워크로드에 집중할 수 있습니까?

- 애플리케이션 포트폴리오의 범위를 구체화하여 처음에 덜 상업적으로 달성되는 특정 워크로드를 제외할 수 있습니까? 예를 들어 퍼블릭 클라우드 인프라에 배포하는 조건이 다르기 때문에 타사 소프트웨어 라이선스가 더 비싼 워크로드를 연기할 수 있나요?
- 최종 실행률 비교가 예상 목표를 충족하지 않는 경우 다음을 살펴봅니다.
  - 먼저 다른 지표가 기대치를 충족하는지 확인합니다. 방향성 비즈니스 사례는 주로 마이그레이션 준비의 다음 단계를 시작하는 것을 정당화할 수 있는 충분한 재정적 기회가 있음을 보여주는 것입니다.
  - 마이그레이션의 초기 단계 AWS 이후에에서 비용 성능을 계속 개선할 수 있는 기회 목록을 식별합니다.

세부 비즈니스 사례를 준비할 때 기회 목록에 대한 평가를 포함합니다. 또한 사례의 지속적인 유지 관리와 마이그레이션 완료 후 month-to-month 비용 최적화 프로세스에 기회 평가를 포함합니다.

## 우선 순위가 지정된 애플리케이션 평가

이전 단계의 주요 결과 중 하나인 [포트폴리오 검색 및 초기 계획](#)은 세부 평가를 위해 [애플리케이션 하위 집합의 우선순위를](#) 지정하는 것이었습니다. 이 섹션에서는 애플리케이션에 대한 자세한 평가를 살펴봅니다.

초기에 몇 가지 애플리케이션의 세부 정보를 살펴보면 가속화가 촉진됩니다. 평가 및 미래 아키텍처 설계 프로세스는 잠재적 차단을 표시하고 더 큰 규모의 마이그레이션을 주도하는 중요한 작업을 명확히 합니다. 이러한 작업에는 랜딩 존과 같은 AWS 기반을 설정 AWS하거나 기존 랜딩 존을 확장 및 검증하기 위한 요구 사항 수집이 포함됩니다. 이 평가는 마이그레이션 단계와 전략을 고려해야 할 때이기도 합니다.

이 단계의 주요 결과는 다음과 같습니다.

- 우선 순위가 지정된 애플리케이션의 검증된 목록
- 문서화된 현재 상태 아키텍처
- 마이그레이션 후보를 위한 문서화된 초기 대상 아키텍처 및 마이그레이션 전략
- 식별된 마이그레이션 패턴 및 도구
- 문서화된 플랫폼 요구 사항(보안, AWS 인프라 및 운영)
- 마이그레이션 계획을 위한 문서화된 전환 고려 사항
- 예상 AWS 실행률

## 세부 평가 데이터 요구 사항 이해

다음 표에서는 마이그레이션의 애플리케이션 및 관련 인프라에 대한 전체 포트폴리오 보기를 얻는 데 필요한 정보를 설명합니다.

테이블은 다음 약어를 사용합니다.

- R, 필수
- O, 선택 사항의 경우
- 해당 없음, 해당 없음

### 애플리케이션

속성 이름	설명	검색, 설계 및 마이그레이션 전략	예상 실행률	권장 충실도 수준 (최소)
고유한 식별자	애플리케이션 ID를 예로 들 수 있습니다. 일반적으로 기존 CMDBs 또는 기타 내부 인벤토리 및 제어 시스템에서 사용할 수 있습니다. 조직에 고유 ID가 정의되지 않은 경우 고유 IDs를 생성하는 것이 좋습니다.	R	O	높음
애플리케이션 이름	이 애플리케이션이 조직에 알려진 이름입니다. 해당하는 경우 상용 off-the-shelf(COTS) 공급업체 및 제품 이름을 포함합니다.	R	R	높음
COTS입니까?	예 또는 아니요. 상용 애플리케이션인지 아니면 내부 개발인지 여부	R	R	높음
COTS 제품 및 버전	상용 소프트웨어 제품 이름 및 버전	R	R	높음

속성 이름	설명	검색, 설계 및 마이그레이션 전략	예상 실행률	권장 충실도 수준 (최소)
설명	기본 애플리케이션 함수 및 컨텍스트	R	O	높음
중요도	예: 전략적 또는 수익 창출 애플리케이션 또는 중요한 함수 지원	R	O	높음
Type	예: 데이터베이스, 고객 관계 관리(CRM), 웹 애플리케이션, 멀티미디어, IT 공유 서비스	R	O	높음
환경	예: 프로덕션, 사전 프로덕션, 개발, 테스트, 샌드박스	R	R	높음
규정 준수 및 규제	워크로드에 적용되는 프레임워크(예: HIPAA, SOX, PCI-DSS, ISO, SOC, FedRAMP) 및 규제 요구 사항	R	O	높음
종속성	내부 및 외부 애플리케이션 또는 서비스에 대한 업스트림 및 다운스트림 종속성	R	해당 사항 없음	높음

속성 이름	설명	검색, 설계 및 마이그레이션 전략	예상 실행률	권장 충실도 수준 (최소)
인프라 매핑	애플리케이션을 구성하는 물리적 및/또는 가상 자산에 매핑	R	R	높음
라이선스	상품 소프트웨어 라이선스 유형(예: Microsoft SQL Server Enterprise)	R	R	높음
비용	소프트웨어 라이선스, 소프트웨어 운영 및 유지 관리 비용	해당 사항 없음	R	중간-높음
사업부	예: 마케팅, 재무, 영업	R	O	높음
소유자 세부 정보	애플리케이션 소유자의 연락처 정보	R	O	높음
아키텍처 유형	예: 웹 애플리케이션, 2티어, 3티어, 마이크로서비스, 서비스 지향 아키텍처(SOA)	R	R	높음
Recovery Point Objective(RPO), Recovery Time Objective(RTO) 및 /service-level Agreement(SLA)	현재 서비스 - 관리 속성	R	R	높음

속성 이름	설명	검색, 설계 및 마이그레이션 전략	예상 실행률	권장 충실도 수준 (최소)
수익 창출 애플리케이션 또는 비즈니스 전략 애플리케이션?	예, 애플리케이션이 회사 수익에 직간접적으로 영향을 미치거나 비즈니스에서 전략적으로 간주되는 경우 가능합니다.	R	O	중간-높음
사용자 수(동시)	예: 내부 또는 외부 사용자 또는 내부 및/또는 외부 사용자/고객	R	R	중간-높음
사용자 위치	사용자 세션의 오리진	R	R	중간-높음
위험 및 문제	알려진 위험 및 문제	R	O	중간-높음
마이그레이션 고려 사항	마이그레이션과 관련이 있을 수 있는 추가 정보	R	R	중간-높음
마이그레이션 전략	예를 들어 마이그레이션을 위한 AWS 6R 중 하나	R	R	중간-높음
데이터베이스 세부 정보	예: 파티셔닝, 암호화, 복제, 확장, SSL(Secure Sockets Layer) 지원	R	R	높음
지원 팀	예: 애플리케이션 운영 팀 이름	R	O	중간-높음

속성 이름	설명	검색, 설계 및 마이그레이션 전략	예상 실행률	권장 충실도 수준 (최소)
모니터링 솔루션	이 애플리케이션을 모니터링하는 데 사용되는 제품	R	O	중간-높음
백업 요구 사항	의 필수 백업 일정 AWS	R	R	중간-높음
DR 정보	예:이 애플리케이션의 재해 복구 구성 요소	R	R	중간-높음
대상 AWS 요구 사항	예: 구성 요소, 계정 배치, 네트워킹, 보안	R	R	높음

## 인프라

속성 이름	설명	검색, 설계 및 마이그레이션 전략	예상 실행률	권장 충실도 수준 (최소)
고유한 식별자	서버 ID를 예로 들 수 있습니다. 일반적으로 기존 CMDBs 또는 기타 내부 인벤토리 및 제어 시스템에서 사용할 수 있습니다. 조직에 고유 ID가 정의되지 않은 경우 고유 IDs를 생성하는 것이 좋습니다.	R	O	높음

네트워크 이름	네트워크의 자산 이름(예: 호스트 이름)	R	O	높음
DNS 이름(정규화된 도메인 이름 또는 FQDN)	DNS 이름	O	O	중간-높음
IP 주소 및 넷마스크	내부 및/또는 퍼블릭 IP 주소	R	R	높음
애셋 유형	예: 물리적 또는 가상 서버, 하이퍼바이저, 컨테이너, 디바이스, 데이터베이스 인스턴스	R	R	높음
제품 이름	상용 공급업체 및 제품 이름(예: VMware ESXi, IBM Power Systems, Exadata)	R	R	높음
운영 체제	예: REHL 8, Windows Server 2019, AIX 6.1	R	R	높음
구성	할당된 CPU, 코어 수, 코어당 스레드 수, 총 메모리, 스토리지, 네트워크 카드	R	R	높음

사용률	CPU, 메모리, 스토리지 피크 및 평균. 데이터베이스 인스턴스 처리량.	R	R	높음
라이선스	상품 라이선스 유형(예: RHEL Standard)	R	R	높음
공유 인프라입니까?	인증 공급자, 모니터링 시스템, 백업 서비스 및 유사한 서비스와 같은 공유 서비스를 제공하는 인프라 서비스를 나타내는 예 또는 아니요	R	O	높음
애플리케이션 매핑	이 인프라에서 실행되는 애플리케이션 또는 애플리케이션 구성 요소	R	O	높음
통신 데이터	예: 프로세스 수준에서 서버 간	R	해당 사항 없음	중간-높음
대상 AWS 요구 사항	예: 인스턴스 유형, 계정, 서브넷, 보안 그룹, 라우팅	R	R	높음
마이그레이션 전략, 패턴 및 도구	예를 들어 마이그레이션을 위한 6R 중 하나, 특정 기술 패턴, 마이그레이션 도구	R	O	높음

위험 및 문제	알려진 위험 및 문제	R	O	중간-높음
---------	-------------	---	---	-------

## 세부 애플리케이션 평가

세부 애플리케이션 평가의 목표는 대상 애플리케이션 및 관련 인프라(컴퓨팅, 스토리지 및 네트워크)를 완전히 이해하는 것입니다. 위험을 방지하려면 충실도가 높은 데이터가 필요합니다. 예를 들어 조직은 애플리케이션을 완전히 이해하고 있다고 가정하는 것이 일반적입니다. 이는 자연스러운 현상이며 대부분의 경우 마찬가지입니다. 그러나 비즈니스에 대한 위험을 최소화하려면 프로그래밍 데이터를 최대한 확보하여 제도적 지식과 정적 문서를 검증하는 것이 중요합니다. 이렇게 하면 검색 프로세스의 부담을 덜 수 있습니다. 비즈니스별 정보, 전략적 로드맵 등과 같은 대체 소스에서 나오는 데이터 요소에 집중할 수 있습니다.

핵심은 마이그레이션 중 및 마이그레이션 후 마지막 순간의 변경을 방지하는 것입니다. 예를 들어 마이그레이션할 때는 서버를 지속적인 마이그레이션 웨이브에 포함해야 할 수 있는 식별되지 않은 종속성에 따른 변경을 피하는 것이 중요합니다. 마이그레이션 직후 트래픽을 허용하거나 추가 서비스를 배포하려면 관련 플랫폼 요구 사항에 따른 변경을 피하는 것이 중요합니다. 이러한 종류의 예상치 못한 변경은 보안 및 운영 문제의 위험을 높입니다. 자세한 애플리케이션 평가를 수행할 때 프로그래밍 방식의 검색 도구를 사용하여 트래픽 패턴과 종속성을 검증하는 것이 좋습니다.

평가를 시작할 때 애플리케이션 이해관계자를 식별해야 합니다. 일반적으로 다음과 같습니다.

- 사업부 책임자
- 애플리케이션 소유자
- 아키텍트
- 운영 및 지원
- 클라우드 지원 팀
- 컴퓨팅, 스토리지 및 네트워크와 같은 특정 플랫폼 팀

자세한 검색을 위한 두 가지 접근 방식이 있습니다. 하향식 검색은 애플리케이션 또는 사용자로 시작하여 인프라로 이동합니다. 이는 애플리케이션 식별이 명확할 때 권장되는 접근 방식입니다. 반대로 상향식 검색은 인프라에서 시작하여 애플리케이션 또는 서비스와 해당 사용자에게 전달됩니다. 이 접근 방식은 인프라 팀이 마이그레이션 프로그램을 주도하고 application-to-infrastructure 매핑이 명확하지 않은 경우에 유용합니다. 일반적으로 둘 다 조합하여 사용할 가능성이 높습니다.

애플리케이션을 자세히 살펴보려면 기존 아키텍처 다이어그램을 사용하는 것이 좋습니다. 사용할 수 없는 경우 현재 지식을 기반으로 생성합니다. 간단한 리호스팅 또는 재배치 마이그레이션 전략의 경우에도 작업의 중요성을 과소평가하지 마세요. 아키텍처 다이어그램을 작성하면 클라우드에서 사소한 변경으로 신속하게 해결할 수 있는 비효율성을 식별할 수 있습니다.

하향식 접근 방식을 수행하는지 상향식 접근 방식을 수행하는지에 따라 초기 다이어그램은 애플리케이션 구성 요소와 서버 및 로드 밸런서와 같은 서비스 또는 인프라 구성 요소를 표시합니다. 주요 구성 요소와 인터페이스를 식별한 후 검색 도구 및 애플리케이션 성능 모니터링 도구의 프로그래밍 방식으로 데이터를 사용하여 검증합니다. 도구는 종속성 분석을 지원하고 구성 요소 간의 통신 정보를 제공해야 합니다. 이 애플리케이션을 구성하는 각 구성 요소를 식별해야 합니다. 다음으로 내부 및 외부의 다른 애플리케이션 및 서비스에 대한 종속성을 문서화합니다.

종속성과 매핑을 검증하는 도구가 없는 경우 수동 접근 방식이 필요합니다. 예를 들어 인프라 구성 요소에 로그인하고 스크립트를 실행하여 열린 포트 및 설정된 연결과 같은 통신 정보를 수집할 수 있습니다. 마찬가지로 실행 중인 프로세스와 설치된 소프트웨어를 식별할 수 있습니다. 수동 검색에 필요한 노력을 과소평가하지 마세요. 프로그래밍 방식 도구는 더 큰 간격(일반적으로 작은 백분율)으로 발생하는 종속성을 제외하고 며칠 내에 대부분의 종속성을 캡처하고 보고할 수 있습니다. 수동 검색은 모든 데이터 포인트를 수집하고 병합하는 데 몇 주가 걸릴 수 있으며 오류와 누락 데이터가 발생하기 쉽습니다.

우선 순위가 지정된 각 애플리케이션 및 매핑된 인프라에 대한 [데이터 요구 사항](#) 섹션에 지정된 정보를 가져옵니다. 다음으로 다음 설문지를 사용하여 자세한 평가 프로세스를 안내합니다. 식별된 이해관계자와 만나 이러한 질문에 대한 답변을 논의합니다.

## 일반

- 이 애플리케이션의 중요도 수준은 어느 정도입니까? 수익이 발생하나요? 비즈니스 전략 또는 지원 비즈니스 애플리케이션입니까? 다른 시스템에서 공유하는 핵심 인프라 서비스입니까?
- 이 애플리케이션에 대해 진행 중인 변환 프로젝트가 있습니까?
- 내부 또는 외부 대상 애플리케이션입니까?

## 아키텍처

- 현재 아키텍처 유형(예: SOA, 마이크로서비스, 모놀리스)은 무엇입니까? 아키텍처에는 몇 개의 티어가 있습니까? 긴밀하게 결합되어 있습니까, 아니면 느슨하게 결합되어 있습니까?
- 구성 요소는 무엇입니까(예: 컴퓨팅, 데이터베이스, 원격 스토리지, 로드 밸런서, 캐싱 서비스)?
- APIs란 무엇입니까? API 이름, 작업, URLs, 포트 및 프로토콜을 포함하여 이에 대해 설명합니다.

- 구성 요소와 이 애플리케이션 또는 서비스 간에 허용되는 최대 지연 시간은 얼마입니까?

## 운영

- 이 애플리케이션은 어떤 위치에서 작동하나요?
- 애플리케이션과 인프라는 누가 운영하나요? 내부 또는 AWS 파트너 팀에서 운영하나요?
- 이 애플리케이션이 다운되면 어떻게 되나요? 누가 영향을 받나요? 어떤 영향이 있나요?
- 사용자 또는 고객은 어디에 있나요? 애플리케이션에 어떻게 액세스하나요? 동시 사용자 수는 몇 명입니까?
- 마지막 기술 새로 고침은 언제였나요? 향후 새로 고침이 예약되나요? 그렇다면 언제입니까?
- 이 애플리케이션에 대해 알려진 위험과 문제는 무엇입니까? 중단, 중간 심각도 및 심각도가 높은 인시던트의 기록은 무엇입니까?
- 사용 주기(영업 시간)는 어떻게 됩니까? 운영 시간대는 어떻게 됩니까?
- 변경 고정 기간은 어떻게 됩니까?
- 이 애플리케이션을 모니터링하는 데 사용되는 솔루션은 무엇입니까?

## 성능

- 수집된 성능 정보는 무엇을 표시하나요? 사용량이 급증하거나 일정하고 예측 가능한가요? 사용 가능한 성능 데이터의 기간, 간격 및 날짜는 어떻게 됩니까?
- 이 애플리케이션의 일부이거나 이 애플리케이션과 상호 작용하는 예약된 배치 작업이 있습니까?

## 소프트웨어 수명 주기

- 현재 변화율(주간, 월간, 분기별 또는 연간)은 얼마입니까?
- 개발 수명 주기(예: 테스트, 개발, QA, UAT, 사전 프로덕션, 프로덕션)는 무엇입니까?
- 애플리케이션 및 인프라의 배포 방법은 무엇입니까?
- 배포 도구란 무엇입니까?
- 이 애플리케이션 또는 인프라는 지속적 통합(CI)/지속적 전달(CD)을 사용하나요? 자동화 수준은 어느 정도입니까? 수동 작업이란 무엇입니까?
- 애플리케이션 및 인프라의 라이선스 요구 사항은 무엇입니까?
- 서비스 수준 계약(SLA)이란 무엇입니까?

- 현재 테스트 메커니즘은 무엇입니까? 테스트 단계는 무엇입니까?

## 마이그레이션

- 마이그레이션 고려 사항은 무엇입니까?

이때이 애플리케이션을 마이그레이션할 때 고려해야 할 사항에 유의하세요. 보다 완전하고 정확한 평가를 위해 다른 이해관계자로부터이 질문에 대한 답변을 얻습니다. 그런 다음 이들의 지식과 의견을 대조합니다.

## 복원력

- 현재 백업 방법은 무엇입니까? 백업에 사용되는 제품은 무엇입니까? 백업 일정은 어떻게 됩니까? 백업 보존 정책이란 무엇입니까?
- 현재 Recovery Point Objective(RPO) 및 Recovery Time Objective(RTO)는 무엇입니까?
- 이 애플리케이션에 재해 복구(DR) 계획이 있습니까? 그렇다면 DR 솔루션은 무엇입니까?
- 마지막 DR 테스트는 언제였나요?

## 보안 및 규정 준수

- 이 애플리케이션에 적용되는 규정 준수 및 규제 프레임워크는 무엇입니까? 마지막 및 다음 감사 날짜는 언제입니까?
- 이 애플리케이션은 민감한 데이터를 호스팅하나요? 데이터 분류란 무엇입니까?
- 데이터가 전송 중 또는 저장 시 암호화됩니까, 아니면 둘 다 암호화됩니까? 암호화 메커니즘이란 무엇입니까?
- 이 애플리케이션은 SSL 인증서를 사용하나요? 발급 기관이란 무엇입니까?
- 사용자, 구성 요소 및 기타 애플리케이션 및 서비스에 대한 인증 방법은 무엇입니까?

## 데이터베이스 수

- 이 애플리케이션에서는 어떤 데이터베이스를 사용하나요?
- 데이터베이스에 대한 일반적인 동시 연결 수는 몇 개입니까? 최소 연결 수와 최대 연결 수는 얼마입니까?
- 연결 방법은 무엇입니까(예: JDBC, ODBC)?

- 연결 문자열이 문서화되어 있습니까? 그렇다면 어디에 있습니까?
- 데이터베이스 스키마란 무엇입니까?
- 데이터베이스가 사용자 지정 데이터 형식을 사용하나요?

## 종속성

- 구성 요소 간의 종속성은 무엇입니까? 해결할 수 없고 구성 요소를 함께 마이그레이션해야 하는 종속성을 기록해 둡니다.
- 구성 요소가 여러 위치로 분할되나요? 이러한 위치(예: WAN, VPN) 간의 연결은 무엇입니까?
- 다른 애플리케이션 또는 서비스에 대한 이 애플리케이션의 종속성은 무엇입니까?
- 운영 종속성이란 무엇입니까? 예를 들어, 패치 기간과 같은 유지 관리 및 릴리스 주기입니다.

## AWS 애플리케이션 설계 및 마이그레이션 전략

애플리케이션의 미래 상태를 설계하고 문서화하는 것은 주요 마이그레이션 성공 요인입니다. 아무리 단순하든 복잡하든 모든 유형의 마이그레이션 전략에 대한 설계를 만드는 것이 좋습니다. 설계를 생성하면 아키텍처가 변경될 것으로 예상되지 않는 경우에도 애플리케이션을 최적화할 수 있는 잠재적 차단기, 종속성 및 기회가 표시됩니다.

또한 마이그레이션 전략 렌즈를 AWS 사용하여 애플리케이션의 미래 상태에 접근하는 것이 좋습니다. 이 단계에서는 이 마이그레이션의 AWS 결과로 애플리케이션이 어떻게 보일지 정의해야 합니다. 결과 설계는 마이그레이션 후 추가 진화를 위한 기반 역할을 합니다.

다음 목록에는 설계 프로세스를 지원하는 리소스가 포함되어 있습니다.

- [AWS 아키텍처 센터](#)는 AWS Well-Architected 프레임워크와 같은 도구와 지침을 결합합니다. 또한 애플리케이션에 사용할 수 있는 참조 아키텍처를 제공합니다.
- [Amazon Builders' Library](#)에는 Amazon이 소프트웨어를 빌드하고 운영하는 방법에 대한 여러 리소스가 포함되어 있습니다.
- [AWS Solutions Library](#)는 수십 가지 기술 및 비즈니스 문제에 AWS대해에서 심사한 클라우드 기반 솔루션 모음을 제공합니다. 여기에는 대규모 참조 아키텍처 모음이 포함됩니다.
- [AWS 권장 가이드](#)는 설계 프로세스 및 마이그레이션 모범 사례를 지원하는 전략, 가이드 및 패턴을 제공합니다.
- [AWS Documentation](#)에는 사용 설명서 및 API 참조를 포함한 AWS 서비스에 대한 정보가 포함되어 있습니다.

- [시작하기 리소스 센터](#)는 빌드를 시작할 수 있도록 몇 가지 실습 자습서와 기초를 배울 수 있는 심층 분석을 제공합니다 AWS.

클라우드 여정의 현재 위치에 따라 AWS 근거가 이미 존재할 수 있습니다. 이러한 AWS 기반에는 다음이 포함됩니다.

- AWS 리전 가 식별되었습니다.
- 계정이 생성되었거나 온디맨드로 가져올 수 있습니다.
- 일반 네트워킹이 구현되었습니다.
- 기본 AWS 서비스가 계정 내에 배포되었습니다.

반대로 프로세스 초기에는 AWS 기초가 아직 설정되지 않았을 수 있습니다. 설정된 기반이 없으면 애플리케이션 설계 범위가 제한되거나 이를 정의하기 위한 추가 작업이 필요할 수 있습니다. 이 경우 애플리케이션 설계 작업과 함께 랜딩 존의 기본 설계를 정의하고 구현하는 것이 좋습니다. 애플리케이션 설계는 AWS 계정 구조, 네트워킹, Virtual Private Cloud(VPCs), Classless Inter-Domain Routing(CIDR) 범위, 공유 서비스, 보안 및 클라우드 운영과 같은 요구 사항을 식별하는 데 도움이 됩니다.

[AWS Control Tower](#)는 랜딩 존이라는 안전한 다중 계정 AWS 환경을 설정하고 관리하는 가장 쉬운 방법을 제공합니다. 이를 사용하여 랜딩 존을 AWS Control Tower 생성합니다. AWS Organizations 랜딩 존은 클라우드로 이동할 때 수천 명의 고객과 협력하는 AWS 모범 사례 기반 경험을 지속적으로 계정 관리 및 거버넌스하고 구현합니다.

## 애플리케이션 미래 상태

먼저 애플리케이션의 초기 마이그레이션 전략을 수립합니다. 이 시점에서 전략은 미래 상태 설계의 일부로 변경될 수 있으므로 초기 전략으로 간주되어 잠재적 제한을 발견할 수 있습니다. 초기 가정을 검증하려면 [6Rs 의사 결정 트리를 참조하세요](#). 또한 잠재적 마이그레이션 단계를 문서화합니다. 예를 들어 애플리케이션은 단일 이벤트로 마이그레이션됩니까(모든 구성 요소가 동시에 마이그레이션됨)? 아니면 단계별 마이그레이션입니까(일부 구성 요소는 나중에 마이그레이션됨)?

특정 애플리케이션에 대한 마이그레이션 전략은 고유하지 않을 수 있습니다. 이는 여러 R 유형을 사용하여 애플리케이션 구성 요소를 마이그레이션할 수 있기 때문입니다. 예를 들어 초기 접근 방식은 변경 없이 애플리케이션을 리프트 앤 시프트하는 것일 수 있습니다. 그러나 애플리케이션의 구성 요소는 다양한 처리가 필요할 수 있는 다양한 인프라 자산에 있을 수 있습니다. 예를 들어 애플리케이션은 각각 별도의 서버에서 실행되는 세 가지 구성 요소로 구성되며, 서버 중 하나는 클라우드에서 지원되지 않는 레거시 운영 체제를 실행합니다. 이 구성 요소에는 리플랫폼 접근 방식이 필요한 반면, 지원되는 서

버 버전에서 실행되는 다른 두 구성 요소는 리호스팅할 수 있습니다. 마이그레이션 중인 각 애플리케이션 구성 요소 및 관련 인프라에 마이그레이션 전략을 할당하는 것이 중요합니다.

그런 다음 컨텍스트와 문제를 문서화하고 현재 상태를 정의하는 기존 아티팩트를 연결합니다.

- 이 애플리케이션을 마이그레이션하는 이유는 무엇입니까?
- 제안된 변경 사항은 무엇입니까?
- 어떤 이점이 있나요?
- 주요 위험이나 차단 요인이 있나요?
- 현재 단점은 무엇인가요?
- 범위 내 및 범위 밖이란 무엇입니까?

## 반복성

설계 작업 전반에 걸쳐이 애플리케이션에 대한이 솔루션과 아키텍처를 다른 애플리케이션에 재사용할 수 있는 방법을 고려합니다. 이 솔루션을 일반화할 수 있습니까?

## 요구 사항

보안을 포함하여이 애플리케이션의 기능 및 비기능 요구 사항을 문서화합니다. 여기에는 선택한 마이그레이션 전략에 따라 현재 및 미래 상태 요구 사항이 포함됩니다. 자세한 애플리케이션 평가 중에 수집된 정보를 사용하여이 프로세스를 안내합니다.

## To-be 아키텍처

이 애플리케이션의 향후 아키텍처를 설명합니다. 소스 환경(온프레미스) 및 대상 AWS 환경(예: 대상, 계정 AWS 리전, VPCs 및 가용 영역)의 구성 요소가 포함된 재사용 가능한 다이어그램 템플릿을 생성하는 것이 좋습니다.

마이그레이션 중인 구성 요소와 새 구성 요소 테이블을 생성합니다. 이 애플리케이션과 상호 작용하는 다른 애플리케이션 및 서비스(온프레미스 또는 클라우드)를 포함합니다.

다음 표에는 예제 구성 요소가 나열되어 있습니다. 참조 아키텍처 또는 검증된 구성을 나타내지 않습니다.

이름	설명	세부 정보
애플리케이션	외부 서비스(인바운드 연결)	서비스는 노출된 API의 데이터를 사용합니다.
DNS	이름 확인(내부)	기존 계정 설정의 일부로 배포된 Amazon Route 53
Application Load Balancer	백엔드 서비스 간에 트래픽을 분산합니다.	온프레미스 로드 밸런서를 대체합니다. 풀 A를 마이그레이션합니다.
애플리케이션 보안	DdoS 보호	를 사용하여 구현됨 AWS Shield
보안 그룹	가상 방화벽	포트 443(인바운드)의 애플리케이션 인스턴스에 대한 액세스를 제한합니다.
Server A	프런트엔드	Amazon Elastic Compute Cloud(Amazon EC2)를 사용하여 리호스팅합니다.
Server B	프런트엔드	Amazon EC2를 사용하여 리호스팅합니다.
서버 C	애플리케이션 로직	Amazon EC2를 사용하여 리호스팅합니다.
서버 D	애플리케이션 로직	Amazon EC2를 사용하여 리호스팅합니다.
Amazon Relational Database Service(RDS) – Amazon Aurora	Database	서버 E 및 F를 대체합니다.
모니터링 및 알림	변경 제어	Amazon CloudWatch
감사 로깅	변경 제어	AWS CloudTrail

이름	설명	세부 정보
패치 및 원격 액세스	정비	AWS Systems Manager
리소스 액세스	보안 액세스 제어	AWS Identity and Access Management (IAM)
Authentication	사용자 액세스	Amazon Cognito
인증서	SSL/TLS	AWS Certificate Manager
API 1	외부 API	Amazon API Gateway
객체 스토리지	이미지 호스팅	Amazon Simple Storage Service(Amazon S3)
자격 증명	자격 증명 관리 및 호스팅	AWS Secrets Manager
AWS Lambda 함수	데이터베이스 자격 증명 및 API 키 검색	AWS Lambda
인터넷 게이트웨이	아웃바운드 인터넷 액세스	VPC에 대한 인터넷 게이트웨이
프라이빗 서브넷 1	백엔드 및 DB	가용 영역 1 – VPC 1
프라이빗 서브넷 2	백엔드 및 DB	가용 영역 2 – VPC 1
퍼블릭 서브넷 1	프론트엔드	가용 영역 1 – VPC 1
퍼블릭 서브넷 2	프론트엔드	가용 영역 2 – VPC 1
백업 서비스	데이터베이스 및 EC2 인스턴스 백업	AWS Backup
DR	Amazon EC2 복원력	AWS Elastic Disaster Recovery

구성 요소를 식별한 후 원하는 도구를 사용하여 다이어그램에 구성 요소를 표시합니다. 애플리케이션 소유자, 엔터프라이즈 아키텍트, 플랫폼 및 마이그레이션 팀을 포함한 주요 애플리케이션 이해관계자와 초기 설계를 공유합니다. 다음 질문을 하는 것이 좋습니다.

- 팀이 일반적으로 설계에 동의하나요?
- 운영 팀이 지원할 수 있나요?
- 설계를 발전시킬 수 있나요?
- 다른 옵션이 있나요?
- 설계가 아키텍처 표준 및 보안 정책을 준수하나요?
- 누락된 구성 요소가 있습니까(예: 코드 리포지토리, CI/CD 도구, VPC 엔드포인트)?

## 아키텍처 결정

설계 프로세스의 일부로 전체 아키텍처 또는 특정 부분에 대한 더 많은 옵션을 찾을 수 있습니다. 이러한 옵션을 기본 설정 또는 선택한 옵션의 근거와 함께 문서화합니다. 이러한 결정은 아키텍처 결정으로 문서화할 수 있습니다.

새 독자가 한 옵션을 다른 옵션보다 사용하기로 결정한 이면의 옵션과 이유를 이해할 수 있도록 기본 옵션을 충분히 자세히 나열하고 설명해야 합니다.

## 소프트웨어 수명 주기 환경

현재 환경에 대한 변경 사항을 문서화합니다. 예를 들어 테스트 및 개발 환경은에서 다시 생성되며 마이그레이션 AWS 되지 않습니다.

## 태그 지정

각 인프라 구성 요소에 대한 필수 및 권장 태그 지정과 이 설계의 태그 지정 값을 설명합니다.

## 마이그레이션 전략

설계의이 시점에서 마이그레이션 전략에 대한 초기 가정을 검증해야 합니다. 선택한 R 전략에 대한 합의가 있는지 확인합니다. 전체 애플리케이션 마이그레이션 전략과 개별 애플리케이션 구성 요소에 대한 전략을 문서화합니다. 앞서 언급했듯이 마이그레이션을 위해 다른 애플리케이션 구성 요소에 다른 R 유형이 필요할 수 있습니다.

또한 마이그레이션 전략을 주요 비즈니스 동인 및 결과에 맞게 조정합니다. 또한 다양한 마이그레이션 이벤트에서 구성 요소의 이동과 같은 마이그레이션에 대한 모든 단계별 접근 방식을 설명합니다.

6R 결정에 대한 자세한 내용은 [AWS Migration Hub 전략 권장 사항](#)을 참조하세요.

## 마이그레이션 패턴 및 도구

애플리케이션 및 인프라 구성 요소에 대해 정의된 마이그레이션 전략을 사용하면 이제 특정 기술 패턴을 탐색할 수 있습니다. 예를 들어와 같은 마이그레이션 도구를 사용하여 리호스팅 전략을 구현할 수 있습니다. [AWS Application Migration Service](#). 상태 또는 데이터를 복제할 필요가 없는 경우 Amazon Machine Image(AMI)와 애플리케이션 배포 파이프라인을 사용하여 애플리케이션을 재배포하여 동일한 결과를 얻을 수 있습니다.

마찬가지로 애플리케이션을 리플랫폼하거나 리팩터링(리아키텍트)하려면, [AWS App2Container](#), [AWS Database Migration Service \(AWS DMS\)](#), [AWS Schema Conversion Tool \(AWS SCT\)](#), 등의 도구를 사용할 수 있습니다. [AWS DataSync](#). 컨테이너화의 경우 [Amazon Elastic Container Service\(Amazon ECS\)](#), [Amazon Elastic Kubernetes Service\(Amazon EKS\)](#) 또는 [AWS Fargate](#)를 사용할 수 있습니다. 재구매 시 특정 제품에 대한 AMI 또는의 서비스형 소프트웨어(SaaS) 솔루션을 사용할 수 있습니다. [AWS Marketplace](#).

목표 달성에 사용할 수 있는 다양한 패턴과 옵션을 평가합니다. 장단점과 마이그레이션 운영 준비 상태를 고려합니다. 분석에 도움이 되도록 다음 질문을 사용합니다.

- 마이그레이션 팀이 이러한 패턴을 지원할 수 있나요?
- 비용과 혜택의 균형은 무엇입니까?
- 이 애플리케이션, 서비스 또는 구성 요소를 관리형 서비스로 이동할 수 있습니까?
- 이 패턴을 구현하기 위한 노력은 무엇입니까?
- 특정 패턴의 사용을 금지하는 규정 또는 규정 준수 정책이 있습니까?
- 이 패턴을 재사용할 수 있나요? 재사용 가능한 패턴이 선호됩니다. 그러나 패턴은 한 번만 사용되는 경우도 있습니다. 재사용 가능한 대체 패턴과 일회용 패턴의 작업 간의 균형을 고려하세요.

[AWS 권장 가이드](#)에는 다양한 마이그레이션 패턴과 기법이 포함되어 있습니다.

## 서비스 관리 및 운영

마이그레이션을 위한 애플리케이션 설계를 생성할 때는 운영 준비 상태를 AWS고려하세요. 애플리케이션 및 인프라 팀과 준비 요구 사항을 평가할 때는 다음 질문을 고려하세요.

- 운영할 준비가 되었나요?
- 인시던트 대응 절차가 정의되어 있습니까?

- 예상되는 서비스 수준 계약(SLA)은 무엇입니까?
- 의무 분리가 필요합니까?
- 여러 팀이 지원 작업을 조정할 준비가 되었습니까?
- 무엇에 대한 책임은 누구에게 있습니까?

## 전환 고려 사항

마이그레이션 전략과 패턴을 고려할 때 애플리케이션이 마이그레이션되는 시점에 알아야 할 중요 사항은 무엇입니까? 전환 계획은 설계 후 활동입니다. 그러나 예상할 수 있는 활동 및 요구 사항에 대한 고려 사항을 문서화합니다. 예를 들어 해당하는 경우 개념 증명을 수행하기 위한 요구 사항을 문서화하고 테스트, 감사 또는 검증 요구 사항을 간략하게 설명합니다.

## 위험, 가정, 문제 및 종속성

아직 해결되지 않은 모든 미해결 위험, 가정 및 잠재적 문제를 문서화합니다. 이러한 항목에 명확한 소유권을 할당하고 진행 상황을 추적하여 전체 설계 및 전략의 구현을 승인할 수 있습니다. 또한이 설계를 구현하기 위한 키 종속성을 문서화합니다.

## 실행 비용 추정

대상 AWS 아키텍처의 비용을 추정하려면 [AWS 요금 계산기](#)를 사용합니다. 설계에 정의된 대로 인프라 구성 요소를 추가하고 예상 실행 비용을 구합니다. 애플리케이션 구성 요소에 필요하고 사용할 AWS 서비스에 아직 포함되지 않은 소프트웨어 라이선스의 팩터입니다.

## 포트폴리오 분석 및 마이그레이션 계획

이 평가 단계는 포트폴리오 검색 및 [초기 계획 섹션에서 시작된 포트폴리오 수준 검색 및 분석](#)을 완료하는 데 중점을 둡니다. 목표는 애플리케이션 및 인프라의 초기 포트폴리오에 대한 기준을 반복하고 설정하는 것입니다. 이 기준에는 모든 종속성 식별, 마이그레이션을 위한 합리화 모델 반복, 자세한 비즈니스 사례 생성, 마이그레이션 웨이브 계획 요약이 포함됩니다. 따라서 필요한 데이터 충실도가 높아집니다. 이 단계에는 시간 투자가 필요합니다. 평가 결과를 가속화하려면 검색 도구와 같은 프로그래밍 방식의 데이터 소스를 최대한 많이 사용하는 것이 좋습니다.

이 단계의 주요 결과는 다음과 같습니다.

- 충실도가 높은 애플리케이션 및 인프라 인벤토리
- 각 애플리케이션에 대한 상위 수준 마이그레이션 전략
- 신뢰도가 높은 마이그레이션 웨이브 플랜
- 세부 비즈니스 사례

## 전체 평가 데이터 요구 사항 이해

다음 표에서는 마이그레이션의 애플리케이션 및 관련 인프라에 대한 전체 포트폴리오 보기를 얻는 데 필요한 정보를 설명합니다.

테이블은 다음 약어를 사용합니다.

- R, 필수
- O, 선택 사항의 경우
- 해당 없음, 해당 없음

### 애플리케이션

속성 이름	설명	인벤토리 및 우선 순위 지정	세부 비즈니스 사례	권장 충실도 수준 (최소)
고유한 식별자	애플리케이션 ID를 예로 들 수 있습니다. 일반적으로 기존 CMDBs	R	R	높음

속성 이름	설명	인벤토리 및 우선 순위 지정	세부 비즈니스 사례	권장 충실도 수준 (최소)
	또는 기타 내부 인벤토리 및 제어 시스템에서 사용할 수 있습니다. 조직에 고유 ID가 정의되지 않은 경우 고유 IDs를 생성하는 것이 좋습니다.			
애플리케이션 이름	이 애플리케이션이 조직에 알려진 이름입니다. 해당하는 경우 상용 off-the-shelf(COTS) 공급업체 및 제품 이름을 포함합니다.	R	R	높음
COTS입니까?	예 또는 아니요. 상용 애플리케이션인지 내부 개발인지 여부	R	R	높음
COTS 제품 및 버전	상용 소프트웨어 제품 이름 및 버전	R	R	높음
설명	기본 애플리케이션 함수 및 컨텍스트	R	R	높음

속성 이름	설명	인벤토리 및 우선 순위 지정	세부 비즈니스 사례	권장 충실도 수준 (최소)
중요도	예: 전략적 또는 수익 창출 애플리케이션 또는 중요한 함수 지원	R	R	높음
Type	예: 데이터베이스, 고객 관계 관리(CRM), 웹 애플리케이션, 멀티미디어, IT 공유 서비스	R	R	높음
환경	예: 프로덕션, 사전 프로덕션, 개발, 테스트, 샌드박스	R	R	높음
규정 준수 및 규제	워크로드에 적용되는 프레임워크(예: HIPAA, SOX, PCI-DSS, ISO, SOC, FedRAMP) 및 규제 요구 사항	R	R	높음

속성 이름	설명	인벤토리 및 우선 순위 지정	세부 비즈니스 사례	권장 충실도 수준 (최소)
종속성	내부 및 외부 애플리케이션 또는 서비스에 대한 업스트림 및 다운스트림 종속성. 운영 요소와 같은 비기술적 종속성 (예: 유지 관리 주기).	R	O	높음
인프라 매핑	애플리케이션을 구성하는 물리적 및/또는 가상 자산에 매핑	R	R	높음
라이선스	상품 소프트웨어 라이선스 유형(예: Microsoft SQL Server Enterprise)	R	R	중간-높음
비용	소프트웨어 라이선스, 소프트웨어 운영 및 유지 관리 비용	해당 사항 없음	R	중간-높음
사업부	예: 마케팅, 재무, 영업	R	R	높음
소유자 세부 정보	애플리케이션 소유자의 연락처 정보	R	R	높음
DR 정보	재해 복구 구성 요소	R	R	높음

속성 이름	설명	인벤토리 및 우선 순위 지정	세부 비즈니스 사례	권장 충실도 수준 (최소)
마이그레이션 전략	예를 들어 로 마이그레이션하기 위한 6R 중 하나 AWS	R	R	높음
지원 티켓	중단, 속도 저하, 트랜잭션 제한 및 배치 기간 초과 의 생산성 및 재정적 영향을 평가하는 데 도움이 되는 12~24개월의 데이터	O	R	중간

인프라

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
고유한 식별자	서버 ID를 예로 들 수 있습니다. 일반적으로 기존 CMDBs 또는 기타 내부 인벤토리 및 제어 시스템에서 사용할 수 있습니다. 조직에 고유 ID가 정의되지 않은 경우 고유 IDs를 생성하는 것이 좋습니다.	R	R	높음

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
네트워크 이름	네트워크의 자산 이름(예: 호스트 이름)	R	R	높음
DNS 이름(정규화된 도메인 이름 또는 FQDN)	DNS 이름	R	O	높음
IP 주소 및 넷마스크	내부 및/또는 퍼블릭 IP 주소	R	R	높음
애셋 유형	예: 물리적 또는 가상 서버, 하이퍼바이저, 컨테이너, 디바이스, 데이터베이스 인스턴스	R	R	높음
제품 이름	상용 공급업체 및 제품 이름(예: VMware ESXi, IBM Power Systems, Exadata)	R	R	높음
운영 체제	예: REHL 8, Windows Server 2019, AIX 6.1	R	R	높음
구성	할당된 CPU, 코어 수, 코어당 스레드 수, 총 메모리, 스토리지, 네트워크 카드	R	R	높음

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
사용률	CPU, 메모리, 스토리지 피크 및 평균. 데이터베이스 인스턴스 처리량.	R	R	높음
라이선스	상품 라이선스 유형(예: RHEL Standard)	R	R	높음
공유 인프라입니까?	인증 공급자, 모니터링 시스템, 백업 서비스 및 유사한 서비스와 같은 공유 서비스를 제공하는 인프라 서비스를 나타내는 예 또는 아니요	R	R	높음
애플리케이션 매핑	이 인프라에서 실행되는 애플리케이션 또는 애플리케이션 구성 요소	R	R	높음
비용	하드웨어, 유지 관리, 운영, 스토리지(SAN, NAS, 객체), 운영 체제 라이선스, 랙 공간 공유, 데이터 센터 오버헤드 등 베어 메탈 서버의 전체 로드 비용	해당 사항 없음	R	중간-높음

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
예상 데이터 전송량(인/아웃)	예를 들어, 30일 동안 하루에 걸쳐 인프라 자산당	O	R	중간

네트워크

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
파이프 크기(Mb/s), 중복성(Y/N)	현재 WAN 링크 사양(예: 1000Mb/s 중복)	R	R	중간-높음
링크 사용률	최대 및 평균 사용률, 아웃바운드 데이터 전송(GB/월)	R	R	중간-높음
지연 시간(ms)	연결된 위치 간의 현재 지연 시간입니다.	R	O	높음
비용	현재 월별 비용	해당 사항 없음	R	중간-높음

마이그레이션

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
리호스팅	각 워크로드(인일), 일일 고객 및 파트너 비용 비율, 도구 비용, 위	해당 사항 없음	R	중간-높음

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
	크로드 수에 대한 고객 및 파트너 노력			
리플랫폼	각 워크로드(인 일), 일일 고객 및 파트너 비용 비율, 워크로드 수에 대한 고객 및 파트너 노력	해당 사항 없음	R	중간-높음
리팩터링	각 워크로드(인 일), 일일 고객 및 파트너 비용 비율, 워크로드 수에 대한 고객 및 파트너 노력	해당 사항 없음	R	중간-높음
완료	서버 수, 평균 폐기 비용	해당 사항 없음	R	중간-높음
랜딩 존	기존 재사용(Y/N), AWS 리전 필요 목록, 비용	해당 사항 없음	R	중간-높음
사람 및 변화	클라우드 운영 및 개발에서 훈련할 직원 수, 1인당 훈련 비용, 1인당 훈련 시간 비용	해당 사항 없음	R	중간-높음
지속 시간	범위 내 워크로드 마이그레이션 기간(개월)	O	R	중간-높음

속성 이름	설명	인벤토리 및 우선 순위 지정	비즈니스 사례	권장 충실도 수준 (최소)
병렬 비용	마이그레이션 중에 있는 그대로 비용을 제거할 수 있는 기간 및 비율	해당 사항 없음	R	중간-높음
	마이그레이션 중에 AWS 제품 및 서비스와 기타 인프라 비용이 도입 되는 기간 및 속도	해당 사항 없음	R	중간-높음

## 애플리케이션 포트폴리오의 기준 설정

신뢰도가 높은 마이그레이션 웨이브 플랜을 생성하려면 애플리케이션 포트폴리오 및 관련 인프라에 대한 기준을 설정해야 합니다. 포트폴리오 기준은 기술 종속성 및 마이그레이션 전략을 포함하여 마이그레이션 범위에 대한 포괄적인 보기를 제공합니다. 포트폴리오 기준은 마이그레이션 범위에 속하는 애플리케이션과 [전체 평가 데이터 요구 사항 이해 섹션에 설명된 데이터](#) 포인트가 수집되는지에 대한 명확성을 제공합니다. 마찬가지로 연결된 모든 인프라(컴퓨팅, 스토리지 네트워크)가 이해되고 애플리케이션에 매핑됩니다.

기술 종속성은 다음 네 가지 범주로 설명할 수 있습니다.

- Application-to-infrastructure 종속성은 소프트웨어와 물리적 또는 가상 하드웨어 간의 연결을 설정합니다. 예를 들어 CRM 애플리케이션과 해당 애플리케이션이 설치된 가상 머신 간에는 종속성이 있습니다.
- 애플리케이션 구성 요소 종속성은 서로 다른 인프라 자산에서 실행되는 구성 요소가 상호 작용하는 방식을 설명합니다. 애플리케이션 구성 요소 종속성의 예로는 가상 머신에서 실행되는 웹 프론트 엔드, 다른 가상 머신에서 실행되는 애플리케이션 계층, 데이터베이스 클러스터에서 실행되는 데이터베이스가 있습니다.
- Application-to-application 종속성은 애플리케이션 또는 애플리케이션 구성 요소와 다른 애플리케이션 또는 해당 구성 요소 간의 상호 작용과 관련이 있습니다. application-to-application 종속성의 예로

는 결제 처리 애플리케이션과 주식 관리 애플리케이션이 있습니다. 이러한 애플리케이션은 독립적이지만 정의된 API 작업을 사용하여 지속적으로 상호 작용합니다.

- 인프라 서비스 자체가 애플리케이션이라는 점을 고려할 때 Application-to-infrastructure 서비스 종속성은 기술적으로 application-to-application 종속성입니다. 그러나 별도로 분류하는 것이 좋습니다. 주된 이유는 인프라 서비스가 일반적으로 많은 애플리케이션에서 공유되므로 종속성 추적이 길기 때문입니다. 또한 일반적으로 다른 마이그레이션 전략과 패턴을 따릅니다. 예를 들어 로드 밸런서는 여러 애플리케이션에 대한 밸런싱 풀을 포함할 수 있습니다. 중요한 것은 종속 애플리케이션과 함께 개별적으로 마이그레이션될 수 있는 풀에 대한 종속성이며, 로드 밸런서 자체가 유지되거나 retired.In 추가하여 application-to-infrastructure 서비스 종속성을 개별화하면 잘못된 종속성 그룹을 방지하는 데 도움이 됩니다. 거짓 종속성 그룹은 여러 비즈니스 애플리케이션이 함께 그룹화되는 경우입니다. 즉, 인프라 서비스에 대한 일반적인 종속성이 있는 동시에 마이그레이션해야 합니다. 예를 들어 Active Directory와 같은 인증 서비스는 대규모 애플리케이션 그룹과 연결될 수 있습니다. 핵심은 이러한 애플리케이션에 개별적으로 접근하고 클라우드 환경에서와 같은 서비스를 활성화하여 종속성을 해결하는 AWS Directory Service for Microsoft Active Directory입니다.

포트폴리오의 기준을 설정할 때 각 애플리케이션 구성 요소에 대한 마이그레이션 전략을 확인하는 것이 좋습니다. 마이그레이션 전략은 마이그레이션을 위한 6R 중 하나가 됩니다([6R 마이그레이션 전략 반복](#) 섹션 참조). 포트폴리오 기준에서 6R 중 하나를 각 애플리케이션과 연결해야 합니다. 또한 6R 전략은 애플리케이션의 각 인프라 구성 요소와 연결되어야 합니다.

종속성 및 마이그레이션 전략을 포함하여 포트폴리오의 기준 버전을 설정하려면 자동 검색 도구를 사용합니다([검색 도구의 필요성 평가](#) 참조). 애플리케이션 소유자 및 인프라 팀과 같은 주요 이해관계자로부터 수집한 정보로 데이터를 보완합니다. 이 단계의 데이터 [요구 사항 섹션에](#) 설명된 속성 및 충실도 수준과 일치하는 전체 포트폴리오 인벤토리를 얻을 때까지 데이터를 계속 수집합니다. 결과 데이터 세트는 마이그레이션을 추진하는 데 중요한 역할을 합니다.

마이그레이션 범위와 사용 가능한 도구에 따라이 활동을 완료하는 데 몇 주가 걸릴 수 있습니다.

## 우선 순위 지정 기준 반복

마이그레이션 웨이브 계획을 생성하기 전에 애플리케이션 우선 순위 기준을 반복하여 파일럿 애플리케이션 선택에서 장기 웨이브 계획으로 전환하는 것이 좋습니다.

이전 단원에서는 간단한 클라우드 지원 애플리케이션의 우선 순위를 지정하는 기본 우선 순위 지정 기준을 도입했습니다([애플리케이션 우선 순위 지정](#) 참조). 이는 초기 단계에서 중요하지 않은 애플리케이션부터 시작하여 마이그레이션 프로세스를 구체화하고 학습한 교훈을 통합하는 것이 권장되기 때문입니다. 그러나이 단계에서 장기 계획을 생성하려면 애플리케이션을 마이그레이션하는 순서가 비즈니스

동인에 맞게 조정되어야 합니다. 새 기준을 적용하면 웨이브 계획을 위한 주요 입력이 될 애플리케이션의 새 순위가 생성됩니다.

애플리케이션 포트폴리오에서 사용 가능한 데이터 포인트를 검토하고 비즈니스 동인을 기반으로 애플리케이션 우선 순위를 결정할 속성을 선택합니다.

먼저 비즈니스 동인을 검증합니다([비즈니스 동인 및 기술 지침 원칙](#) 참조). 그런 다음 비즈니스 동인에 따라 마이그레이션을 위해 애플리케이션의 우선순위를 지정하는 데 도움이 되는 속성을 선택합니다.

다음 표에는 혁신을 위한 비즈니스 동인에 부합하는 우선순위 지정 기준의 예가 나와 있습니다.

속성 또는 데이터 포인트	가능한 값	점수(0~99)	중요도 또는 관련성 곱하기 인자
운영 체제	AIX	80	높음(1x)
	솔라리스	80	
	HP-UX	80	
	메인프레임	70	
	Windows	50	
	Linux	20	
비즈니스 중요도	높음	60	높음(1x)
	중간	40	
	낮음	20	
아키텍처	긴밀하게 결합됨	60	높음(1x)
	느슨하게 결합됨	20	
운영 모델	기존 - CI/CD 없음	60	중간-높음(0.8x)
	기본 CI/CD	40	
	전체 DevOps	20	

속성 또는 데이터 포인트	가능한 값	점수(0~99)	중요도 또는 관련성 곱하기 인자
컴퓨팅 인스턴스 수	1-3	60	중간-높음(0.8x)
	4-10	40	
	11개 이상	20	
마이그레이션 전략	리팩터링(리아키텍트)	70	중간(0.6x)
	리플랫폼	40	
	재구매	30	
	리호스팅	10	

다음 표에는 빠른 비용 절감을 위한 비즈니스 동인에 맞는 우선순위 지정 기준의 예가 나와 있습니다.

속성 또는 데이터 포인트	가능한 값	점수(0~99)	중요도 또는 관련성 곱하기 계수
데이터베이스 제품	Oracle	70	높음(1x)
	Microsoft SQL	70	
	기타	20	
운영 체제	Windows	70	높음(1x)
	Linux	70	
	기타	20	
CPU 사용률(평균)	36% 초과	60	높음(1x)
	36% 미만	40	
컴퓨팅 인스턴스 수	11개 이상	60	중간-높음(0.8x)

속성 또는 데이터 포인트	가능한 값	점수(0~99)	중요도 또는 관련성 곱하기 계수
	4-10	40	
	1-3	20	
마이그레이션 전략	만료	80	중간(0.6x)
	리호스팅	70	
	리플랫폼	50	
	리팩터링(리아키텍트)	10	

우선 순위 기준을 테스트하고 일반적으로 출력에 동의할 때까지 반복합니다. 기존 버전을 얻으려면 최소 3회 또는 4회의 반복이 필요합니다.

## 6Rs 마이그레이션 전략 선택 반복

이 단계에서는 6R 의사 결정 트리를 반복하고 발전시키는 것이 좋습니다. [마이그레이션을 위한 R 유형 결정](#) 섹션에는 기본 결정 트리가 도입되었습니다. 트리를 수정하고, 초기 파일럿 애플리케이션의 마이그레이션 전반에서 학습한 내용을 고려하고, 비즈니스 동인, 우선순위 기준 및 고유한 상황에 부합하는지 확인하는 것이 좋습니다. 샘플 애플리케이션으로 의사 결정 트리를 검증하고 여전히 예상 전략을 생성하는지 확인합니다. 그렇지 않으면 그에 따라 로직을 업데이트합니다. 결과 트리는 애플리케이션 포트폴리오의 기준을 설정하고 각 애플리케이션 구성 요소에 마이그레이션 전략을 할당하는 데 중요합니다.

이전 [6R 섹션에서](#) 설명한 대로 6R은 인프라에도 적용되며 그에 따라 할당하는 것도 중요합니다. 특정 애플리케이션 구성 요소에는 마이그레이션 전략이 있지만 인프라 수준에서 각 인프라 자산은 지원하는 애플리케이션 구성 요소에 대해 설정된 전략과 다를 수 있는 지정된 마이그레이션 전략을 따릅니다.

6Rs 의사 결정 트리는 애플리케이션 구성 요소에만 적용됩니다. 인프라에 대한 마이그레이션 전략은 애플리케이션에 대해 선택한 전략에서 파생됩니다. 예를 들어 리플랫폼될 애플리케이션 구성 요소의 경우 해당 구성 요소를 호스팅하는 현재 인프라는 사용 중지될 수 있습니다.

마이그레이션 전략이 각 애플리케이션 구성 요소 및 관련 인프라에 할당되었는지 확인합니다. 이 정보는 필요한 노력, 용량 및 기술을 추정할 때와 마이그레이션 웨이브 계획을 생성할 때 중요한 요소가 됩니다.

6R 결정에 대한 자세한 내용은 [AWS Migration Hub 전략 권장 사항](#)을 참조하세요.

## 웨이브 계획

웨이브 계획에서 종속성 그룹은 해결할 수 없는 기술적 및 비기술적 종속성이 있는 애플리케이션 및 인프라 모음입니다. 이러한 종속성으로 인해 종속성 그룹의 애플리케이션과 인프라는 동시에 또는 특정 날짜에 마이그레이션해야 합니다. 예를 들어, 지연 시간이 짧은 요구 사항이나 트래픽이 많은 볼륨 및 복잡한 쿼리가 있는 가상 머신에서 실행되는 애플리케이션과 별도의 가상 머신에서 실행되는 데이터 베이스는 클라우드에서 한 구성 요소를 운영하고 온프레미스에서 다른 구성 요소를 운영하는 대신 함께 마이그레이션될 가능성이 높습니다. 마찬가지로 지연 시간이 짧은 유사한 요구 사항이 있는 API를 통해 상호 작용하는 독립 애플리케이션도 동시에 마이그레이션됩니다.

마이그레이션 웨이브는 일반적으로 4~8주에 걸쳐 있으며 하나 이상의 마이그레이션 이벤트를 포함할 수 있습니다. 종속성 그룹은 웨이브로 결합되므로 웨이브에 하나 이상의 종속성 그룹이 포함될 수 있습니다. 웨이브에는 마이그레이션에 필요한 다른 활동도 포함되어 있습니다. 여기에는 AWS 인프라 설정(예: 랜딩 존, 보안 및 운영), 마이그레이션 도구, 데이터 복제, 전환 계획, 테스트 및 마이그레이션 후 지원과 같은 마이그레이션 활동이 포함됩니다.

성공을 측정하고 진행 상황을 추적하려면 성과 및 비즈니스 동인에 따라 웨이브를 조정해야 합니다. 이는 파도 지속 시간 및 파도에 포함된 종속성 그룹에도 영향을 미칩니다. 웨이브 완료에는 측정 가능한 업적이 반영되어야 합니다. 파도 계획은 기술 가이드 원칙과 같은 다른 요인도 결합할 수 있습니다. 예를 들어, 웨이브는 환경(예: 개발, 테스트, 프로덕션) 또는 마이그레이션 전략(예: 리호스팅 웨이브, 리플랫폼 웨이브)에 의해 정의될 수 있습니다.

효과적이고 신뢰도가 높은 마이그레이션 웨이브 플랜을 생성하려면 애플리케이션 포트폴리오, 관련 인프라(컴퓨팅, 스토리지, 네트워크), 종속성 매핑 및 마이그레이션 전략을 전체적으로 파악해야 합니다.

[애플리케이션 포트폴리오의 기준을 설정하는](#) 단원에서는 네 가지 범주의 기술 종속성을 설명했습니다. 이러한 종속성은 마이그레이션 웨이브 생성과 종속성 그룹의 정의에 기여합니다. 종속성 그룹은 종속성의 중요도에 따라 결정됩니다. 또한 비기술적 종속성을 고려해야 합니다. 예를 들어 애플리케이션 릴리스 일정, 유지 관리 기간, 월말 분기말 처리와 같은 주요 비즈니스 날짜가 웨이브 플랜에 영향을 미칩니다.

종속성이 소프트웨어인지 하드웨어인지 확인합니다. 소프트웨어 종속성은 두 개 이상의 자산 간 또는 자산과 제약 조건 간의 관계로, 구성 요소의 위치에 따라 달라지지 않습니다. 예를 들어 동일한 로컬 네트워크(또는 동일한 인프라)에서 작동하는 두 시스템은 해당 시스템 중 하나를 클라우드로 이동하여 분할할 수 있지만 다른 하나는 온프레미스에 남아 있습니다. 또 다른 예는 유지 관리 활동에 영향을 주지 않고 유지 관리 기간 동안 마이그레이션할 수 있는 시스템입니다.

하드 종속성은 위치에 따라 두 개 이상의 자산 또는 자산과 제약 조건 간의 관계입니다. 예를 들어 동일한 로컬 네트워크에서 작동하고 애플리케이션 서버와 데이터베이스 서버 간의 통신에 짧은 지연 시간에 크게 의존하는 두 시스템은 엄격한 종속성을 갖습니다. 이러한 시스템 중 하나만 클라우드로 이동하면 해결할 수 없는 기능 또는 성능 문제가 발생합니다. 마찬가지로 리소스 가용성(예: 마이그레이션을 수행하는 팀) 또는 지정된 기간에만 두 시스템을 마이그레이션할 수 있는 유지 관리 기간과 같은 운영 제약 조건과 같은 비기술적 이유로 인해 이러한 자산에 대한 엄격한 종속성이 발생할 수 있습니다.

마이그레이션 웨이브 플랜을 생성하려면 특수 검색 도구와 같은 신뢰할 수 있는 데이터 소스에서 종속성을 분석하여 종속성 그룹을 결정하고 이 정보를 애플리케이션 우선 순위 기준 및 운영 상황과 결합합니다. 우선 순위 순위 순위 상단의 애플리케이션은 초기 마이그레이션 웨이브를 대상으로 해야 합니다. 리소스 가용성, 위험 허용 범위, 비즈니스 및 기술적 제약 조건, 경험 및 기술을 기반으로 파도 용량(파도에 포함될 수 있는 애플리케이션 수)을 결정합니다. 전문 서비스 또는 AWS 마이그레이션 역량 파트너와 협력하여 AWS 프로세스 전반에 걸쳐 전문가를 지원할 수 있습니다.

우선 순위 지정 기준은 애플리케이션을 클라우드로 이동하는 순서의 초기 표시입니다. 그러나 종속성 그룹은 지정된 시간에 이동할 애플리케이션의 실제 결정자입니다. 이는 우선 순위가 높은 애플리케이션이 순위 중간 또는 하단에 있는 애플리케이션에 대해 엄격한 종속성을 가질 수 있기 때문입니다.

마이그레이션 전략은 파도 구성에도 영향을 미칩니다. 예를 들어 몇 주 또는 몇 달의 분석, 설계, 테스트 및 준비가 필요할 수 있는 리팩터링 전략이 필요한 우선 순위가 높은 애플리케이션은 이후 웨이브에 배치될 가능성이 높습니다.

## 웨이브 플랜 생성

애플리케이션 웨이브를 마이그레이션하기 위한 사전 조건은 애플리케이션 포트폴리오 데이터와 웨이브에서 마이그레이션할 애플리케이션 그룹에 대한 자세한 애플리케이션 평가입니다. 세부 평가에는 웨이브의 애플리케이션 목록, 관련 인프라 세부 정보, 대상 설계 및 각 애플리케이션의 마이그레이션 전략이 포함되어야 합니다.

파도 소유권 및 거버넌스를 설정하는 것은 파도 작업, 프로그램 종속성, 변경 관리, 문제 및 위험을 관리하고 추적하는 데 중요합니다. 계획을 관리하기 위한 거버넌스 프레임워크가 마련되어 있는지 확인합니다.

웨이브 플랜을 간략하게 설명하려면 기본 웨이브 구성으로 시작합니다. 파도 내에서는 어떻게 되나요? 초기 입력이 정의되면 웨이브가 시작될 수 있습니다. 일반적으로 활동은 다음과 같습니다.

1. 전환 계획을 구체화합니다. 이 활동은 다른 내부 및 외부 팀과의 조정을 포함하여 마이그레이션 시점에 수행해야 하는 실행서와 단계를 간략하게 설명해야 합니다.
2. 롤백 계획을 구체화합니다. 문제가 발생할 경우 애플리케이션을 롤백하려면 어떻게 해야 하나요?

3. 대상 인프라를 준비합니다. 예를 들어 AWS 랜딩 존(, 보안AWS 계정, 네트워킹, 인프라 서비스, 기타 지원 인프라)을 생성하거나 확장할 수 있습니다.
4. 대상 인프라를 테스트합니다.
5. 마이그레이션 도구를 운영합니다. 예를 들어 복제 에이전트를 설치하고 데이터 전송을 시작합니다.
6. 전환 계획 및 런북 드라이 런을 수행합니다. 참여하는 모든 팀원을 그룹화하고 모든 단계를 미리 검토합니다.
7. 데이터 복제 및 인프라 배포를 모니터링합니다.
8. 에서 인프라 및 애플리케이션 운영 준비 상태를 확인합니다 AWS.
9. 보안 준비 상태를 확인합니다.
- 10.해당하는 경우 규정 준수 및 규제 요구 사항(예: 마이그레이션 전 및 마이그레이션 후 워크로드 검증)을 확인합니다.
- 11.애플리케이션을 로 마이그레이션 AWS 하고 가동 전 테스트를 수행합니다.
- 12.운영 팀과 마이그레이션 팀이 문제를 해결할 수 있고 최적화를 적용할 수 있는 3일 등의 기간 동안 마이그레이션 후 지원을 제공합니다.
- 13.마이그레이션 후 검토를 수행합니다. 학습한 내용을 문서화하고 향후 웨이브에 통합합니다.
- 14.보고를 위한 지표의 운영 인계 및 획득을 확인하여 웨이브 종료를 수행합니다.

이러한 각 활동의 소요 시간은 범위의 복잡성, 파도 용량, 관련된 사람 및 고유한 상황에 따라 결정됩니다. 가능하면 파도가 작을수록 지연 또는 마이그레이션 차단기의 영향을 줄일 수 있기 때문에 더 좋습니다. 팀과 함께 파도의 기본 지속 시간을 결정합니다.

다음으로 날짜를 분석하여 빈 웨이브의 초기 상위 수준 구조(아직 애플리케이션이 할당되지 않음)를 생성합니다. 다음 질문을 고려하세요.

- 총 마이그레이션 프로그램 길이는 얼마입니까?
- 기한은 어떻게 됩니까?
- 고정된 데이터 센터 종료 날짜가 있나요?
- 콜로케이션 계약 종료일이 있나요?
- 애플리케이션 및 인프라 새로 고침 주기란 무엇입니까?
- 애플리케이션 유지 관리 및 릴리스 주기는 어떻게 됩니까?
- 마이그레이션을 피해야 하는 날짜(예: 릴리스 및 유지 관리 주기, 연말, 공휴일, 월말 처리)가 있나요?

이러한 고려 사항을 고려하여 웨이브를 계획에 표시합니다. 마이그레이션 프로세스를 가속화하려면 가능하면 겹치는 웨이브를 사용하는 것이 좋습니다. 겹치는 파도의 핵심은 파도 내에서 발생하는 일을 정의하고 고려하는 것입니다. 일반적으로 배포 활동, 대상 인프라 검증 및 데이터 동기화는 웨이브의 상반기 동안 발생합니다. 하반기는 실제 마이그레이션, 테스트 및 운영 인계에 중점을 둡니다. 즉, 프로세스의 각 절반에 서로 다른 팀이 관여하며 효율성을 어느 정도 높일 수 있습니다. 예를 들어, 대상 인프라 준비에 관여한 팀이 작업을 완료하자마자 다음 웨이브의 요구 사항에 대한 작업을 시작할 수 있습니다. 일반적으로 대부분의 웨이브는 마이그레이션에 대한 공장과 유사한 접근 방식을 용이하게 하기 위해 길이와 구조가 비슷한 것이 좋습니다. 그러나 웨이브 계획 프로세스 중에 종속성 또는 운영 요구 사항을 충족하도록 지정된 웨이브의 크기를 확장할 수 있습니다.

그런 다음 식별된 종속성 그룹을 기반으로 포함할 수 있는 종속성 그룹 수를 기준으로 웨이브의 최대 크기를 결정합니다. 파도 크기는 일반적으로 위험 선호도(예: 병렬 변경을 허용할 수 있는 정도) 및 리소스 가용성(예: 사용 가능한 리소스, 기술 및 예산으로 병렬 변경을 수행할 수 있는 정도)에 따라 결정됩니다. 그러나 초기 계획 중에 리소스 요구 사항 및 가용성에 의해 제한되지 않습니다. 둘 이상의 종속성 그룹을 포함하는 파도는 향후 반복에서 더 작은 파도로 분해될 수 있습니다.

지정된 웨이브에 대한 종속성 그룹이 확인되면 웨이브 마이그레이션을 위한 리소스 요구 사항을 검토합니다. 리소스 요구 사항에 따라 파도 크기(포함된 종속성 그룹 수)를 조정하는 것이 좋습니다. 이로 인해 파도가 작거나 커질 수 있습니다. 모든 웨이브가 정의될 때까지 필요에 따라 웨이브 플랜을 반복합니다.

## 변경 관리

마이그레이션 프로그램의 수명 주기 동안 애플리케이션 및 관련 인프라 포트폴리오가 변경됩니다. 장기 실행 마이그레이션 프로그램은 정상적인 비즈니스 진화 및 변화와 공존합니다. 애플리케이션은 마이그레이션을 기다리면서 계속 진화합니다. 서버가 추가 또는 제거되고 새 인프라가 온프레미스에 배포됩니다. 웨이브 또는 종속성 그룹의 범위는 변경해야 합니다. 특히 마이그레이션 날짜에 가까워지거나, 이전에 알려지지 않은 종속성이 식별되거나, 인벤토리에 새 서버가 포함된 경우 변경이 필요합니다. 마이그레이션 자체 중에는 문제가 발생할 수 있습니다.

범위 변경은 종속성 그룹 및 웨이브에 영향을 미칩니다. 변경을 처리하고 영향을 최소화하려면 범위 제어 메커니즘을 설정하는 것이 중요합니다. 범위 변경 제어 메커니즘을 사용하려면 범위에 대한 단일 정확한 소스를 정의해야 합니다. 마이그레이션 프로그램 거버넌스에서 정의한 범위 또는 .csv 파일, 스프레드시트 또는 데이터베이스를 관리하는 도구일 수 있습니다. 조치를 취할 수 있도록 변경 사항을 식별하고, 영향을 분석하고, 관련 이해관계자에게 변경 사항을 전달해야 합니다. 그 결과 웨이브 플랜이 반복됩니다.

## 세부 비즈니스 사례

이 단계에서는 변환 프로그램을 지원하기 위해 더 높은 수준의 세부 정보를 제공하기 위해 비즈니스 사례의 범위를 검증하고 확장하는 것이 좋습니다. 빠르게 조합된 초기 방향성 비즈니스 사례는 기본 단계와 다음 수준의 세부 계획에 투자할 수 있는 충분한 확신을 제공하도록 설계되었습니다.

세부 비즈니스 사례를 개발하면 다음과 같은 방법으로 이 계획 프로세스를 지원합니다.

- 마이그레이션 및 현대화해야 할 사항, 선택할 옵션, 작업의 단계화 및 우선 순위 지정 방법을 결정하는 데 도움이 되는 재무 분석 제공
- 세부적으로 재검토하여 원래 방향성 재무 사례를 검증, 개선 및 개발합니다.
  - 인프라 비용 절감 가능성
  - 내부 IT 생산성 및 아웃소싱된 운영 효율성
  - 프로그램 설정, 마이그레이션 및 현대화에 필요한 투자 추정치
- 마이그레이션이 가져오는 추가 가치 동인을 추적하기 위한 프로세스 식별, 규모 추정 및 설정

세부 비즈니스 사례에서는 다음을 설정합니다.

- 최소한 마이그레이션의 첫 단계를 구현하기 위한 권한 및 투자를 확보하기 위한 목표 기반
- 프로그램에 대한 기존 최소 재무 성과 기대치
- 다양한 마이그레이션 설계 및 우선순위 결정이 이루어지는 재정적 근거에 대한 명확성. 따라서 프로그램 과정에서 상황과 사람이 변경될 때 새로운 리더십이 정보에 입각한 선택을 할 수 있습니다.
- 워크로드 마이그레이션 및 작업 시작 시 초기 사용 데이터를 사용할 수 있게 된 후 탐색할 비용 최적화의 증분 영역에 대한 인사이트
- 복원력과 민첩성 향상으로 인해 클라우드 혁신이 비즈니스에 미치는 가치 추정
- 복원력 및 민첩성 개선으로 인한 재무 수익률을 추정하는 데 사용되는 관련 KPIs, 지표 및 가정은 프로그램에서 주요 이점 실현을 추진하기 위한 기준을 형성합니다.

## 사례에 필요한 시나리오 결정

세부 비즈니스 사례를 구축할 때는 일반적으로 비즈니스 사례가 사용되는 다양한 목적을 지원하기 위해 여러 시나리오를 개발해야 합니다.

최소 변경 시나리오 - 최소 재무 성과 기대치를 평가하려면 상태 할당량에 대한 최소 예상 변경을 가정하는 시나리오를 준비합니다. 최악의 시나리오인이 시나리오는 마이그레이션에 투자할 권한을 얻을

때 유용한 지원입니다. 이 시나리오는 가용성 및 복원력과 같은 다른 quality-of-service 요구 사항에 대한 최소 예상 용량 증가 수준과 최소 변경 사항을 모델링합니다. 최소 변경 사항은 현재 운영 모델에 대해 최저 비용과 최소 리소스 비효율성을 생성합니다.

가능성이 가장 높은 시나리오 - 프로그램 전략 및 우선순위 결정을 알려려면 비즈니스가 발생할 것으로 예상되는 상황을 반영하는 시나리오를 준비합니다. 이 시나리오에는 비즈니스의 높은 수준의 서비스 품질(특히 가용성 및 복원력)에 대한 수요를 충족하기 위해 가능한 최대 사용률 증가 또는 감소와 업그레이드 비용이 포함되어야 합니다.

기타 특정 시나리오 - 비즈니스 사례에 큰 영향을 미칠 수 있는 가정이 여전히 필요한 경우 가정이 사실인 경우와 그렇지 않은 경우 모두에 대한 시나리오를 개발합니다. 그러나 이러한 대체 시나리오의 수를 절대 최소값으로 유지하는 것이 좋습니다. 총 3~4개 이상의 시나리오를 생성하면 진행 속도가 느려지고 비용이 많이 들고 혼란스러우며 유지 관리가 어려워집니다. 가능하면 실험을 수행하고 더 큰 가정을 제거하기 위해 노력합니다.

## 인프라 및 마이그레이션 비용 모델 검증 및 구체화

포트폴리오 분석을 완료하고 대상의 설계 및 크기를 준비한 후 각 시나리오에 대한 현재 운영 모델(COM) 및 향후 운영 모델(FOM)에 대한 실행 비용 추정치 AWS 를 AWS 서비스구체화합니다. 일반적으로 다음에 대한 추정치를 구체화해야 합니다.

- 하이퍼바이저 호스트 서버, 베어 메탈 서버, 스토리지, 네트워크 디바이스, 보안 어플라이언스 하드웨어 새로 고침, 설치 및 유지 관리의 COM 인프라 비용. 시나리오에 필요한 용량에 대한 실제 요금 및 할인 수준으로 이를 계산합니다.
- 공간, 쿨링, 전력, 랙, 무정전 전원 공급 장치(UPS), 케이블, 물리적 보안 시스템을 포함한 COM 데이터 센터 및 공동 배치 시설 비용은 성장에 맞게 크기가 조정되고 용량을 충족하도록 지정되며 시나리오의 고가용성 및 재해 복구(DR) 수준을 충족합니다.
- WAN 링크, 콘텐츠 전송 네트워크 및 가상 프라이빗 네트워크(VPNs) 비용을 포함한 COM 네트워크 서비스 비용은 시나리오의 연결, 대역폭, 처리량 및 지연 시간 요구 사항에 대한 계약 요금을 사용하여 계산됩니다.
- 시나리오의 사용량 증가 또는 감소를 제공하기 위해 기존 계약에 따른 COM 애플리케이션 및 인프라 소프트웨어 비용입니다.
- 정교한 서비스 아키텍처, 인스턴스 크기, 선호 요금 모델, 예상 사용량 및 사용량 변동성에 따라 필요에 따라 기술 지원 및 관리형 서비스를 포함한 FOM AWS 유틸리티 비용.
- 최종 애플리케이션 설계, 애플리케이션을 실행하는 인프라 구성, 시간 경과에 따른 성장, 라이선스 이전성 규칙을 기반으로 하는 FOM 애플리케이션 라이선스입니다.

- FOM 마이그레이션 및 현대화 비용 추정치는 시나리오의 기존 마이그레이션 웨이브 계획을 반영하도록 개선되었으며 각 워크로드, 특히 리플랫폼, 리구매 또는 리팩터링할 워크로드에 대한 비용을 제공하도록 자세히 설명되었습니다.
- 자산 상각 및 계약 조기 종료 비용 추정치를 포함한 FOM 폐기 비용, 기존 마이그레이션 웨이브 계획의 폐기 시점을 반영하도록 수정, 상각을 최소화하기 위해 용도를 변경할 수 있는 자산과 전환할 수 있는 자산 확인, 물리적 자산 및 미디어의 폐기 비용.
- 마이그레이션 병렬 실행 비용은 각 마이그레이션 전환 시점과 각 기존 서비스 폐기 시점을 반영하도록 개선되었습니다.

## IT 생산성 및 IT 운영 개선 및 효율성 가치 모델 지원

방향성 비즈니스 사례와 마찬가지로 IT 운영 및 지원과 관련된 가치 모델을 구체화하고 개발하는 데는 두 가지 주요 접근 방식이 있습니다. 선택하는 접근 방식은 COM이 사내에서 관리되는지 아니면 계약 업체 또는 아웃소싱 서비스를 통해 관리되는지에 따라 달라집니다.

### 내부 팀 생산성 개선

IT 운영 및 지원이 사내에서 관리되는 경우 비즈니스 사례의 초점은 다음과 같습니다.

- 마이그레이션 및 범위에 포함된 운영 자동화로 인한 생산성 향상 식별 및 정량화
- 사내 팀의 여유 시간을 다른 일반적으로 가치가 높은 활동에 쉽고 생산적으로 적용할 수 있는지 검증하여 팀에 진행 기회를 제공하고 팀에 더 큰 보상을 제공하며 조직에 더 많은 가치를 제공합니다.

팀 내 각 역할의 각 구성원이 다양한 정기 활동에 얼마나 많은 시간을 할애하는지 평가하고 다양한 활동에 대한 예상 워크로드 감소에 대한 지침을 제공합니다.

다음 표는 대량의 IT 운영을 소비하고 팀의 다양한 역할에 걸쳐 노력을 지원하는 작업에 대한 활동별 일반적인 워크로드 감소 수준에 대한 초기 지침을 제공합니다. 이 표에는 생산성 달성 방법에 대한 설명이 포함되어 있습니다.

#### Note

나열된 활동은 일반적으로 여러 역할의 팀원이 수행하므로 팀의 전체 역할 집합에서 각 작업에 대한 생산성 절감을 평가해야 합니다. 예를 들어 인프라 타워(예: 컴퓨팅, 스토리지, 네트워킹) 별로 구성된 IT 운영 팀에서는 각 타워의 타워 리드에 자본 지출 계획 및 예산 책정이 일반적일 수 있습니다.

운영 및 지원 활동	절감 수준	생산성 드라이버
인프라 설계	중간	고려해야 할 파라미터가 적어 설계가 간소화됩니다.
자본 지출 계획 및 예산 책정	높음	OPEX 중심의 탄력적 서비스는 사실상 모든 예산 책정 및 계획 문제를 제거합니다.
구매	높음	가 설정되면 조달이 크게 간소화 AWS 계정 됩니다.
용량 계획	보통-매우 높음	네트워킹 및 컴퓨팅 용량 관리 워크로드는 일반적으로 제거되지만 스토리지의 경우 크게 간소화됩니다.
튜닝	매우 높음	인스턴스의 크기는 언제든지 변경할 수 있으므로 관리형 서비스에는 튜닝이 필요하지 않고 다른 서비스에는 거의 필요하지 않습니다.
하드웨어 장애 관리	매우 높음	클라우드에서 하드웨어를 처리하는 모든 측면은에서 투명하게 처리됩니다 AWS.
서버 가용성 및 통신 모니터링	높음	도구 지원 및 자동화를 통해 AWS 모니터링 및 통신이 광범위하게 간소화됩니다.
보안 관리	중간	AWS 보안 기능과 AWS 클라우드 하드웨어, 소프트웨어, 네트워킹 및 시설에 대한 <a href="#">보안 책임을</a> AWS 소유하면 워크로드가 크게 줄어듭니다.
네트워크 및 스토리지 업그레이드, 유지 관리 및 패치.	매우 높음	클라우드의 네트워크 및 스토리지 유지 관리의 모든 측면

운영 및 지원 활동	절감 수준	생산성 드라이버
		<p>은에서 투명하게 처리됩니다 AWS.</p>
<p>랙킹 및 스택킹 - 하드웨어 물류</p>	<p>매우 높음</p>	<p>클라우드에서 하드웨어를 관리하는 모든 측면은에서 투명하게 처리됩니다 AWS.</p>
<p>백업</p>	<p>중간</p>	<p>백업은 AWS 도구, 유연한 스토리지 시스템 및 자동화를 통해 광범위하게 간소화됩니다.</p>
<p>관리형 서비스(예: Amazon S3, Amazon RDS AWS Lambda 및 AWS Fargate)</p>	<p>매우 높음</p>	<p>관리형 서비스는에서 완벽하게 관리하는 환경에서 실행되므로 유지 관리, 패치 AWS 적용, 모니터링 또는 프로비저닝 관리 활동이 필요하지 않습니다.</p>
<p>디바이스 및 서비스 설정 및 커미셔닝</p>	<p>매우 높음</p>	<p>VPNs을 설정하거나 AWS 데이터 센터에 연결 AWS Direct Connect 하기 위한 WAN 연결 디바이스를 제외하고 로 마이그레이션된 자산의 하드웨어 설정 활동은 AWS 일반적으로 줄어듭니다.</p>
<p>엔드포인트 보호 및 바이러스 백신 보호</p>	<p>높음</p>	<p>엔드포인트 보호 및 바이러스 백신 서비스의 애플리케이션 및 유지 관리는 일반적으로 마이그레이션 설계의 일부로 광범위하게 자동화됩니다.</p>
<p>위험, 취약성 및 위험 평가</p>	<p>높음</p>	<p>AWS 는 핵심 플랫폼에 초점을 맞춘이 요소의 지원을 제공하며,가 아키텍처를 보호하기 위해 AWS 제공하는 메커니즘은 평가를 간소화합니다.</p>

운영 및 지원 활동	절감 수준	생산성 드라이버
데이터 센터 인프라 프로젝트 관리	높음	인프라 서비스의 확장, 새로 고침 또는 폐기를 위한 설치 작업의 프로젝트 관리. 인프라 소프트웨어 및 서비스의 일부 관리는 남아 있지만 이는 온프레미스 인프라보다 훨씬 간단하며 하드웨어 활동이 제거됩니다.
데이터 센터 시설 관리	보통-매우 높음	모든 서버, 스토리지 디바이스, 보안 어플라이언스 및 관련 랙에서 발생하는 시설 관리 작업은 마이그레이션되는 모든 항목에 대해 제거됩니다. 그러나 일부 작업은 일반적으로 WAN 링크 네트워크 디바이스 및 하이브리드 아키텍처의 온프레미스에 보관되는 모든 인프라를 위한 시설을 제공하기 위해 남아 있습니다.
애플리케이션 아키텍처, 개발, 관리 및 테스트	낮음	애자일 개발 도구 체인을 필요에 따라 테스트 환경을 구축하기 위한 애플리케이션 스택 인스턴스화 및 폐기 자동화와 함께 사용하면 애플리케이션 개발 리드 타임을 줄이고 많은 수동 테스트 단계를 제거할 수 있습니다.

운영 및 지원 활동	절감 수준	생산성 드라이버
애플리케이션 소프트웨어 설치 및 구성	중간	전체 애플리케이션 스택 설치 및 구성은 AWS CloudFormation 와 같은 서비스를 사용하여 쉽게 자동화되고를 사용하여 쉽게 구성할 수 있는 랜딩 존을 사용하여 간소화됩니다 AWS Control Tower.
IT 지원	중간	셀프 서비스 프로비저닝을 위한 Service Catalog 기능 사용, 저렴한 고가용성 아키텍처 사용 증가(중단 감소, 자동 조정 및 엣지 컴퓨팅 구성)를 통해 용량 및 성능 문제를 줄임으로써 L1 및 L2 지원을 줄일 수 있습니다.
데이터베이스 관리	최소-낮음	이러한 활동은 대부분 변경되지 않습니다. 일반적으로 AWS 온프레미스 인프라와 동일한 수준의 리소스가 제공됩니다.
인프라 및 보안 요구 사항 캡처, 분석 및 설계	최소화	
설명서	최소화	
애플리케이션 및 성능 모니터링	최소화	
L3 기술 지원, 쿼리 응답, 문제 해결 및 문제 해결	최소화	
애플리케이션 소프트웨어 설치 및 구성	최소화	
애플리케이션 L3 지원(예산 책정 및 장기 용량 계획 제외)	최소화	

다음 표에는 워크로드 감소 수준별 예상 절감액이 나와 있습니다.

수준	예상
매우 높음	85% - 100%
높음	60% - 90%
중간	30% - 70%
낮음	10% - 35%
최소화	0% - 10%

이러한 지표는 생산성 향상을 평가하고 이를 세부 비즈니스 사례에 포함하기 위한 출발점을 제공합니다. 실제 생산성 향상은 특정 상황에 따라 달라집니다. 범위의 중간점과 하단 모두에서 생산성 절감을 계산하여 일반적인 시나리오와 보수적인 시나리오를 추정하는 것이 유용할 수 있습니다.

프로그램이 진행됨에 따라 역할별로 각 활동에 소요된 시간에 대한 실제 데이터를 캡처하는 것이 중요합니다. 이 데이터는 운영을 추정하기 위한 개선된 기반을 구축하고 새로운 프로젝트 및 서비스 확장에 대한 비용을 지원합니다.

### 아웃소싱된 IT 운영 및 비용 절감 지원

IT 운영 및 지원이 주로 계약업체와 아웃소싱되거나 관리되는 경우 파트너 AWS 주도(AMS)를 포함한 관리형 서비스 솔루션을 제공하는 AWS 파트너에게 견적을 요청하여 향후 운영 모델(FOM)에 대한 비용 할당을 준비할 수 있습니다. [AWS Managed Services 또한 방향성 비즈니스 사례 생성](#) 섹션의 [운영 비용 최적화 구축에](#) 대한 하위 섹션에 설명된 대로 AWS 계정 관리자에게 문의하여 AMS 가격을 직접 요청할 수 있습니다.

자세한 비즈니스 사례의 경우 벤치마크 수치를 수정된 AWS 서비스 자재 명세서 및 예상 서비스 소비, AMS 패키지 및 필요한 옵션, 필요한 서비스 수준에 따른 인용 부호로 바꿉니다. 비용에는 일회성 구현 구성 요소와 소비 기반 실행 속도가 포함됩니다.

나머지 IT 작업, 마이그레이션되지 않을 서비스에 대해 유지되어야 하는 지원 AWS, 계약 벌금(예: 조기 종료)이 있는 경우 일회성 비용을 포함합니다.

## 복원력 가치 모델 개발

에서 광범위한 고 AWS가용성, 재해 복구 및 내결함성 아키텍처를 구성할 수 있습니다. 소비 기반 요금은 서비스 사용 시에만 요금이 청구됨을 의미합니다. 이 두 가지 요소를 함께 사용하면 복원력을 위해 뛰어난 비용 성능을 얻을 수 있습니다.

또한 AWS 고객은 이를 사용하여 워크로드의 복원력을 개선했습니다. [IDC 2018 설문](#) 조사는 참여 고객이 연간 73% 더 적은 중단, 평균 복구 시간(MTTR) 58% 감소, 생산성 손실 94% 감소를 달성하는 예를 제공합니다. 동일한 설문 조사에 따르면 복원력 향상을 통해 얻은 재정적 이점이 IT 인프라 비용 절감 이점보다 50% 더 큰 것으로 나타났습니다.

또한 애플리케이션의 소프트웨어 개발 수명 주기를 현대화하여 추가 복원력을 확보할 수 있습니다. 테스트 자동화가 포함된 CI/CD 파이프라인이 비즈니스 민첩성 향상을 지원하기 위해 도입되는 경우 개발 주기 초기에 소프트웨어 결함이 발견되어 소프트웨어 유지 관리 비용이 크게 절감됩니다.

비즈니스 사례에이 값을 평가하고 포함하려면 먼저 애플리케이션 비즈니스 소유자와 협력하여 마이그레이션할 각 워크로드의 총 이점 기회를 파악합니다. 여기에는 다음 항목이 포함될 수 있습니다.

- 서비스 중단 횟수, 평균 기간 및 특성:
  - 서비스 중단의 예로는 중단, 성능 저하, 계획된 배치 및 유지 관리 기간 오버런, 주요 함수의 버그, 피크 기간 동안의 액세스 제한 등이 있습니다.
- 전자 상거래 시스템과 같은 수익 창출 서비스 중단으로 인한 수익에 미치는 영향:
  - 중단 시간 및 트랜잭션 비율에 따라 서비스 중단을 통해 완료할 수 없는 트랜잭션 수
  - 영향을 받는 각 트랜잭션의 평균 값
- 지원 엔지니어가 프로덕션 시스템의 결함을 해결하는 데 드는 추가 비용과 개발 프로세스 초기에 결함을 발견하는 데 드는 비용 비교
- 내부 사용자의 생산성 및 시간 손실 비용에 미치는 영향

그런 다음 서비스 중단으로 인한 예상 시간과 더 보수적인 시간 단축을 평가하여 복원력을 높일 수 있습니다. 예를 들어 다음 항목을 포함하는 것이 좋습니다.

- 고가용성 아키텍처를 사용하여 중단 및 MTTR 수를 줄이고 복구 시간 목표(RTO) 및 복구 시점 목표(RPO)를 개선했습니다.
- 자동 조정과 같은 기능을 사용하여 속도 저하 감소, 용량 제한 제거 및 배치 처리 오버런 방지
- CI/CD 파이프라인 구현과 비용 최소화를 위해 스핀업 및 스핀다운 인프라에 대한 자동 회귀 테스트를 통해 프로덕션에서만 검색되는 애플리케이션 버그 수 감소

마이그레이션 및 현대화할 애플리케이션 포트폴리오에 대해 이를 결합하고 사례의 각 연도에 대해 예상되고 보수적인 비즈니스 가치 수치를 계산합니다. 이점은 마이그레이션 일정에 따라 증가한 다음 기여 애플리케이션의 사용량 증가 기대치에 따라 볼륨을 확장해야 합니다.

## 비즈니스 민첩성 가치 모델 개발

비즈니스 민첩성은 AWS 고객이 마이그레이션하는 주요 이유입니다 AWS. [IDC 2018 고객 설문 조사](#)에 따르면 비즈니스 민첩성 이점은 측정된 총 이점의 47%를 차지했으며 인프라 비용 절감으로 인한 이점의 5배 이상을 차지했습니다. AWS

모든 혁신에서 얻을 수 있는 모든 비즈니스 민첩성 이점을 정확하게 예측하는 것은 어렵습니다. 그러나 많은 수의 사용자를 지원하거나 비즈니스 차별화의 소스인 애플리케이션에 집중하면 이러한 이점의 중요한 부분을 모델링하여 기본 세부 비즈니스 사례에 포함할 수 있습니다.

마이그레이션이 진행됨에 따라 더 많은 이점이 정량화 가능해짐에 따라 비즈니스 민첩성 가치 모델을 점진적으로 구체화하고 확장합니다. 이렇게 하면 비즈니스 사례가 관련성을 유지하므로 프로그램을 주도하는 데 사용할 수 있는 기본 의사 결정 지원 도구로 사용할 수 있습니다.

비즈니스 민첩성 가치 모델을 구축하려면 다음 지침을 사용합니다.

- 다음과 같이 최고의 비즈니스 성과 개선을 추진할 기회가 있는 워크로드를 선택합니다.
  - 수익 생성 워크로드
  - 효율성을 높이고 비즈니스에서 비용을 절감할 수 있는 범위가 있는 비즈니스 운영 워크로드
  - 대규모 사용자 기반을 지원하는 비즈니스 생산성 도구
- 수익 및 효율성 생성 워크로드의 경우 다음을 수행합니다.
  - 메이저 및 마이너 애플리케이션 업그레이드가 주도할 것으로 예상되는 수익 증가 또는 운영 효율성을 현실적이고 보수적으로 평가합니다.
  - AWS 애플리케이션 개발 속도를 높이고 인프라 배포 시간을 단축할 수 있는 연간 메이저 및 마이너 릴리스 수 증가를 추정합니다. 이에 대한 일부 기준 지표는 IDC 보고서에 제공됩니다.
  - 현실적이고 보수적인 혜택 기대치를 계산합니다. 비즈니스 사례 기간 동안 매핑하여 각 워크로드가 마이그레이션된 후 일정 시간 동안 효율성을 최대한 높일 수 있습니다.
- 비즈니스 생산성 도구의 경우 다음을 수행합니다.
  - 메이저 및 마이너 애플리케이션 업그레이드가 주도할 것으로 예상되는 시간 절감을 현실적이고 보수적으로 평가합니다.
  - 영향을 받는 사용자 기반에서 인력의 평균 시간과 노력을 추정합니다.

- 수치를 사용하여 메이저 및 마이너 릴리스 빈도를 늘리고 비즈니스 사례 기간 동안의 이점을 계산합니다.

개발자 생산성이 향상되고 시작 시간이 단축되면 추가 리소스가 필요하지 않으므로 할인된 현금 흐름, NPV, ROI, MIRR 및 페이백 계산에 포함할 수 있도록 각 워크로드의 순 혜택 라인을 비즈니스 사례 현금 흐름 모델에 추가합니다.

## 지속적인 평가 및 개선

이 평가 단계는 다음 두 가지 측면에 중점을 둡니다.

- 각 애플리케이션 웨이브에 대한 지속적인 세부 애플리케이션 평가
- 포트폴리오의 지속적인 발전 및 개선

첫 번째 측면인 지속적인 세부 애플리케이션 평가는 아키텍처 및 기술 수준까지 세부 검색 및 분석에 중점을 두고 주어진 웨이브의 각 애플리케이션, 제안된 AWS 설계 및 마이그레이션 전략을 완전히 이해합니다. 이 마이그레이션 준비 상태 평가는 지정된 마이그레이션 웨이브를 시작하기 위한 사전 조건입니다.

두 번째 측면인 포트폴리오의 지속적인 진화 및 개선은 포트폴리오 관리와 비즈니스 사례의 진화 및 추적을 포함하여 시간 경과에 따른 애플리케이션 개선 계획에 중점을 둡니다.

이 단계의 기본 마이그레이션 결과는 다음과 같습니다.

- 각 웨이브에 대해 검증된 마이그레이션 범위
- 지정된 마이그레이션 웨이브의 애플리케이션에 대한 문서화된 대상 아키텍처 및 마이그레이션 전략
- 식별 및 검증된 마이그레이션 패턴 및 도구
- 각 웨이브에 대해 문서화된 요구 사항(보안, AWS 인프라 및 운영) 및 마이그레이션 전환 고려 사항

이 단계의 기본 최적화 결과는 다음과 같습니다.

- 포트폴리오 합리화 모델 및 비즈니스 성과
- 제안된 아키텍처 및 기술 변경 사항과 예상되는 이점
- 플랫폼 요구 사항(보안, AWS 인프라 및 운영)
- 구현 계획

## 지속적 평가 데이터 요구 사항 이해

애플리케이션 포트폴리오의 지속적인 평가 및 개선을 위한 데이터 요구 사항은 이전 섹션의 데이터 요구 사항의 조합입니다. 포트폴리오 마이그레이션 및 진화를 지속적으로 관리하려면 다음 섹션을 참조하여 데이터 요구 사항을 이해합니다.

- 파도 평가 및 애플리케이션 최적화를 위해 [우선 순위가 지정된 애플리케이션 평가](#) 섹션의 데이터 요구 사항을 사용합니다.
- 지속적인 포트폴리오 관리를 위해 [포트폴리오 분석 및 마이그레이션 계획](#) 섹션의 데이터 요구 사항을 사용합니다.
- 웨이브 계획을 정의하려면 [웨이브 계획](#) 섹션을 참조하세요.

## 세부 파도 평가

마이그레이션 웨이브에 앞서 마이그레이션을 위한 주요 활성화 도구로서 애플리케이션에 대한 자세한 평가는 [우선 순위가 지정된 애플리케이션 평가](#) 단계와 동일한 요구 사항 및 권장 사항을 갖습니다. 목표는 특정 웨이브에서 애플리케이션의 현재 상태를 자세히 이해하고 운영 측면, 도구 및 특정 마이그레이션 패턴을 포함한 미래 상태 아키텍처 설계 및 마이그레이션 전략을 생성하는 것입니다.

지정된 웨이브의 [애플리케이션 그룹에 우선 순위가 지정된 애플리케이션 평가를](#) 적용합니다. 마이그레이션 계획의 각 웨이브에 앞서이 프로세스를 반복합니다. 핵심은 세부 평가와 웨이브 시작 사이에 충분한 시간을 예약하는 것입니다. 필요한 시간은 웨이브 요구 사항을 구현하고 마이그레이션을 수행하는 플랫폼 및 마이그레이션 팀의 요구 사항에 따라 결정됩니다. 이러한 팀과 협력하여 자세한 파도 평가와 파도를 예약합니다. 프로덕션 라인을 에뮬레이션하는 공장과 유사한 모델을 구현하는 것이 좋습니다.

## 최적화 및 현대화 평가

이미 로 마이그레이션된 워크로드 최적화 및 현대화를 위한 평가 프로세스는 마이그레이션할 워크로드 평가와 AWS 유사합니다 AWS. 주로 평가를 수행하기 위한 데이터 소스가 변경됩니다. 에는에서 실행되는 애플리케이션에 대한 자세한 정보를 얻는 데 사용할 수 있는 몇 AWS가지 out-of-the-box 도구 및 서비스가 있습니다 AWS.

애플리케이션을 최적화하고 현대화하는 대상과 방법은 고유한 동인과 상황을 기반으로 합니다. 최적화는 현재 아키텍처 및 기술에 변경 사항을 적용하여 비용을 절감하고 성능 요구 사항을 조정하며 학습한 교훈을 통합하는 데 중점을 둡니다. 현대화는 서버리스 모델 및 마이크로서비스 아키텍처 채택과 같은 애플리케이션을 한 차원 높이는 데 중점을 둡니다.

[우선 순위가 지정된 애플리케이션 평가](#)의 지침을 따릅니다. 최적화 및 현대화 작업을 추가로 지원하려면 다음 리소스를 참조하세요.

- [AWS 비용 최적화](#)는 IT 최적화 및 IT 비용 절감에 대한 정보를 제공합니다.

- [AWS Compute Optimizer](#)는 기계 학습을 사용하여 과거 사용률 지표를 분석하여 비용을 절감하고 성능을 개선하기 위해 워크로드에 AWS 리소스를 권장합니다.
- [AWS 비용 및 용량 최적화 서비스 및 도구](#)는 컴퓨팅 리소스를 관리하는 데 도움이 되므로 컴퓨팅 비용을 구축하는 데 더 많은 시간을 소비하고 관리하는 데 소요되는 시간을 줄일 수 있습니다.
- [Amazon S3 Storage Lens](#)는 객체 스토리지 사용량 및 활동 추세에 대한 조직 전체의 가시성을 제공합니다. 비용 효율성을 개선하고 데이터 보호 모범 사례를 적용하기 위해 실행 가능한 권장 사항을 제시합니다.
- [Database Freedom](#)을 사용하면 AWS 데이터베이스 및 분석 서비스로 쉽게 마이그레이션할 수 있습니다.
- [Amazon CodeGuru](#)는 코드 품질을 개선하고 애플리케이션의 가장 비용이 많이 드는 코드 라인을 식별하기 위한 지능형 권장 사항을 제공하는 개발자 도구입니다.
- [AWS 하이브리드 클라우드 서비스](#)는 클라우드, 온프레미스, 엣지 등 필요한 모든 곳에서 일관된 AWS 경험을 제공합니다.

## 추가 리소스

- [비용 최적화 및 혁신: 애플리케이션 현대화 소개](#)(블로그 게시물)
- [서버리스 웹 애플리케이션의 비용 최적화](#)(블로그 게시물)
- [Windows on AWS](#)(블로그)
- [최신 애플리케이션](#)
- [애플리케이션 현대화](#)(AWS re:Invent 2020)
- [AWS 마이크로서비스 가이드](#)

## 웨이브 플랜 반복

마이그레이션 프로그램이 발전하고 더 많은 웨이브가 마이그레이션됨에 따라 학습한 교훈과 변화하는 비즈니스 우선 순위가 따라 마이그레이션 웨이브 플랜을 발전시키는 것이 중요합니다. 특히 장기 실행 마이그레이션 프로그램의 경우 비즈니스 동인과 조직 변화를 재평가하고 마이그레이션 웨이브 플랜이 여전히 유효한지 확인하는 것이 중요합니다.

마찬가지로 마이그레이션에서 얻은 교훈은 웨이브 플랜 구성과 각 웨이브의 범위에 영향을 미칩니다. 진행 중인 상황에 대한 가시성 손실을 방지하려면 [웨이브 플랜](#)을 최신 상태로 유지하세요. 계획은 전달되는 내용을 반영하고 추적해야 하며 마이그레이션 범위에 대한 변경을 관리하고 평가해야 합니다.

## 비즈니스 사례 개선 및 추적

마이그레이션이 진행됨에 따라 특히 장기 실행 프로그램의 경우 비즈니스 부담으로 인해 마이그레이션 및 현대화 우선 순위가 정기적으로 재검토될 수 있습니다.

새로운 정보가 제공되면 둘 다 비즈니스 사례를 발전시키고 세부 비즈니스 사례에 문서화된 기대치와 비교하여 실제 상용 성과를 추적하는 것이 좋습니다. 이러한 권장 사항은 다음과 같습니다.

- 비즈니스 우선순위에 영향을 미치고 이를 통해 IT 전략 및 애플리케이션 포트폴리오에 영향을 미치는 조직의 새로운 구조적 변화
- 애플리케이션 포트폴리오의 한 부분 또는 마이그레이션 및 현대화가 목표로 하는 변경 사항의 상업적 중요성 증가
- 점진적 현대화 사례의 크기 조정, 정량화 및 확인을 포함하여 마이그레이션된 애플리케이션에 대한 실제 리소스 사용률 데이터의 가용성
- IT 운영 및 지원 활동에 사용된 노력에 대한 데이터의 가용성, 가능한 운영 개선 및 자동화 분석
- 소프트웨어 개발 및 유지 관리 주기 시간의 변경 사항을 측정하는 데이터의 가용성, 개발 단계 및 서비스 가용성 정보별 소프트웨어 결함, 추가 개선이 필요한 영역에 대한 근본 원인 분석

비즈니스 사례를 기준으로 성능을 추적하면 마이그레이션이 시작된 후 보다 쉽게 평가하고 정량화할 수 있는 추가 개선 사항을 포함하도록 사례를 발전시킬 수 있습니다. 프로그램 거버넌스 조직은 변화하는 비즈니스 압력에 대응하고 관리 가능하고 허용 가능한 위험 수준에서 가장 큰 가치를 창출하는 방향으로 혁신을 주도할 수 있는 역량이 훨씬 더 뛰어납니다.

이는 사례 내의 IT 생산성, 복원력 및 비즈니스 민첩성 이점에 특히 중요합니다. 이는 일반적으로 미리 평가하기가 더 크고 더 어려운 동인입니다. 팀은 이러한 동인의 성능을 추적하여 심층적으로 분석하고 이점 실현을 방해하는 문제를 해결할 수 있습니다. 또는 비즈니스 사례를 조정하여 가장 지속적인 재무 성과 최적화를 달성하는 이니셔티브의 우선순위를 지정할 수 있습니다.

# 리소스

## AWS 참조

- [The Amazon Builders' Library](#)
- [애플리케이션 현대화\(AWS re:Invent 2020\)](#)
- [애플리케이션 포트폴리오 평가 전략](#)
- [AWS 아키텍처 센터](#)
- [AWS Compute Optimizer](#)
- [AWS 비용 및 용량 최적화 서비스 및 도구](#)
- [AWS 비용 최적화](#)
- [비용 최적화 및 혁신: 애플리케이션 현대화 소개\(블로그 게시물\)](#)
- [검색, 계획 및 추천 마이그레이션 도구](#)
- [AWS Documentation](#)
- [리소스 센터 시작하기](#)
- [AWS Marketplace](#)
- [AWS Managed Services 파트너](#)
- [AWS 마이크로서비스 가이드](#)
- [AWS 마이그레이션 역량 파트너](#)
- [최신 애플리케이션](#)
- [서버리스 웹 애플리케이션의 비용 최적화\(블로그 게시물\)](#)
- [AWS 권장 가이드](#)
- [AWS 전문 서비스](#)
- [AWS 솔루션 라이브러리](#)
- [Windows on AWS \(블로그\)](#)

## AWS 서비스

- [AWS App2Container](#)
- [AWS Application Migration Service](#)
- [Amazon CodeGuru](#)

- [AWS Control Tower](#)
- [데이터베이스 자유](#)
- [AWS Database Migration Service](#)
- [AWS DataSync](#)
- [AWS Direct Connect](#)
- [Amazon ECS](#)
- [Amazon EKS](#)
- [AWS Fargate](#)
- [AWS Managed Services](#)
- [마이그레이션 평가자](#)
- [AWS Migration Hub 전략 권장 사항](#)
- [AWS 랜딩 존](#)
- [AWS Pricing Calculator](#)
- [AWS Schema Conversion Tool](#)
- [Amazon S3 Storage Lens](#)
- [AWS Snowball Edge](#)
- [AWS Snowball Edge](#)
- [Site-to-Site VPN](#)

#### 기타 리소스

- [Amazon Web Services를 통한 비즈니스 가치 창출을 위한 비즈니스 및 조직 변환 촉진](#)
- [IDC 2018 설문 조사](#)

## 문서 이력

다음 표에서는 이 전략에 대한 중요한 변경 사항을 설명합니다. 향후 업데이트에 대한 알림을 받으려면 [RSS 피드](#)를 구독하십시오.

변경 사항	설명	날짜
<a href="#">업데이트</a>	포트폴리오 검색 및 초기 계획 섹션 검색 가속화 및 초기 계획의 이름을 변경하고 의사 결정 트리 다이어그램을 업데이트했습니다.	2024년 5월 20일
<a href="#">=</a>	최초 게시	2021년 11월 12일

# AWS 권장 가이드 용어집

다음은 AWS 권장 가이드에서 제공하는 전략, 가이드 및 패턴에서 일반적으로 사용되는 용어입니다. 용어집 항목을 제안하려면 용어집 끝에 있는 피드백 제공 링크를 사용하십시오.

## 숫자

### 7가지 전략

애플리케이션을 클라우드로 이전하기 위한 7가지 일반적인 마이그레이션 전략 이러한 전략은 Gartner가 2011년에 파악한 5가지 전략을 기반으로 하며 다음으로 구성됩니다.

- 리팩터링/리아키텍트 - 클라우드 네이티브 기능을 최대한 활용하여 애플리케이션을 이동하고 해당 아키텍처를 수정함으로써 민첩성, 성능 및 확장성을 개선합니다. 여기에는 일반적으로 운영 체제와 데이터베이스 이식이 포함됩니다. 예: 온프레미스 Oracle 데이터베이스를 Amazon Aurora PostgreSQL 호환 에디션으로 마이그레이션합니다.
- 리플랫폼(리프트 앤드 리세이프) - 애플리케이션을 클라우드로 이동하고 일정 수준의 최적화를 도입하여 클라우드 기능을 활용합니다. 예: 온프레미스 Oracle 데이터베이스를 AWS 클라우드의 Amazon Relational Database Service(Amazon RDS) for Oracle로 마이그레이션합니다.
- 재구매(드롭 앤드 슝) - 일반적으로 기존 라이선스에서 SaaS 모델로 전환하여 다른 제품으로 전환합니다. 예: 고객 관계 관리(CRM) 시스템을 Salesforce.com으로 마이그레이션합니다.
- 리호스팅(리프트 앤드 시프트) - 애플리케이션을 변경하지 않고 클라우드로 이동하여 클라우드 기능을 활용합니다. 예: 온프레미스 Oracle 데이터베이스를 AWS 클라우드클라우드의 EC2 인스턴스에 있는 Oracle로 마이그레이션합니다.
- 재배포(하이퍼바이저 수준의 리프트 앤 시프트) - 새 하드웨어를 구매하거나, 애플리케이션을 다시 작성하거나, 기존 운영을 수정하지 않고도 인프라를 클라우드로 이동합니다. 온프레미스 플랫폼에서 동일한 플랫폼의 클라우드 서비스로 서버를 마이그레이션합니다. 예: Microsoft Hyper-V 애플리케이션을 로 마이그레이션합니다 AWS.
- 유지(보관) - 소스 환경에 애플리케이션을 유지합니다. 대규모 리팩터링이 필요하고 해당 작업을 나중에 연기하려는 애플리케이션과 비즈니스 차원에서 마이그레이션할 이유가 없어 유지하려는 레거시 애플리케이션이 여기에 포함될 수 있습니다.
- 사용 중지 - 소스 환경에서 더 이상 필요하지 않은 애플리케이션을 폐기하거나 제거합니다.

# A

## ABAC

[속성 기반 액세스 제어](#)를 참조하세요.

### 추상화된 서비스

[관리형 서비스](#)를 참조하세요.

## ACID

[원자성, 일관성, 격리성, 내구성](#)을 참조하세요.

### 능동-능동 마이그레이션

양방향 복제 도구 또는 이중 쓰기 작업을 사용하여 소스 데이터베이스와 대상 데이터베이스가 동기화된 상태로 유지되고, 두 데이터베이스 모두 마이그레이션 중 연결 애플리케이션의 트랜잭션을 처리하는 데이터베이스 마이그레이션 방법입니다. 이 방법은 일회성 전환이 필요한 대신 소규모의 제어된 배치로 마이그레이션을 지원합니다. 더 유연하지만 [액티브 패시브 마이그레이션](#)보다 더 많은 작업이 필요합니다.

### 능동-수동 마이그레이션

소스 데이터베이스와 대상 데이터베이스가 동기화된 상태로 유지되지만 소스 데이터베이스만 연결 애플리케이션의 트랜잭션을 처리하고 데이터는 대상 데이터베이스로 복제되는 데이터베이스 마이그레이션 방법입니다. 대상 데이터베이스는 마이그레이션 중 어떤 트랜잭션도 허용하지 않습니다.

### 집계 함수

행 그룹에서 작동하고 그룹에 대한 단일 반환 값을 계산하는 SQL 함수입니다. 집계 함수의 예로 SUM 및 MAX가 있습니다.

## AI

[인공 지능](#)을 참조하세요.

### AIOps

[인공 지능 운영](#)을 참조하세요.

### 익명화

데이터세트에서 개인 정보를 영구적으로 삭제하는 프로세스입니다. 익명화는 개인 정보 보호에 도움이 될 수 있습니다. 익명화된 데이터는 더 이상 개인 데이터로 간주되지 않습니다.

## 안티 패턴

솔루션이 다른 솔루션보다 비생산적이거나 비효율적이거나 덜 효과적이어서 반복되는 문제에 자주 사용되는 솔루션입니다.

### 애플리케이션 제어

맬웨어로부터 시스템을 보호하기 위해 승인된 애플리케이션만 사용하도록 허용하는 보안 접근 방식입니다.

### 애플리케이션 포트폴리오

애플리케이션 구축 및 유지 관리 비용과 애플리케이션의 비즈니스 가치를 비롯하여 조직에서 사용하는 각 애플리케이션에 대한 세부 정보 모음입니다. 이 정보는 [포트폴리오 탐색 및 분석 프로세스](#)의 핵심이며 마이그레이션, 현대화 및 최적화할 애플리케이션을 식별하고 우선순위를 정하는 데 도움이 됩니다.

### 인공 지능

컴퓨터 기술을 사용하여 학습, 문제 해결, 패턴 인식 등 일반적으로 인간과 관련된 인지 기능을 수행하는 것을 전문으로 하는 컴퓨터 과학 분야입니다. 자세한 내용은 [What is Artificial Intelligence?](#)를 참조하십시오.

### 인공 지능 운영(AIOps)

기계 학습 기법을 사용하여 운영 문제를 해결하고, 운영 인시던트 및 사용자 개입을 줄이고, 서비스 품질을 높이는 프로세스입니다. AWS 마이그레이션 전략에서 AIOps가 사용되는 방법에 대한 자세한 내용은 [운영 통합 가이드](#)를 참조하십시오.

### 비대칭 암호화

한 쌍의 키, 즉 암호화를 위한 퍼블릭 키와 복호화를 위한 프라이빗 키를 사용하는 암호화 알고리즘입니다. 퍼블릭 키는 복호화에 사용되지 않으므로 공유할 수 있지만 프라이빗 키에 대한 액세스는 엄격히 제한되어야 합니다.

### 원자성, 일관성, 격리성, 내구성(ACID)

오류, 정전 또는 기타 문제가 발생한 경우에도 데이터베이스의 데이터 유효성과 운영 신뢰성을 보장하는 소프트웨어 속성 세트입니다.

### ABAC(속성 기반 액세스 제어)

부서, 직무, 팀 이름 등의 사용자 속성을 기반으로 세분화된 권한을 생성하는 방식입니다. 자세한 내용은 AWS Identity and Access Management (IAM) 설명서의 [용 ABAC AWS](#)를 참조하세요.

## 신뢰할 수 있는 데이터 소스

가장 신뢰할 수 있는 정보 소스로 간주되는 기본 버전의 데이터를 저장하는 위치입니다. 익명화, 편집 또는 가명화와 같은 데이터 처리 또는 수정의 목적으로 신뢰할 수 있는 데이터 소스의 데이터를 다른 위치로 복사할 수 있습니다.

### 가용 영역

다른 가용 영역의 장애로부터 격리 AWS 리전 되고 동일한 리전의 다른 가용 영역에 저렴하고 지연 시간이 짧은 네트워크 연결을 제공하는 내의 고유한 위치입니다.

### AWS 클라우드 채택 프레임워크(AWS CAF)

조직이 클라우드로 성공적으로 전환 AWS 하기 위한 효율적이고 효과적인 계획을 개발하는 데 도움이 되는 지침 및 모범 사례 프레임워크입니다. AWS CAF는 지침을 비즈니스, 사람, 거버넌스, 플랫폼, 보안 및 운영이라는 6가지 중점 영역으로 구성합니다. 비즈니스, 사람 및 거버넌스 관점은 비즈니스 기술과 프로세스에 초점을 맞추고, 플랫폼, 보안 및 운영 관점은 전문 기술과 프로세스에 중점을 둡니다. 예를 들어, 사람 관점은 인사(HR), 직원 배치 기능 및 인력 관리를 담당하는 이해관계자를 대상으로 합니다. 이러한 관점에서 AWS CAF는 성공적인 클라우드 채택을 위해 조직을 준비하는 데 도움이 되는 인력 개발, 교육 및 커뮤니케이션에 대한 지침을 제공합니다. 자세한 내용은 [AWS CAF 웹사이트](#)와 [AWS CAF 백서](#)를 참조하세요.

### AWS 워크로드 검증 프레임워크(AWS WQF)

데이터베이스 마이그레이션 워크로드를 평가하고, 마이그레이션 전략을 권장하고, 작업 견적을 제공하는 도구입니다. AWS WQF는 AWS Schema Conversion Tool (AWS SCT)에 포함되어 있습니다. 데이터베이스 스키마 및 코드 객체, 애플리케이션 코드, 종속성 및 성능 특성을 분석하고 평가 보고서를 제공합니다.

## B

### 악성 봇

개인 또는 조직을 방해하거나 해를 입히기 위한 [봇](#)입니다.

### BCP

[비즈니스 연속성 계획](#)을 참조하세요.

## 동작 그래프

리소스 동작과 시간 경과에 따른 상호 작용에 대한 통합된 대화형 뷰입니다. Amazon Detective에서 동작 그래프를 사용하여 실패한 로그온 시도, 의심스러운 API 직접 호출 및 유사한 작업을 검사할 수 있습니다. 자세한 내용은 Detective 설명서의 [Data in a behavior graph](#)를 참조하십시오.

## 빅 엔디안 시스템

가장 중요한 바이트를 먼저 저장하는 시스템입니다. [엔디안](#)도 참조하세요.

## 바이너리 분류

바이너리 결과(가능한 두 클래스 중 하나)를 예측하는 프로세스입니다. 예를 들어, ML 모델이 “이 이메일이 스팸인가요, 스팸이 아닌가요?”, ‘이 제품은 책임가요, 자동차인가요?’ 등의 문제를 예측해야 할 수 있습니다.

## 블룸 필터

요소가 세트의 멤버인지 여부를 테스트하는 데 사용되는 메모리 효율성이 높은 확률론적 데이터 구조입니다.

## 블루/그린(Blue/Green) 배포

동일하지만 별개의 두 환경을 생성하는 배포 전략입니다. 하나의 환경(파란색)에서 현재 애플리케이션 버전을 실행하고 새 애플리케이션 버전은 다른 환경(녹색)에서 실행합니다. 이 전략을 사용하면 영향을 최소화하면서 신속하게 롤백할 수 있습니다.

## bot

인터넷을 통해 자동화된 태스크를 실행하고 인적 활동이나 상호 작용을 시뮬레이션하는 소프트웨어 애플리케이션입니다. 인터넷에서 정보를 인덱싱하는 웹 크롤러와 같이 유용하거나 이로운 봇도 있습니다. 악성 봇이라고 하는 다른 일부 봇은 개인 또는 조직을 방해하거나 해를 입히기 위한 봇입니다.

## 봇넷

[맬웨어](#)에 감염되고 봇 허더 또는 봇 운영자와 같은 단일 당사자가 제어하는 [봇](#) 네트워크입니다. 봇넷은 봇의 규모와 봇의 영향 범위를 확대하는 가장 잘 알려진 메커니즘입니다.

## 브랜치

코드 리포지토리의 포함된 영역입니다. 리포지토리에 생성되는 첫 번째 브랜치가 기본 브랜치입니다. 기존 브랜치에서 새 브랜치를 생성한 다음 새 브랜치에서 기능을 개발하거나 버그를 수정할 수 있습니다. 기능을 구축하기 위해 생성하는 브랜치를 일반적으로 기능 브랜치라고 합니다. 기능을 출시할 준비가 되면 기능 브랜치를 기본 브랜치에 다시 병합합니다. 자세한 내용은 [About branches](#)(GitHub 설명서)를 참조하십시오.

## 긴급 액세스 권한

예외적인 상황에서 승인된 프로세스를 통해 사용자가 일반적으로 액세스할 권한이 없는데 액세스할 수 있는 빠른 방법입니다. 자세한 내용은 AWS Well-Architected 지침의 [Implement break-glass procedures](#) 지표를 참조하세요.

## 브라운필드 전략

사용자 환경의 기존 인프라 시스템 아키텍처에 브라운필드 전략을 채택할 때는 현재 시스템 및 인프라의 제약 조건을 중심으로 아키텍처를 설계합니다. 기존 인프라를 확장하는 경우 브라운필드 전략과 [그린필드](#) 전략을 혼합할 수 있습니다.

## 버퍼 캐시

가장 자주 액세스하는 데이터가 저장되는 메모리 영역입니다.

## 사업 역량

기업이 가치를 창출하기 위해 하는 일(예: 영업, 고객 서비스 또는 마케팅)입니다. 마이크로서비스 아키텍처 및 개발 결정은 비즈니스 역량에 따라 이루어질 수 있습니다. 자세한 내용은 백서의 [AWS에서 컨테이너화된 마이크로서비스 실행의 비즈니스 역량 중심의 구성화](#) 섹션을 참조하십시오.

## 비즈니스 연속성 계획(BCP)

대규모 마이그레이션과 같은 중단 이벤트가 운영에 미치는 잠재적 영향을 해결하고 비즈니스가 신속하게 운영을 재개할 수 있도록 지원하는 계획입니다.

# C

## CAF

[AWS Cloud Adoption Framework](#)를 참조하세요.

## 카나리 배포

최종 사용자에게 제공하는 느린 증분 릴리스 버전입니다. 확신이 들면 새 버전을 배포하고 현재 버전을 완전히 교체합니다.

## CCoE

[클라우드 혁신 센터](#)를 참조하세요.

## CDC

[데이터 캡처 변경](#)을 참조하세요.

## 변경 데이터 캡처(CDC)

데이터베이스 테이블과 같은 데이터 소스의 변경 내용을 추적하고 변경 사항에 대한 메타데이터를 기록하는 프로세스입니다. 대상 시스템의 변경 내용을 감사하거나 복제하여 동기화를 유지하는 등의 다양한 용도로 CDC를 사용할 수 있습니다.

## 카오스 엔지니어링

시스템의 복원력을 테스트하기 위해 의도적으로 장애나 중단 이벤트를 도입합니다. [AWS Fault Injection Service \(AWS FIS\)](#)를 사용하여 AWS 워크로드에 스트레스를 주고 응답을 평가하는 실험을 수행할 수 있습니다.

## CI/CD

[지속적 통합 및 지속적 전송](#)을 참조하세요.

## 분류

예측을 생성하는 데 도움이 되는 분류 프로세스입니다. 분류 문제에 대한 ML 모델은 이산 값을 예측합니다. 이산 값은 항상 서로 다릅니다. 예를 들어, 모델이 이미지에 자동차가 있는지 여부를 평가해야 할 수 있습니다.

## 클라이언트측 암호화

대상이 데이터를 AWS 서비스 수신하기 전에 로컬에서 데이터를 암호화합니다.

## 클라우드 혁신 센터(CCoE)

클라우드 모범 사례 개발, 리소스 동원, 마이그레이션 타임라인 설정, 대규모 혁신을 통한 조직 선도 등 조직 전체에서 클라우드 채택 노력을 추진하는 다분야 팀입니다. 자세한 내용은 AWS 클라우드 엔터프라이즈 전략 블로그의 [CCoE 게시물](#)을 참조하세요.

## 클라우드 컴퓨팅

원격 데이터 스토리지와 IoT 디바이스 관리에 일반적으로 사용되는 클라우드 기술 클라우드 컴퓨팅은 일반적으로 [엣지 컴퓨팅](#) 기술에 연결되어 있습니다.

## 클라우드 운영 모델

IT 조직에서 하나 이상의 클라우드 환경을 구축, 성숙화 및 최적화하는 데 사용되는 운영 모델입니다. 자세한 내용은 [클라우드 운영 모델 구축](#)을 참조하십시오.

## 클라우드 채택 단계

조직이 AWS 클라우드로 마이그레이션할 때 일반적으로 거치는 4단계는 다음과 같습니다.

- 프로젝트 - 개념 증명 및 학습 목적으로 몇 가지 클라우드 관련 프로젝트 실행
- 기반 - 클라우드 채택 확장을 위한 기초 투자(예: 랜딩 존 생성, CCoE 정의, 운영 모델 구축)
- 마이그레이션 - 개별 애플리케이션 마이그레이션
- Re-invention - 제품 및 서비스 최적화와 클라우드 혁신

이러한 단계는 Stephen Orban이 블로그 게시물 [The Journey Toward Cloud-First and the Stages of Adoption](#) on the AWS 클라우드 Enterprise Strategy 블로그에서 정의했습니다. AWS 마이그레이션 전략과 어떤 관련이 있는지에 대한 자세한 내용은 [마이그레이션 준비 가이드](#)를 참조하세요.

## CMDB

[구성 관리 데이터베이스](#)를 참조하세요.

## 코드 리포지토리

소스 코드와 설명서, 샘플, 스크립트 등의 기타 자산이 버전 관리 프로세스를 통해 저장되고 업데이트되는 위치입니다. 일반적인 클라우드 리포지토리로 GitHub 또는 Bitbucket Cloud가 포함됩니다. 코드의 각 버전을 브랜치라고 합니다. 마이크로서비스 구조에서 각 리포지토리는 단일 기능 전용입니다. 단일 CI/CD 파이프라인은 여러 리포지토리를 사용할 수 있습니다.

## 콜드 캐시

비어 있거나, 제대로 채워지지 않았거나, 오래되었거나 관련 없는 데이터를 포함하는 버퍼 캐시입니다. 주 메모리나 디스크에서 데이터베이스 인스턴스를 읽어야 하기 때문에 성능에 영향을 미치며, 이는 버퍼 캐시에서 읽는 것보다 느립니다.

## 콜드 데이터

거의 액세스되지 않고 일반적으로 과거 데이터인 데이터. 이런 종류의 데이터를 쿼리할 때는 일반적으로 느린 쿼리가 허용됩니다. 이 데이터를 성능이 낮고 비용이 저렴한 스토리지 계층 또는 클래스로 옮기면 비용을 절감할 수 있습니다.

## 컴퓨터 비전(CV)

기계 학습을 사용하여 디지털 이미지 및 비디오와 같은 시각적 형식에서 정보를 분석하고 추출하는 [AI](#) 필드입니다. 예를 들어 Amazon SageMaker AI는 CV에 대한 이미지 처리 알고리즘을 제공합니다.

## 구성 드리프트

워크로드의 경우 구성이 예상되는 상태에서 변경됩니다. 이로 인해 워크로드가 규정을 준수하지 않을 수 있으며, 이는 일반적으로 점진적이고 의도되지 않은 작업입니다.

## 구성 관리 데이터베이스(CMDB)

하드웨어 및 소프트웨어 구성 요소와 해당 구성을 포함하여 데이터베이스와 해당 IT 환경에 대한 정보를 저장하고 관리하는 리포지토리입니다. 일반적으로 마이그레이션의 포트폴리오 탐색 및 분석 단계에서 CMDB의 데이터를 사용합니다.

### 규정 준수 팩

규정 준수 및 보안 검사를 사용자 지정하기 위해 조합할 수 있는 AWS Config 규칙 및 수정 작업 모음입니다. YAML 템플릿을 사용하여 적합성 팩을 AWS 계정 및 리전 또는 조직 전체에 단일 엔터티로 배포할 수 있습니다. 자세한 내용은 AWS Config 설명서의 [적합성 팩](#)을 참조하세요.

### 지속적 통합 및 지속적 전달(CI/CD)

소프트웨어 릴리스 프로세스의 소스, 빌드, 테스트, 스테이징 및 프로덕션 단계를 자동화하는 프로세스입니다. CI/CD는 일반적으로 파이프라인으로 설명됩니다. CI/CD를 통해 프로세스를 자동화하고, 생산성을 높이고, 코드 품질을 개선하고, 더 빠르게 제공할 수 있습니다. 자세한 내용은 [지속적 전달의 이점](#)을 참조하십시오. CD는 지속적 배포를 의미하기도 합니다. 자세한 내용은 [지속적 전달\(Continuous Delivery\)](#)과 [지속적인 개발](#)을 참조하십시오.

### CV

[컴퓨터 비전](#)을 참조하세요.

## D

### 저장 데이터

스토리지에 있는 데이터와 같이 네트워크에 고정되어 있는 데이터입니다.

### 데이터 분류

중요도와 민감도를 기준으로 네트워크의 데이터를 식별하고 분류하는 프로세스입니다. 이 프로세스는 데이터에 대한 적절한 보호 및 보존 제어를 결정하는 데 도움이 되므로 사이버 보안 위험 관리 전략의 중요한 구성 요소입니다. 데이터 분류는 AWS Well-Architected Framework의 보안 원칙 구성 요소입니다. 자세한 내용은 [데이터 분류](#)를 참조하십시오.

### 데이터 드리프트

프로덕션 데이터와 ML 모델 학습에 사용된 데이터 간의 상당한 차이 또는 시간 경과에 따른 입력 데이터의 의미 있는 변화. 데이터 드리프트는 ML 모델 예측의 전반적인 품질, 정확성 및 공정성을 저하시킬 수 있습니다.

## 전송 중 데이터

네트워크를 통과하고 있는 데이터입니다. 네트워크 리소스 사이를 이동 중인 데이터를 예로 들 수 있습니다.

## 데이터 메시

중앙 집중식 관리 및 거버넌스를 통해 분산되고 탈중앙화된 데이터 소유권을 제공하는 아키텍처 프레임워크입니다.

## 데이터 최소화

꼭 필요한 데이터만 수집하고 처리하는 원칙입니다. 에서 데이터를 최소화하면 개인 정보 보호 위험, 비용 및 분석 탄소 발자국을 줄일 수 있습니다.

## 데이터 경계

신뢰할 수 있는 자격 증명만 예상 네트워크에서 신뢰할 수 있는 리소스에 액세스하도록 하는 데 도움이 되는 AWS 환경의 예방 가드레일 세트입니다. 자세한 내용은 [데이터 경계 구축을 참조하세요 AWS](#).

## 데이터 사전 처리

원시 데이터를 ML 모델이 쉽게 구문 분석할 수 있는 형식으로 변환하는 것입니다. 데이터를 사전 처리한다는 것은 특정 열이나 행을 제거하고 누락된 값, 일관성이 없는 값 또는 중복 값을 처리함을 의미할 수 있습니다.

## 데이터 출처

라이프사이클 전반에 걸쳐 데이터의 출처와 기록을 추적하는 프로세스(예: 데이터 생성, 전송, 저장 방법).

## 데이터 주체

데이터를 수집 및 처리하는 개인입니다.

## 데이터 웨어하우스

분석과 같은 비즈니스 인텔리전스를 지원하는 데이터 관리 시스템입니다. 데이터 웨어하우스에는 보통 많은 양의 기록 데이터가 포함되며 일반적으로 쿼리 및 분석에 사용됩니다.

## 데이터 정의 언어(DDL)

데이터베이스에서 테이블 및 객체의 구조를 만들거나 수정하기 위한 명령문 또는 명령입니다.

## 데이터베이스 조작 언어(DML)

데이터베이스에서 정보를 수정(삽입, 업데이트 및 삭제)하기 위한 명령문 또는 명령입니다.

## DDL

[데이터 정의 언어](#)를 참조하세요.

### 딥 앙상블

예측을 위해 여러 딥 러닝 모델을 결합하는 것입니다. 딥 앙상블을 사용하여 더 정확한 예측을 얻거나 예측의 불확실성을 추정할 수 있습니다.

### 딥 러닝

여러 계층의 인공 신경망을 사용하여 입력 데이터와 관심 대상 변수 간의 매핑을 식별하는 ML 하위 분야입니다.

### 심층 방어

네트워크와 그 안의 데이터 기밀성, 무결성 및 가용성을 보호하기 위해 컴퓨터 네트워크 전체에 일련의 보안 메커니즘과 제어를 신중하게 계층화하는 정보 보안 접근 방식입니다. 이 전략을 채택하면 AWS Organizations 구조의 여러 계층에 여러 제어를 AWS 추가하여 리소스를 보호할 수 있습니다. 예를 들어, 심층 방어 접근 방식은 다단계 인증, 네트워크 세분화 및 암호화를 결합할 수 있습니다.

### 위임된 관리자

에서 AWS Organizations 호환되는 서비스는 AWS 멤버 계정을 등록하여 조직의 계정을 관리하고 해당 서비스에 대한 권한을 관리할 수 있습니다. 이러한 계정을 해당 서비스의 위임된 관리자라고 합니다. 자세한 내용과 호환되는 서비스 목록은 AWS Organizations 설명서의 [AWS Organizations와 함께 사용할 수 있는 AWS 서비스](#)를 참조하십시오.

### 배포

대상 환경에서 애플리케이션, 새 기능 또는 코드 수정 사항을 사용할 수 있도록 하는 프로세스입니다. 배포에는 코드 베이스의 변경 사항을 구현한 다음 애플리케이션 환경에서 해당 코드베이스를 구축하고 실행하는 작업이 포함됩니다.

### 개발 환경

[환경](#)을 참조하세요.

### 탐지 제어

이벤트 발생 후 탐지, 기록 및 알림을 수행하도록 설계된 보안 제어입니다. 이러한 제어는 기존의 예방적 제어를 우회한 보안 이벤트를 알리는 2차 방어선입니다. 자세한 내용은 AWS에서 보안 제어 구현의 [탐지 제어](#)를 참조하세요.

## 개발 가치 흐름 매핑 (DVSM)

소프트웨어 개발 라이프사이클에서 속도와 품질에 부정적인 영향을 미치는 제약 조건을 식별하고 우선 순위를 지정하는 데 사용되는 프로세스입니다. DVSM은 원래 린 제조 방식을 위해 설계된 가치 흐름 매핑 프로세스를 확장합니다. 소프트웨어 개발 프로세스를 통해 가치를 창출하고 이동하는 데 필요한 단계와 팀에 중점을 둡니다.

### 디지털 트윈

건물, 공장, 산업 장비 또는 생산 라인과 같은 실제 시스템을 가상으로 표현한 것입니다. 디지털 트윈은 예측 유지 보수, 원격 모니터링, 생산 최적화를 지원합니다.

### 차원 테이블

[스타 스키마](#)에서 팩트 테이블의 정량적 데이터에 대한 데이터 속성을 포함하는 더 작은 테이블을 말합니다. 차원 테이블 속성은 일반적으로 텍스트 필드나 텍스트처럼 동작하는 개별 숫자입니다. 이러한 속성은 보통 쿼리 제약, 필터링 및 결과 세트 레이블 지정에 사용됩니다.

### 재해

워크로드 또는 시스템이 기본 배포 위치에서 비즈니스 목표를 달성하지 못하게 방해하는 이벤트입니다. 이러한 이벤트는 자연재해, 기술적 오류, 의도하지 않은 구성 오류 또는 멀웨어 공격과 같은 사람의 행동으로 인한 결과일 수 있습니다.

### 재해 복구(DR)

[재해](#)로 인한 가동 중지 시간 및 데이터 손실을 최소화하기 위해 사용하는 전략 및 프로세스입니다. 자세한 내용은 AWS Well-Architected Framework의 [Disaster Recovery of Workloads on AWS: Recovery in the Cloud](#)를 참조하세요.

### DML

[데이터베이스 조작 언어](#)를 참조하세요.

### 도메인 기반 설계

구성 요소를 각 구성 요소가 제공하는 진화하는 도메인 또는 핵심 비즈니스 목표에 연결하여 복잡한 소프트웨어 시스템을 개발하는 접근 방식입니다. 이 개념은 에릭 에반스에 의해 그의 저서인 도메인 기반 디자인: 소프트웨어 중심의 복잡성 해결(Boston: Addison-Wesley Professional, 2003)에서 소개되었습니다. Strangler Fig 패턴과 함께 도메인 기반 설계를 사용하는 방법에 대한 자세한 내용은 [컨테이너 및 Amazon API Gateway를 사용하여 기존의 Microsoft ASP.NET\(ASMX\) 웹 서비스를 점진적으로 현대화하는 방법](#)을 참조하십시오.

### DR

[재해 복구](#)를 참조하세요.

## 드리프트 감지

기준이 되는 구성과의 편차 추적을 말합니다. 예를 들어 AWS CloudFormation 를 사용하여 [시스템 리소스의 드리프트를 감지](#)하거나 사용하여 AWS Control Tower 거버넌스 요구 사항 준수에 영향을 미칠 수 있는 [랜딩 존의 변경 사항을 감지](#)할 수 있습니다.

## DVSM

[개발 가치 흐름 매핑](#)을 참조하세요.

## E

### EDA

[탐색 데이터 분석](#)을 참조하세요.

### EDI

[전자 데이터 교환](#)을 참조하세요.

## 엣지 컴퓨팅

IoT 네트워크의 엣지에서 스마트 디바이스의 컴퓨팅 성능을 개선하는 기술 엣지 컴퓨팅은 [클라우드 컴퓨팅](#)에 비해 보다 통신 지연 시간을 줄이고 응답 시간을 개선할 수 있습니다.

## 전자 데이터 교환(EDI)

조직 간 비즈니스 문서의 자동화된 교환을 나타냅니다. 자세한 내용은 [전자 데이터 교환\(EDI\)이란 무엇인가요?](#)를 참조하세요.

## 암호화

사람이 읽을 수 있는 일반 텍스트 데이터를 사이버텍스트로 변환하는 컴퓨팅 프로세스입니다.

## 암호화 키

암호화 알고리즘에 의해 생성되는 무작위 비트의 암호화 문자열입니다. 키의 길이는 다양할 수 있으며 각 키는 예측할 수 없고 고유하게 설계되었습니다.

## 엔디안

컴퓨터 메모리에 바이트가 저장되는 순서입니다. 빅 엔디안 시스템은 가장 중요한 바이트를 먼저 저장합니다. 리틀 엔디안 시스템은 가장 덜 중요한 바이트를 먼저 저장합니다.

## 엔드포인트

[서비스 엔드포인트](#)를 참조하세요.

## 엔드포인트 서비스

Virtual Private Cloud(VPC)에서 호스팅하여 다른 사용자와 공유할 수 있는 서비스입니다. 를 사용하여 엔드포인트 서비스를 생성하고 다른 AWS 계정 또는 AWS Identity and Access Management (IAM) 보안 주체에 권한을 AWS PrivateLink 부여할 수 있습니다. 이러한 계정 또는 보안 주체는 인터페이스 VPC 엔드포인트를 생성하여 엔드포인트 서비스에 비공개로 연결할 수 있습니다. 자세한 내용은 Amazon Virtual Private Cloud(VPC) 설명서의 [엔드포인트 서비스 생성](#)을 참조하십시오.

## 엔터프라이즈 리소스 계획(ERP)

엔터프라이즈의 주요 비즈니스 프로세스(예: 회계, [MES](#), 프로젝트 관리)를 자동화하고 관리하는 시스템입니다.

## 봉투 암호화

암호화 키를 다른 암호화 키로 암호화하는 프로세스입니다. 자세한 내용은 AWS Key Management Service (AWS KMS) 설명서의 [봉투 암호화](#)를 참조하세요.

## 환경

실행 중인 애플리케이션의 인스턴스입니다. 다음은 클라우드 컴퓨팅의 일반적인 환경 유형입니다.

- 개발 환경 - 애플리케이션 유지 관리를 담당하는 핵심 팀만 사용할 수 있는 실행 중인 애플리케이션의 인스턴스입니다. 개발 환경은 변경 사항을 상위 환경으로 승격하기 전에 테스트하는 데 사용됩니다. 이러한 유형의 환경을 테스트 환경이라고도 합니다.
- 하위 환경 - 초기 빌드 및 테스트에 사용되는 환경을 비롯한 애플리케이션의 모든 개발 환경입니다.
- 프로덕션 환경 - 최종 사용자가 액세스할 수 있는 실행 중인 애플리케이션의 인스턴스입니다. CI/CD 파이프라인에서 프로덕션 환경이 마지막 배포 환경입니다.
- 상위 환경 - 핵심 개발 팀 이외의 사용자가 액세스할 수 있는 모든 환경입니다. 프로덕션 환경, 프로덕션 이전 환경 및 사용자 수용 테스트를 위한 환경이 여기에 포함될 수 있습니다.

## 에픽

애자일 방법론에서 작업을 구성하고 우선순위를 정하는 데 도움이 되는 기능적 범주입니다. 에픽은 요구 사항 및 구현 작업에 대한 개괄적인 설명을 제공합니다. 예를 들어, AWS CAF 보안 에픽에는 ID 및 액세스 관리, 탐지 제어, 인프라 보안, 데이터 보호 및 인시던트 대응이 포함됩니다. AWS 마이그레이션 전략의 에픽에 대한 자세한 내용은 [프로그램 구현 가이드](#)를 참조하십시오.

## ERP

[엔터프라이즈 리소스 계획](#)을 참조하세요.

## 탐색 데이터 분석(EDA)

데이터 세트를 분석하여 주요 특성을 파악하는 프로세스입니다. 데이터를 수집 또는 집계한 다음 초기 조사를 수행하여 패턴을 찾고, 이상을 탐지하고, 가정을 확인합니다. EDA는 요약 통계를 계산하고 데이터 시각화를 생성하여 수행됩니다.

## F

### 팩트 테이블

[스타 스키마](#)의 중앙 테이블입니다. 비즈니스 운영에 대한 정량적 데이터를 저장합니다. 일반적으로 팩트 테이블은 측정값이 있는 열 및 차원 테이블에 대한 외래 키가 있는 열과 같이 두 가지 열 유형을 포함합니다.

### 빠른 실패

개발 수명 주기를 줄이기 위해 빈번한 증분 테스트를 사용하는 철학입니다. 애자일 접근 방식의 핵심입니다.

### 장애 격리 경계

에서 장애의 영향을 제한하고 워크로드의 복원력을 개선하는 데 도움이 되는 가용 영역, AWS 리전 컨트롤 플레인 또는 데이터 플레인과 같은 AWS 클라우드경계입니다. 자세한 내용은 [AWS 장애 격리 경계](#)를 참조하세요.

### 기능 브랜치

[브랜치](#)를 참조하세요.

### 기능

예측에 사용하는 입력 데이터입니다. 예를 들어, 제조 환경에서 기능은 제조 라인에서 주기적으로 캡처되는 이미지일 수 있습니다.

### 기능 중요도

모델의 예측에 특성이 얼마나 중요한지를 나타냅니다. 이는 일반적으로 SHAP(Shapley Additive Descriptions) 및 통합 그래디언트와 같은 다양한 기법을 통해 계산할 수 있는 수치 점수로 표현됩니다. 자세한 내용은 [기계 학습 모델 해석 가능성을 참조하세요 AWS](#).

### 기능 변환

추가 소스로 데이터를 보강하거나, 값을 조정하거나, 단일 데이터 필드에서 여러 정보 세트를 추출하는 등 ML 프로세스를 위해 데이터를 최적화하는 것입니다. 이를 통해 ML 모델이 데이터를 활용

할 수 있습니다. 예를 들어, 날짜 '2021-05-27 00:15:37'을 '2021년', '5월', '목', '15일'로 분류하면 학습 알고리즘이 다양한 데이터 구성 요소와 관련된 미묘한 패턴을 학습하는 데 도움이 됩니다.

### 퓨샷 프롬프팅

유사한 태스크를 수행하도록 요청하기 전에 [LLM](#)에 태스크와 원하는 출력을 보여주는 몇 가지 예제를 제공합니다. 이 기법은 모델이 프롬프트에 포함된 예제(샷)에서 학습하는 컨텍스트 내 학습을 적용합니다. 퓨샷 프롬프팅은 특정 형식 지정, 추론 또는 분야별 지식이 필요한 태스크에 효과적일 수 있습니다. [제로샷 프롬프팅](#)도 참조하세요.

### FGAC

[세분화된 액세스 제어](#)를 참조하세요.

#### 세분화된 액세스 제어(FGAC)

여러 조건을 사용하여 액세스 요청을 허용하거나 거부합니다.

### 플래시컷 마이그레이션

단계적 접근 방식을 사용하는 대신 [변경 데이터 캡처](#)를 통해 지속적 데이터 복제를 사용하여 최단 시간에 데이터를 마이그레이션하는 데이터베이스 마이그레이션 방법입니다. 목표는 가동 중지 시간을 최소화하는 것입니다.

### FM

[파운데이션 모델](#)을 참조하세요.

#### 파운데이션 모델(FM)

일반화되고 레이블이 지정되지 않은 데이터의 대규모 데이터세트에서 훈련된 대규모 딥 러닝 신경망입니다. FM은 언어 이해, 텍스트 및 이미지 생성, 자연어 대화와 같은 다양한 일반 태스크를 수행할 수 있습니다. 자세한 내용은 [파운데이션 모델이란?](#)을 참조하세요.

## G

### 생성형 AI

대량의 데이터에서 훈련되었으며 간단한 텍스트 프롬프트를 사용하여 이미지, 비디오, 텍스트, 오디오와 같은 새 콘텐츠와 아티팩트를 생성할 수 있는 [AI](#) 모델의 하위 세트입니다. 자세한 내용은 [생성형 AI란 무엇인가요?](#)를 참조하세요.

#### 지리적 차단

[지리적 제한](#)을 참조하세요.

## 지리적 제한(지리적 차단)

Amazon CloudFront에서 특정 국가의 사용자가 콘텐츠 배포에 액세스하지 못하도록 하는 옵션입니다. 허용 목록 또는 차단 목록을 사용하여 승인된 국가와 차단된 국가를 지정할 수 있습니다. 자세한 내용은 CloudFront 설명서의 [콘텐츠의 지리적 배포 제한](#)을 참조하십시오.

## Gitflow 워크플로

하위 환경과 상위 환경이 소스 코드 리포지토리의 서로 다른 브랜치를 사용하는 방식입니다. Gitflow 워크플로는 레거시로 간주되며 [트렁크 기반 워크플로](#)는 선호되는 현대적 접근 방식입니다.

## 골든 이미지

시스템 또는 소프트웨어의 새 인스턴스를 배포하기 위한 템플릿으로 사용되는 해당 시스템 또는 소프트웨어의 스냅샷입니다. 예를 들어 제조 분야에서는 골든 이미지를 사용하여 여러 디바이스에서 소프트웨어를 프로비저닝할 수 있으며 이를 통해 디바이스 제조 작업의 속도, 확장성 및 생산성을 개선할 수 있습니다.

## 브라운필드 전략

새로운 환경에서 기존 인프라의 부재 시스템 아키텍처에 대한 그린필드 전략을 채택할 때 [브라운필드](#)라고도 하는 기존 인프라와의 호환성 제한 없이 모든 새로운 기술을 선택할 수 있습니다. 기존 인프라를 확장하는 경우 브라운필드 전략과 그린필드 전략을 혼합할 수 있습니다.

## 가드레일

조직 단위(OU) 전체에서 리소스, 정책 및 규정 준수를 관리하는 데 도움이 되는 중요 규칙입니다. 예방 가드레일은 규정 준수 표준에 부합하도록 정책을 시행하며, 서비스 제어 정책과 IAM 권한 경계를 사용하여 구현됩니다. 탐지 가드레일은 정책 위반 및 규정 준수 문제를 감지하고 해결을 위한 알림을 생성하며, 이는 AWS Config, Amazon GuardDuty, AWS Security Hub CSPM, , AWS Trusted Advisor, Amazon Inspector 및 사용자 지정 AWS Lambda 검사를 사용하여 구현됩니다.

# H

## HA

[고가용성](#)을 참조하세요.

## 이기종 데이터베이스 마이그레이션

다른 데이터베이스 엔진을 사용하는 대상 데이터베이스로 소스 데이터베이스 마이그레이션(예: Oracle에서 Amazon Aurora로) 이기종 마이그레이션은 일반적으로 리아키텍트 작업의 일부이며 스

키마를 변환하는 것은 복잡한 작업일 수 있습니다. AWS 는 스키마 변환에 도움이 되는 [AWS SCT](#)를 제공합니다.

## 높은 가용성(HA)

문제나 재해 발생 시 개입 없이 지속적으로 운영할 수 있는 워크로드의 능력. HA 시스템은 자동으로 장애 조치되고, 지속적으로 고품질 성능을 제공하고, 성능에 미치는 영향을 최소화하면서 다양한 부하와 장애를 처리하도록 설계되었습니다.

## 히스토리언 현대화

제조 산업의 요구 사항을 더 잘 충족하도록 운영 기술(OT) 시스템을 현대화하고 업그레이드하는 데 사용되는 접근 방식입니다. 히스토리언은 공장의 다양한 출처에서 데이터를 수집하고 저장하는 데 사용되는 일종의 데이터베이스입니다.

## 홀드아웃 데이터

[기계 학습](#) 모델을 훈련하는 데 사용되는 데이터세트에서 보류되는 레이블이 지정된 기록 데이터의 일부입니다. 홀드아웃 데이터를 사용하여 모델 예측을 홀드아웃 데이터와 비교해 모델 성능을 평가할 수 있습니다.

## 동종 데이터베이스 마이그레이션

동일한 데이터베이스 엔진을 공유하는 대상 데이터베이스로 소스 데이터베이스 마이그레이션(예: Microsoft SQL Server에서 Amazon RDS for SQL Server로) 동종 마이그레이션은 일반적으로 리호스팅 또는 리플랫폼 작업의 일부입니다. 네이티브 데이터베이스 유틸리티를 사용하여 스키마를 마이그레이션할 수 있습니다.

## 핫 데이터

자주 액세스하는 데이터(예: 실시간 데이터 또는 최근 번역 데이터). 일반적으로 이 데이터에는 빠른 쿼리 응답을 제공하기 위한 고성능 스토리지 계층 또는 클래스가 필요합니다.

## 핫픽스

프로덕션 환경의 중요한 문제를 해결하기 위한 긴급 수정입니다. 핫픽스는 긴급하기 때문에 일반적인 DevOps 릴리스 워크플로 외부에서 실행됩니다.

## 하이퍼케어 기간

전환 직후 마이그레이션 팀이 문제를 해결하기 위해 클라우드에서 마이그레이션된 애플리케이션을 관리하고 모니터링하는 기간입니다. 일반적으로 이 기간은 1~4일입니다. 하이퍼케어 기간이 끝나면 마이그레이션 팀은 일반적으로 애플리케이션에 대한 책임을 클라우드 운영 팀에 넘깁니다.

## I

## IaC

[코드형 인프라](#)를 참조하세요.

## 자격 증명 기반 정책

AWS 클라우드 환경 내에서 권한을 정의하는 하나 이상의 IAM 보안 주체에 연결된 정책입니다.

## 유휴 애플리케이션

90일 동안 평균 CPU 및 메모리 사용량이 5~20%인 애플리케이션입니다. 마이그레이션 프로젝트에서는 이러한 애플리케이션을 사용 중지하거나 온프레미스에 유지하는 것이 일반적입니다.

## IIoT

[산업용 사물 인터넷](#)을 참조하세요.

## 변경 불가능한 인프라

기존 인프라를 업데이트, 패치 또는 수정하는 대신 프로덕션 워크로드에 대한 새 인프라를 배포하는 모델입니다. 변경 불가능한 인프라는 [변경 가능한 인프라](#)보다 본질적으로 더 일관되고 안정적이며 예측 가능합니다. 자세한 내용은 AWS Well-Architected Framework의 [변경 불가능한 인프라를 사용하여 배포](#) 모범 사례를 참조하세요.

## 인바운드(수신) VPC

AWS 다중 계정 아키텍처에서 애플리케이션 외부에서 네트워크 연결을 수락, 검사 및 라우팅하는 VPC입니다. [AWS Security Reference Architecture](#)에서는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPC로 네트워크 계정을 설정할 것을 권장합니다.

## 증분 마이그레이션

한 번에 전체 전환을 수행하는 대신 애플리케이션을 조금씩 마이그레이션하는 전환 전략입니다. 예를 들어, 처음에는 소수의 마이크로서비스나 사용자만 새 시스템으로 이동할 수 있습니다. 모든 것이 제대로 작동하는지 확인한 후에는 레거시 시스템을 폐기할 수 있을 때까지 추가 마이크로서비스 또는 사용자를 점진적으로 이동할 수 있습니다. 이 전략을 사용하면 대규모 마이그레이션과 관련된 위험을 줄일 수 있습니다.

## Industry 4.0

연결성, 실시간 데이터, 자동화, 분석 및 AI/ML의 발전을 통해 제조 프로세스의 현대화를 나타내기 위해 2016년에 [Klaus Schwab](#)에서 도입한 용어입니다.

## 인프라

애플리케이션의 환경 내에 포함된 모든 리소스와 자산입니다.

### 코드형 인프라(IaC)

구성 파일 세트를 통해 애플리케이션의 인프라를 프로비저닝하고 관리하는 프로세스입니다. IaC는 새로운 환경의 반복 가능성, 신뢰성 및 일관성을 위해 인프라 관리를 중앙 집중화하고, 리소스를 표준화하고, 빠르게 확장할 수 있도록 설계되었습니다.

### 산업용 사물 인터넷(IIoT)

제조, 에너지, 자동차, 의료, 생명과학, 농업 등의 산업 부문에서 인터넷에 연결된 센서 및 디바이스의 사용 자세한 내용은 [산업용 사물 인터넷\(IoT\) 디지털 트랜스포메이션 전략 구축](#)을 참조하십시오.

### 검사 VPC

AWS 다중 계정 아키텍처에서는 VPC(동일하거나 다른 AWS 리전), 인터넷 및 온프레미스 네트워크 간의 네트워크 트래픽 검사를 관리하는 중앙 집중식 VPCs입니다. [AWS Security Reference Architecture](#)에서는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPC로 네트워크 계정을 설정할 것을 권장합니다.

### 사물 인터넷(IoT)

인터넷이나 로컬 통신 네트워크를 통해 다른 디바이스 및 시스템과 통신하는 센서 또는 프로세서가 내장된 연결된 물리적 객체의 네트워크 자세한 내용은 [IoT란?](#)을 참조하십시오.

### 해석력

모델의 예측이 입력에 따라 어떻게 달라지는지를 사람이 이해할 수 있는 정도를 설명하는 기계 학습 모델의 특성입니다. 자세한 내용은 [기계 학습 모델 해석 가능성을 참조하세요 AWS](#).

### IoT

[사물 인터넷](#)을 참조하세요.

### IT 정보 라이브러리(ITIL)

IT 서비스를 제공하고 이러한 서비스를 비즈니스 요구 사항에 맞게 조정하기 위한 일련의 모범 사례 ITIL은 ITSM의 기반을 제공합니다.

### IT 서비스 관리(ITSM)

조직의 IT 서비스 설계, 구현, 관리 및 지원과 관련된 활동 클라우드 운영을 ITSM 도구와 통합하는 방법에 대한 자세한 내용은 [운영 통합 가이드](#)를 참조하십시오.

## ITIL

[IT 정보 라이브러리](#)를 참조하세요.

## ITSM

[IT 서비스 관리](#)를 참조하세요.

## L

### 레이블 기반 액세스 제어(LBAC)

사용자 및 데이터 자체에 각각 보안 레이블 값을 명시적으로 할당하는 필수 액세스 제어(MAC)를 구현한 것입니다. 사용자 보안 레이블과 데이터 보안 레이블 간의 교차 부분에 따라 사용자가 볼 수 있는 행과 열이 결정됩니다.

### 랜딩 존

랜딩 존은 확장 가능하고 안전한 잘 설계된 다중 계정 AWS 환경입니다. 조직은 여기에서부터 보안 및 인프라 환경에 대한 확신을 가지고 워크로드와 애플리케이션을 신속하게 시작하고 배포할 수 있습니다. 랜딩 존에 대한 자세한 내용은 [안전하고 확장 가능한 다중 계정 AWS 환경 설정](#)을 참조하십시오.

### 대규모 언어 모델(LLM)

방대한 양의 데이터에서 사전 훈련된 딥 러닝 AI 모델입니다. LLM은 질문에 대한 답변, 문서 요약, 텍스트를 다른 언어로 번역, 문장 완성과 같은 여러 태스크를 수행할 수 있습니다. 자세한 내용은 [대규모 언어 모델\(LLM\)이란 무엇인가요?](#)를 참조하세요.

### 대규모 마이그레이션

300대 이상의 서버 마이그레이션입니다.

### LBAC

[레이블 기반 액세스 제어](#)를 참조하세요.

### 최소 권한

작업을 수행하는 데 필요한 최소 권한을 부여하는 보안 모범 사례입니다. 자세한 내용은 IAM 설명서의 [최소 권한 적용](#)을 참조하십시오.

### 리프트 앤드 시프트

[7R](#)을 참조하세요.

## 리틀 엔디안 시스템

가장 덜 중요한 바이트를 먼저 저장하는 시스템입니다. [엔디안](#)도 참조하세요.

## LLM

[대규모 언어 모델](#)을 참조하세요.

## 하위 환경

[환경](#)을 참조하세요.

## M

### 기계 학습(ML)

패턴 인식 및 학습에 알고리즘과 기법을 사용하는 인공지능의 한 유형입니다. ML은 사물 인터넷 (IoT) 데이터와 같은 기록된 데이터를 분석하고 학습하여 패턴을 기반으로 통계 모델을 생성합니다. 자세한 내용은 [기계 학습](#)을 참조하십시오.

### 기본 브랜치

[브랜치](#)를 참조하세요.

### 맬웨어

컴퓨터 보안 또는 프라이버시를 위협하도록 설계된 소프트웨어입니다. 맬웨어는 컴퓨터 시스템을 방해하거나 민감한 정보를 유출하거나 무단 액세스 권한을 확보할 수 있습니다. 맬웨어의 예로 바이러스, 웜, 랜섬웨어, 트로이 목마, 스파이웨어, 키로거 등이 있습니다.

### 관리형 서비스

AWS 서비스는 인프라 계층, 운영 체제 및 플랫폼을 AWS 운영하고, 사용자는 엔드포인트에 액세스하여 데이터를 저장하고 검색합니다. 관리형 서비스의 예로 Amazon Simple Storage Service(Amazon S3) 및 Amazon DynamoDB가 있습니다. 이를 추상화된 서비스라고도 합니다.

### 제조 실행 시스템(MES)

원자재를 생산 현장에서 완제품으로 변환하는 생산 프로세스를 추적, 모니터링, 문서화 및 제어하기 위한 소프트웨어 시스템입니다.

## MAP

[Migration Acceleration Program](#)을 참조하세요.

## 메커니즘

도구를 생성하고 도구 채택을 유도한 다음 조정을 위해 결과를 검사하는 전체 프로세스입니다. 메커니즘은 작동 시 자체적으로 강화하고 개선하는 주기입니다. 자세한 내용은 AWS Well-Architected Framework의 [메커니즘 구축](#)을 참조하세요.

## 멤버 계정

조직의 일부인 관리 계정을 AWS 계정 제외한 모든 계정. AWS Organizations 하나의 계정은 한 번에 하나의 조직 멤버만 될 수 있습니다.

## MES

[제조 실행 시스템](#)을 참조하세요.

## 메시지 큐 원격 분석 전송(MQTT)

리소스 제약이 있는 [IoT](#) 디바이스에 대한 [게시 및 구독](#) 패턴을 기반으로 하는 경량 Machine-to-Machine(M2M) 통신 프로토콜입니다.

## 마이크로서비스

잘 정의된 API를 통해 통신하고 일반적으로 소규모 자체 팀이 소유하는 소규모 독립 서비스입니다. 예를 들어, 보험 시스템에는 영업, 마케팅 등의 비즈니스 역량이나 구매, 청구, 분석 등의 하위 영역에 매핑되는 마이크로 서비스가 포함될 수 있습니다. 마이크로서비스의 이점으로 민첩성, 유연한 확장, 손쉬운 배포, 재사용 가능한 코드, 복원력 등이 있습니다. 자세한 내용은 [AWS 서버리스 서비스를 사용하여 마이크로서비스 통합을 참조하세요](#).

## 마이크로서비스 아키텍처

각 애플리케이션 프로세스를 마이크로서비스로 실행하는 독립 구성 요소를 사용하여 애플리케이션을 구축하는 접근 방식입니다. 이러한 마이크로서비스는 경량 API를 사용하여 잘 정의된 인터페이스를 통해 통신합니다. 애플리케이션의 특정 기능에 대한 수요에 맞게 이 아키텍처의 각 마이크로 서비스를 업데이트, 배포 및 조정할 수 있습니다. 자세한 내용은 [에서 마이크로서비스 구현을 참조하세요 AWS](#).

## Migration Acceleration Program(MAP)

조직이 클라우드로 전환하기 위한 강력한 운영 기반을 구축하고 초기 마이그레이션 비용을 상쇄하는 데 도움이 되는 컨설팅 지원, 교육 및 서비스를 제공하는 AWS 프로그램입니다. MAP에는 레거시 마이그레이션을 체계적인 방식으로 실행하기 위한 마이그레이션 방법론과 일반적인 마이그레이션 시나리오를 자동화하고 가속화하는 도구 세트가 포함되어 있습니다.

## 대규모 마이그레이션

애플리케이션 포트폴리오의 대다수를 웨이브를 통해 클라우드로 이동하는 프로세스로, 각 웨이브에서 더 많은 애플리케이션이 더 빠른 속도로 이동합니다. 이 단계에서는 이전 단계에서 배운 모범 사례와 교훈을 사용하여 팀, 도구 및 프로세스의 마이그레이션 팩토리를 구현하여 자동화 및 민첩한 제공을 통해 워크로드 마이그레이션을 간소화합니다. 이것은 [AWS 마이그레이션 전략](#)의 세 번째 단계입니다.

### 마이그레이션 팩토리

자동화되고 민첩한 접근 방식을 통해 워크로드 마이그레이션을 간소화하는 다기능 팀입니다. 마이그레이션 팩토리 팀에는 일반적으로 스프린트에서 일하는 운영, 비즈니스 분석가 및 소유자, 마이그레이션 엔지니어, 개발자, DevOps 전문가가 포함됩니다. 엔터프라이즈 애플리케이션 포트폴리오의 20~50%는 공장 접근 방식으로 최적화할 수 있는 반복되는 패턴으로 구성되어 있습니다. 자세한 내용은 이 콘텐츠 세트의 [클라우드 마이그레이션 팩토리 가이드](#)와 [마이그레이션 팩토리에 대한 설명](#)을 참조하십시오.

### 마이그레이션 메타데이터

마이그레이션을 완료하는 데 필요한 애플리케이션 및 서버에 대한 정보 각 마이그레이션 패턴에는 서로 다른 마이그레이션 메타데이터 세트가 필요합니다. 마이그레이션 메타데이터의 예로는 대상 서브넷, 보안 그룹 및 AWS 계정이 있습니다.

### 마이그레이션 패턴

사용되는 마이그레이션 전략, 마이그레이션 대상, 마이그레이션 애플리케이션 또는 서비스를 자세히 설명하는 반복 가능한 마이그레이션 작업입니다. 예: AWS Application Migration Service를 사용하여 Amazon EC2로 마이그레이션을 리호스팅합니다.

### Migration Portfolio Assessment(MPA)

AWS 클라우드로 마이그레이션하는 비즈니스 사례를 검증하기 위한 정보를 제공하는 온라인 도구입니다. MPA는 상세한 포트폴리오 평가(서버 적정 규모 조정, 가격 책정, TCO 비교, 마이그레이션 비용 분석)와 마이그레이션 계획(애플리케이션 데이터 분석 및 데이터 수집, 애플리케이션 그룹화, 마이그레이션 우선순위 지정, 웨이브 계획)을 제공합니다. [MPA 도구](#)(로그인 필요)는 모든 AWS 컨설턴트와 APN 파트너 컨설턴트가 무료로 사용할 수 있습니다.

### 마이그레이션 준비 상태 평가(MRA)

AWS CAF를 사용하여 조직의 클라우드 준비 상태에 대한 인사이트를 얻고, 강점과 약점을 식별하고, 식별된 격차를 해소하기 위한 행동 계획을 수립하는 프로세스입니다. 자세한 내용은 [마이그레이션 준비 가이드](#)를 참조하십시오. MRA는 [AWS 마이그레이션 전략](#)의 첫 번째 단계입니다.

## 마이그레이션 전략

워크로드를 AWS 클라우드로 마이그레이션하는 데 사용되는 접근 방식입니다. 자세한 내용은 이 용어집의 [7R 항목](#)과 [조직을 동원하여 대규모 마이그레이션 가속화](#)를 참조하세요.

### ML

[기계 학습](#)을 참조하세요.

### 현대화

비용을 절감하고 효율성을 높이고 혁신을 활용하기 위해 구식(레거시 또는 모놀리식) 애플리케이션과 해당 인프라를 클라우드의 민첩하고 탄력적이고 가용성이 높은 시스템으로 전환하는 것입니다. 자세한 내용은 [AWS 클라우드에서 애플리케이션을 현대화하기 위한 전략](#)을 참조하세요.

### 현대화 준비 상태 평가

조직 애플리케이션의 현대화 준비 상태를 파악하고, 이점, 위험 및 종속성을 식별하고, 조직이 해당 애플리케이션의 향후 상태를 얼마나 잘 지원할 수 있는지를 확인하는 데 도움이 되는 평가입니다. 평가 결과는 대상 아키텍처의 청사진, 현대화 프로세스의 개발 단계와 마일스톤을 자세히 설명하는 로드맵 및 파악된 격차를 해소하기 위한 실행 계획입니다. 자세한 내용은 [AWS 클라우드에서 애플리케이션의 현대화 준비 상태 평가](#)를 참조하세요.

### 모놀리식 애플리케이션(모놀리식 유형)

긴밀하게 연결된 프로세스를 사용하여 단일 서비스로 실행되는 애플리케이션입니다. 모놀리식 애플리케이션에는 몇 가지 단점이 있습니다. 한 애플리케이션 기능에 대한 수요가 급증하면 전체 아키텍처 규모를 조정해야 합니다. 코드 베이스가 커지면 모놀리식 애플리케이션의 기능을 추가하거나 개선하는 것도 더 복잡해집니다. 이러한 문제를 해결하기 위해 마이크로서비스 아키텍처를 사용할 수 있습니다. 자세한 내용은 [마이크로서비스로 모놀리식 유형 분해](#)를 참조하십시오.

### MPA

[Migration Portfolio Assessment](#)를 참조하세요.

### MQTT

[메시지 큐 원격 분석 전송](#)을 참조하세요.

### 멀티클래스 분류

여러 클래스에 대한 예측(2개 이상의 결과 중 하나 예측)을 생성하는 데 도움이 되는 프로세스입니다. 예를 들어, ML 모델이 '이 제품은 책인가요, 자동차인가요, 휴대폰인가요?' 또는 '이 고객이 가장 관심을 갖는 제품 범주는 무엇인가요?'라고 물을 수 있습니다.

## 변경 가능한 인프라

프로덕션 워크로드에 대한 기존 인프라를 업데이트하고 수정하는 모델입니다. 일관성, 신뢰성 및 예측 가능성을 높이기 위해 AWS Well-Architected Framework에서는 [변경 불가능한 인프라](#)를 모범 사례로 사용할 것을 권장합니다.

## O

### OAC

[오리진 액세스 제어](#)를 참조하세요.

### OAI

[오리진 액세스 ID](#)를 참조하세요.

### OCM

[조직 변경 관리](#)를 참조하세요.

### 오프라인 마이그레이션

마이그레이션 프로세스 중 소스 워크로드가 중단되는 마이그레이션 방법입니다. 이 방법은 가동 중지 증가를 수반하며 일반적으로 작고 중요하지 않은 워크로드에 사용됩니다.

## OI

[운영 통합](#)을 참조하세요.

### OLA

[운영 수준 계약](#)을 참조하세요.

### 온라인 마이그레이션

소스 워크로드를 오프라인 상태로 전환하지 않고 대상 시스템에 복사하는 마이그레이션 방법입니다. 워크로드에 연결된 애플리케이션은 마이그레이션 중에도 계속 작동할 수 있습니다. 이 방법은 가동 중지 차단 또는 최소화를 수반하며 일반적으로 중요한 프로덕션 워크로드에 사용됩니다.

### OPC-UA

[Open Process Communications - Unified Architecture\(OPC-UA\)](#)를 참조하세요.

### Open Process Communications - Unified Architecture(OPC-UA)

산업 자동화를 위한 Machine-to-Machine(M2M) 통신 프로토콜입니다. OPC-UA는 데이터 암호화, 인증 및 권한 부여 체계에 관한 상호 운용성 표준을 제공합니다.

## 운영 수준 협약(OLA)

서비스 수준에 관한 계약(SLA)을 지원하기 위해 직무 IT 그룹이 서로에게 제공하기로 약속한 내용을 명확히 하는 계약입니다.

## 운영 준비 상태 검토(ORR)

인시던트 및 잠재적 장애의 범위를 이해, 평가 또는 예방하거나 줄이는 데 도움이 되는 질문 체크리스트 및 관련 모범 사례입니다. 자세한 내용은 AWS Well-Architected Framework의 [운영 준비 상태 검토\(ORR\)](#)를 참조하세요.

## 운영 기술(OT)

물리적 환경에서 작동하여 산업 운영, 장비 및 인프라를 제어하는 하드웨어 및 소프트웨어 시스템입니다. 제조 분야에서 OT 및 정보 기술(IT) 시스템의 통합은 [Industry 4.0](#) 트랜스포메이션의 주요 중점 사항입니다.

## 운영 통합(OI)

클라우드에서 운영을 현대화하는 프로세스로 준비 계획, 자동화 및 통합을 수반합니다. 자세한 내용은 [운영 통합 가이드](#)를 참조하십시오.

## 조직 트레일

조직 AWS 계정 내 모든에 대한 모든 이벤트를 로깅 AWS CloudTrail 하는에서 생성된 추적입니다 AWS Organizations. 이 트레일은 조직에 속한 각 AWS 계정에 생성되고 각 계정의 활동을 추적합니다. 자세한 내용은 CloudTrail 설명서의 [Creating a trail for an organization](#)을 참조하십시오.

## 조직 변경 관리(OCM)

사람, 문화 및 리더십 관점에서 중대하고 파괴적인 비즈니스 혁신을 관리하기 위한 프레임워크입니다. OCM은 변화 채택을 가속화하고, 과도기적 문제를 해결하고, 문화 및 조직적 변화를 주도함으로써 조직이 새로운 시스템 및 전략을 준비하고 전환할 수 있도록 지원합니다. AWS 마이그레이션 전략에서는 클라우드 채택 프로젝트에 필요한 변경 속도 때문에이 프레임워크를 인력 가속화라고 합니다. 자세한 내용은 [사용 가이드](#)를 참조하십시오.

## 오리진 액세스 제어(OAC)

CloudFront에서 Amazon Simple Storage Service(S3) 콘텐츠를 보호하기 위해 액세스를 제한하는 고급 옵션입니다. OAC는 AWS KMS (SSE-KMS)를 사용한 모든 서버 측 암호화 AWS 리전와 S3 버킷에 대한 동적 PUT 및 DELETE 요청에서 모든 S3 버킷을 지원합니다.

## 오리진 액세스 ID(OAI)

CloudFront에서 Amazon S3 콘텐츠를 보호하기 위해 액세스를 제한하는 옵션입니다. OAI를 사용하면 CloudFront는 Amazon S3가 인증할 수 있는 보안 주체를 생성합니다. 인증된 보안 주체는 특

정 CloudFront 배포를 통해서만 S3 버킷의 콘텐츠에 액세스할 수 있습니다. 더 세분화되고 향상된 액세스 제어를 제공하는 [OAC](#)도 참조하십시오.

## ORR

[운영 준비 상태 검토](#)를 참조하세요.

## OT

[운영 기술](#)을 참조하세요.

## 아웃바운드(송신) VPC

AWS 다중 계정 아키텍처에서 애플리케이션 내에서 시작된 네트워크 연결을 처리하는 VPC입니다. [AWS Security Reference Architecture](#)에서는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPC로 네트워크 계정을 설정할 것을 권장합니다.

## P

### 권한 경계

사용자나 역할이 가질 수 있는 최대 권한을 설정하기 위해 IAM 보안 주체에 연결되는 IAM 관리 정책입니다. 자세한 내용은 IAM 설명서의 [권한 경계](#)를 참조하십시오.

### 개인 식별 정보(PII)

직접 보거나 다른 관련 데이터와 함께 짝을 지을 때 개인의 신원을 합리적으로 추론하는 데 사용할 수 있는 정보입니다. PII의 예로는 이름, 주소, 연락처 정보 등이 있습니다.

## PII

[개인 식별 정보](#)를 참조하세요.

### 플레이북

클라우드에서 핵심 운영 기능을 제공하는 등 마이그레이션과 관련된 작업을 캡처하는 일련의 사전 정의된 단계입니다. 플레이북은 스크립트, 자동화된 런북 또는 현대화된 환경을 운영하는 데 필요한 프로세스나 단계 요약의 형태를 취할 수 있습니다.

## PLC

[프로그래밍 가능 로직 컨트롤러](#)를 참조하세요.

## PLM

[제품 수명 주기 관리](#)를 참조하세요.

### 정책

권한 정의([ID 기반 정책](#) 참조), 액세스 조건 지정([리소스 기반 정책](#) 참조), AWS Organizations 내 조직의 모든 계정에 대한 최대 권한 정의([서비스 제어 정책](#) 참조)와 같은 작업을 수행할 수 있는 객체입니다.

### 다국어 지속성

데이터 액세스 패턴 및 기타 요구 사항을 기반으로 독립적으로 마이크로서비스의 데이터 스토리지 기술 선택. 마이크로서비스가 동일한 데이터 스토리지 기술을 사용하는 경우 구현 문제가 발생하거나 성능이 저하될 수 있습니다. 요구 사항에 가장 적합한 데이터 저장소를 사용하면 마이크로서비스를 더 쉽게 구현하고 성능과 확장성을 높일 수 있습니다.

### 포트폴리오 평가

마이그레이션을 계획하기 위해 애플리케이션 포트폴리오를 검색 및 분석하고 우선순위를 정하는 프로세스입니다. 자세한 내용은 [마이그레이션 준비 상태 평가](#)를 참조하십시오.

### 조건자

보통 WHERE 절에 있는 true 또는 false를 반환하는 쿼리 조건입니다.

### 푸시다운 조건자

전송 전에 쿼리의 데이터를 필터링하는 데이터베이스 쿼리 최적화 기법입니다. 이렇게 하면 관계형 데이터베이스에서 검색하고 처리해야 하는 데이터의 양이 줄고 쿼리 성능이 향상됩니다.

### 예방적 제어

이벤트 발생을 방지하도록 설계된 보안 제어입니다. 이 제어는 네트워크에 대한 무단 액세스나 원치 않는 변경을 방지하는 데 도움이 되는 1차 방어선입니다. 자세한 내용은 Implementing security controls on AWS의 [Preventative controls](#)를 참조하십시오.

### 보안 주체

작업을 수행하고 리소스에 액세스할 수 있는 AWS 있는의 엔터티입니다. 이 엔터티는 일반적으로 , AWS 계정 IAM 역할 또는 사용자의 루트 사용자입니다. 자세한 내용은 IAM 설명서의 [역할 용어 및 개념](#)의 보안 주체를 참조하십시오.

### 개인 정보 보호 중심 설계

전체 개발 프로세스에서 개인 정보를 고려하는 시스템 엔지니어링에서의 접근 방식입니다.

## 프라이빗 호스팅 영역

Amazon Route 53에서 하나 이상의 VPC 내 도메인과 하위 도메인에 대한 DNS 쿼리에 응답하는 방법에 대한 정보가 담긴 컨테이너입니다. 자세한 내용은 Route 53 설명서의 [프라이빗 호스팅 영역 작업을 참조하십시오](#).

### 선제적 제어

규정 미준수 리소스의 배포를 방지하도록 설계된 [보안 제어](#)입니다. 이러한 제어는 리소스를 프로비저닝하기 전에 리소스를 스캔합니다. 리소스가 제어를 준수하지 않으면 프로비저닝되지 않습니다. 자세한 내용은 AWS Control Tower 설명서의 [제어 참조 가이드](#)를 참조하고 보안 [제어 구현의 사전 예방적 제어](#)를 참조하세요. AWS

### 제품 수명 주기 관리(PLM)

설계, 개발 및 출시부터 성장 및 성숙도를 거쳐 거부 및 제거에 이르기까지 전체 수명 주기 동안 제품의 데이터 및 프로세스 관리를 나타냅니다.

### 프로덕션 환경

[환경](#)을 참조하세요.

### 프로그래밍 가능 로직 컨트롤러(PLC)

제조 분야에서 기계를 모니터링하고 제조 프로세스를 자동화하는 매우 안정적이고 적응력이 뛰어난 컴퓨터입니다.

### 프롬프트 체이닝

한 [LLM](#) 프롬프트의 출력을 다음 프롬프트의 입력으로 사용하여 더 나은 응답을 생성합니다. 이 기법은 복잡한 작업을 하위 작업으로 나누거나 예비 응답을 반복적으로 세부 조정하거나 확장하는 데 사용됩니다. 이를 통해 모델 응답의 정확성과 관련성을 개선하고 보다 세분화되고 개인화된 결과를 얻을 수 있습니다.

### 가명화

데이터세트의 개인 식별자를 자리 표시자 값으로 바꾸는 프로세스입니다. 가명화는 개인 정보를 보호하는 데 도움이 될 수 있습니다. 가명화된 데이터는 여전히 개인 데이터로 간주됩니다.

### 게시/구독(pub/sub)

여러 마이크로서비스에서 비동기 통신을 지원하여 확장성과 응답성을 개선하는 패턴입니다. 예를 들어 마이크로서비스 기반 [MES](#)에서 마이크로서비스는 다른 마이크로서비스가 구독할 수 있는 채널에 이벤트 메시지를 게시할 수 있습니다. 시스템은 게시 서비스를 변경하지 않고도 새 마이크로서비스를 추가할 수 있습니다.

## Q

### 쿼리 계획

SQL 관계형 데이터베이스 시스템의 데이터에 액세스하는 데 사용되는 명령어와 같은 일련의 단계입니다.

### 쿼리 계획 회귀

데이터베이스 서비스 최적화 프로그램이 데이터베이스 환경을 변경하기 전보다 덜 최적의 계획을 선택하는 경우입니다. 통계, 제한 사항, 환경 설정, 쿼리 파라미터 바인딩 및 데이터베이스 엔진 업데이트의 변경으로 인해 발생할 수 있습니다.

## R

### RACI 매트릭스

[Responsible, Accountable, Consulted, Informed\(RACI\)](#)를 참조하세요.

### RAG

[검색 증강 생성](#)을 참조하세요.

### 랜섬웨어

결제가 완료될 때까지 컴퓨터 시스템이나 데이터에 대한 액세스를 차단하도록 설계된 악성 소프트웨어입니다.

### RASCI 매트릭스

[Responsible, Accountable, Consulted, Informed\(RACI\)](#)를 참조하세요.

### RCAC

[행 및 열 액세스 제어](#)를 참조하세요.

### 읽기 전용 복제본

읽기 전용 용도로 사용되는 데이터베이스의 사본입니다. 쿼리를 읽기 전용 복제본으로 라우팅하여 기본 데이터베이스의 로드를 줄일 수 있습니다.

### 리아키텍팅

[7R](#)을 참조하세요.

## Recovery Point Objective(RPO)

마지막 데이터 복구 시점 이후 허용되는 최대 시간입니다. 이에 따라 마지막 복구 시점과 서비스 중단 사이에 허용되는 데이터 손실로 간주되는 범위가 결정됩니다.

## Recovery Time Objective(RTO)

서비스 중단과 서비스 복원 사이의 허용 가능한 지연 시간입니다.

### 리팩터링

[7R](#)을 참조하세요.

### 리전

지리적 영역의 AWS 리소스 모음입니다. 각 AWS 리전은 내결함성, 안정성 및 복원력을 제공하기 위해 서로 격리되고 독립적입니다. 자세한 내용은 [계정에서 사용할 수 있는 AWS 리전 지정](#)을 참조하세요.

### 회귀

숫자 값을 예측하는 ML 기법입니다. 예를 들어, '이 집은 얼마에 팔릴까?'라는 문제를 풀기 위해 ML 모델은 선형 회귀 모델을 사용하여 주택에 대해 알려진 사실(예: 면적)을 기반으로 주택의 매매 가격을 예측할 수 있습니다.

### 리호스팅

[7R](#)을 참조하세요.

### 릴리스

배포 프로세스에서 변경 사항을 프로덕션 환경으로 승격시키는 행위입니다.

### 재배치

[7R](#)을 참조하세요.

### 리플랫폼

[7R](#)을 참조하세요.

### 재구매

[7R](#)을 참조하세요.

### 복원력

중단에 저항하거나 중단을 복구할 수 있는 애플리케이션의 기능입니다. [고가용성](#) 및 [재해 복구](#)는 AWS 클라우드에서 복원력을 계획할 때 일반적인 고려 사항입니다. 자세한 내용은 [AWS 클라우드 복원력](#)을 참조하세요.

## 리소스 기반 정책

Amazon S3 버킷, 엔드포인트, 암호화 키 등의 리소스에 연결된 정책입니다. 이 유형의 정책은 액세스가 허용된 보안 주체, 지원되는 작업 및 충족해야 하는 기타 조건을 지정합니다.

## RACI(Responsible, Accountable, Consulted, Informed) 매트릭스

마이그레이션 활동 및 클라우드 운영에 참여하는 모든 당사자의 역할과 책임을 정의하는 매트릭스입니다. 매트릭스 이름은 매트릭스에 정의된 책임 유형에서 파생됩니다. 실무 담당자 (R), 의사 결정권자 (A), 업무 수행 조언자 (C), 결과 통보 대상자 (I). 지원자는 (S) 선택사항입니다. 지원자를 포함하면 매트릭스를 RASCI 매트릭스라고 하고, 지원자를 제외하면 RACI 매트릭스라고 합니다.

## 대응 제어

보안 기준에서 벗어나거나 부정적인 이벤트를 해결하도록 설계된 보안 제어입니다. 자세한 내용은 AWS에서 보안 제어 구현의 [대응 제어](#)를 참조하세요.

## retain

[7R](#)을 참조하세요.

## 사용 중지

[7R](#)을 참조하세요.

## 검색 증강 세대(RAG)

응답을 생성하기 전에 [LLM](#)이 훈련 데이터 소스 외부에 있는 신뢰할 수 있는 데이터 소스를 참조하는 [생성형 AI](#) 기술입니다. 예를 들어 RAG 모델은 조직의 지식 기반 또는 사용자 지정 데이터에 대한 시맨틱 검색을 수행할 수 있습니다. 자세한 내용은 [검색 증강 생성\(RAG\)이란 무엇인가요?](#)를 참조하세요.

## 교체

공격자가 자격 증명에 액세스하는 것을 더욱 어렵게 만들기 위해 [보안 암호](#)를 주기적으로 업데이트 하는 프로세스입니다.

## 행 및 열 액세스 제어(RCAC)

액세스 규칙이 정의된 기본적이고 유연한 SQL 표현식을 사용합니다. RCAC는 행 권한과 열 마스크로 구성됩니다.

## RPO

[목표 복구 시점\(RPO\)](#)을 참조하세요.

## RTO

[목표 복구 시간\(RTO\)](#)을 참조하세요.

### 런북

특정 작업을 수행하는 데 필요한 일련의 수동 또는 자동 절차입니다. 일반적으로 오류율이 높은 반복 작업이나 절차를 간소화하기 위해 런북을 만듭니다.

## S

### SAML 2.0

많은 ID 제공업체(idP)에서 사용하는 개방형 표준입니다. 이 기능을 사용하면 연동 SSO(Single Sign-On)를 AWS Management Console 사용할 수 있으므로 사용자는 조직의 모든 사용자에게 대해 IAM에서 사용자를 생성하지 않고도 로그인하거나 AWS API 작업을 호출할 수 있습니다. SAML 2.0 기반 페더레이션에 대한 자세한 내용은 IAM 설명서의 [SAML 2.0 기반 페더레이션 정보](#)를 참조하십시오.

### SCADA

[감독 제어 및 데이터 획득](#)을 참조하세요.

### SCP

[서비스 제어 정책](#)을 참조하세요.

### 보안 암호

에는 암호 또는 사용자 자격 증명과 같이 암호화된 형식으로 저장하는 AWS Secrets Manager 키 또는 제한된 정보가 있습니다. 보안 암호 값과 메타데이터로 구성됩니다. 보안 암호 값은 바이너리, 단일 문자열 또는 여러 문자열일 수 있습니다. 자세한 내용은 AWS Secrets Manager 설명서의 [Secrets Manager 보안 암호란 무엇인가요?](#)를 참조하세요.

### 보안 중심 설계

전체 개발 프로세스에서 보안을 고려하는 시스템 엔지니어링에서의 접근 방식입니다.

### 보안 제어

위험 행위자가 보안 취약성을 악용하는 능력을 방지, 탐지 또는 감소시키는 기술적 또는 관리적 가드레일입니다. 보안 제어는 [예방](#), [감지](#), [대응](#), [선제적](#)과 같은 기본적인 네 가지 보안 제어 유형으로 구분됩니다.

## 보안 강화

공격 표면을 줄여 공격에 대한 저항력을 높이는 프로세스입니다. 더 이상 필요하지 않은 리소스 제거, 최소 권한 부여의 보안 모범 사례 구현, 구성 파일의 불필요한 기능 비활성화 등의 작업이 여기에 포함될 수 있습니다.

### 보안 정보 및 이벤트 관리(SIEM) 시스템

보안 정보 관리(SIM)와 보안 이벤트 관리(SEM) 시스템을 결합하는 도구 및 서비스입니다. SIEM 시스템은 서버, 네트워크, 디바이스 및 기타 소스에서 데이터를 수집, 모니터링 및 분석하여 위협과 보안 침해를 탐지하고 알림을 생성합니다.

### 보안 응답 자동화

보안 이벤트에 자동으로 응답하거나 이를 해결하도록 설계된 사전 정의되고 프로그래밍된 작업입니다. 이러한 자동화는 보안 모범 사례를 구현하는 데 도움이 되는 [탐지](#) 또는 [대응](#) AWS 보안 제어 역할을 합니다. 자동화된 응답 작업의 예로 VPC 보안 그룹 수정, Amazon EC2 인스턴스 패치 적용 또는 자격 증명 교체 등이 있습니다.

### 서버 측 암호화

대상에서 데이터를 수신하는 AWS 서비스에 의한 데이터 암호화.

### 서비스 제어 정책(SCP)

AWS Organizations에 속한 조직의 모든 계정에 대한 권한을 중앙 집중식으로 제어하는 정책입니다. SCP는 관리자가 사용자 또는 역할에 위임할 수 있는 작업에 대해 제한을 설정하거나 가드레일을 정의합니다. SCP를 허용 목록 또는 거부 목록으로 사용하여 허용하거나 금지할 서비스 또는 작업을 지정할 수 있습니다. 자세한 내용은 AWS Organizations 설명서의 [서비스 제어 정책을](#) 참조하세요.

### 서비스 엔드포인트

에 대한 진입점의 URL입니다 AWS 서비스. 엔드포인트를 사용하여 대상 서비스에 프로그래밍 방식으로 연결할 수 있습니다. 자세한 내용은 AWS 일반 참조의 [AWS 서비스 엔드포인트](#)를 참조하십시오.

### 서비스 수준에 관한 계약(SLA)

IT 팀이 고객에게 제공하기로 약속한 내용(예: 서비스 가동 시간 및 성능)을 명시한 계약입니다.

### 서비스 수준 지표(SLI)

오류 발생률, 가용성 또는 처리량과 같은 서비스의 성능 측면에 대한 측정값입니다.

## 서비스 수준 목표(SLO)

[서비스 수준 지표](#)로 측정되는 서비스의 상태를 나타내는 목표 지표입니다.

### 공동 책임 모델

클라우드 보안 및 규정 준수를 AWS 위해와 공유하는 책임을 설명하는 모델입니다. AWS 는 클라우드의 보안을 담당하는 반면, 사용자는 클라우드의 보안을 담당합니다. 자세한 내용은 [공동 책임 모델](#)을 참조하십시오.

### SIEM

[보안 정보 및 이벤트 관리 시스템](#)을 참조하세요.

### 단일 장애점(SPOF)

애플리케이션을 중단시킬 수 있는 애플리케이션의 중요한 단일 구성 요소에서 발생하는 장애입니다.

### SLA

[서비스 수준 계약](#)을 참조하세요.

### SLI

[서비스 수준 지표](#)를 참조하세요.

### SLO

[서비스 수준 목표](#)를 참조하세요.

### 분할 앤 시드 모델

현대화 프로젝트를 확장하고 가속화하기 위한 패턴입니다. 새로운 기능과 제품 릴리스가 정의되면 핵심 팀이 분할되어 새로운 제품 팀이 만들어집니다. 이를 통해 조직의 역량과 서비스 규모를 조정하고, 개발자 생산성을 개선하고, 신속한 혁신을 지원할 수 있습니다. 자세한 내용은 [AWS 클라우드에서 애플리케이션을 현대화하기 위한 단계별 접근 방식](#)을 참조하세요.

### SPOF

[단일 장애점](#)을 참조하세요.

### 스타 스키마

하나의 큰 팩트 테이블을 사용하여 트랜잭션 또는 측정된 데이터를 저장하고 하나 이상의 더 작은 차원 테이블을 사용하여 데이터 속성을 저장하는 데이터베이스 조직 구조입니다. 이 구조는 [데이터 웨어하우스](#)에서 또는 비즈니스 인텔리전스 목적으로 사용하도록 설계되었습니다.

## Strangler Fig 패턴

레거시 시스템을 폐기할 수 있을 때까지 시스템 기능을 점진적으로 다시 작성하고 교체하여 모놀리식 시스템을 현대화하기 위한 접근 방식. 이 패턴은 무화과 덩굴이 나무로 자라 결국 숙주를 압도하고 대체하는 것과 비슷합니다. [Martin Fowler](#)가 모놀리식 시스템을 다시 작성할 때 위험을 관리하는 방법으로 이 패턴을 도입했습니다. 이 패턴을 적용하는 방법의 예는 [컨테이너 및 Amazon API Gateway를 사용하여 기존의 Microsoft ASP.NET\(ASMX\) 웹 서비스를 점진적으로 현대화하는 방법](#)을 참조하십시오.

### 서브넷

VPC의 IP 주소 범위입니다. 서브넷은 단일 가용 영역에 상주해야 합니다.

### 감독 제어 및 데이터 획득(SCADA)

제조 분야에서 하드웨어와 소프트웨어를 사용하여 물리적 자산과 프로덕션 작업을 모니터링하는 시스템입니다.

### 대칭 암호화

동일한 키를 사용하여 데이터를 암호화하고 복호화하는 암호화 알고리즘입니다.

### 합성 테스트

사용자 상호 작용을 시뮬레이션하여 잠재적 문제를 감지하거나 성능을 모니터링하는 방식으로 진행되는 시스템 테스트입니다. [Amazon CloudWatch Synthetics](#)를 사용하여 이러한 테스트를 생성할 수 있습니다.

### 시스템 프롬프트

[LLM](#)에 컨텍스트, 명령 또는 지침을 제공하여 동작을 지시하는 기법입니다. 시스템 프롬프트는 컨텍스트를 설정하고 사용자와의 상호 작용을 위한 규칙을 설정하는 데 도움이 됩니다.

## T

### tags

AWS 리소스를 구성하기 위한 메타데이터 역할을 하는 키-값 페어입니다. 태그를 사용하면 리소스를 손쉽게 관리, 식별, 정리, 검색, 필터링할 수 있습니다. 자세한 내용은 [AWS 리소스에 태그 지정](#)을 참조하십시오.

### 대상 변수

지도 ML에서 예측하려는 값으로, 결과 변수라고도 합니다. 예를 들어, 제조 설정에서 대상 변수는 제품 결함일 수 있습니다.

## 작업 목록

런북을 통해 진행 상황을 추적하는 데 사용되는 도구입니다. 작업 목록에는 런북의 개요와 완료해야 할 일반 작업 목록이 포함되어 있습니다. 각 일반 작업에 대한 예상 소요 시간, 소유자 및 진행 상황이 작업 목록에 포함됩니다.

## 테스트 환경

[환경](#)을 참조하세요.

## 훈련

ML 모델이 학습할 수 있는 데이터를 제공하는 것입니다. 훈련 데이터에는 정답이 포함되어야 합니다. 학습 알고리즘은 훈련 데이터에서 대상(예측하려는 답)에 입력 데이터 속성을 매핑하는 패턴을 찾고, 이러한 패턴을 캡처하는 ML 모델을 출력합니다. 그런 다음 ML 모델을 사용하여 대상을 모르는 새 데이터에 대한 예측을 할 수 있습니다.

## Transit Gateway

VPC와 온프레미스 네트워크를 상호 연결하는 데 사용할 수 있는 네트워크 전송 허브입니다. 자세한 내용은 AWS Transit Gateway 설명서의 [전송 게이트웨이란 무엇입니까?](#)를 참조하세요.

## 트렁크 기반 워크플로

개발자가 기능 브랜치에서 로컬로 기능을 구축하고 테스트한 다음 해당 변경 사항을 기본 브랜치에 병합하는 접근 방식입니다. 이후 기본 브랜치는 개발, 프로덕션 이전 및 프로덕션 환경에 순차적으로 구축됩니다.

## 신뢰할 수 있는 액세스

사용자를 대신하여 AWS Organizations 및 해당 계정에서 조직에서 작업을 수행하도록 지정하는 서비스에 대한 권한 부여. 신뢰할 수 있는 서비스는 필요할 때 각 계정에 서비스 연결 역할을 생성하여 관리 작업을 수행합니다. 자세한 내용은 설명서의 [다른 AWS 서비스와 AWS Organizations 함께 사용](#)을 참조하세요 AWS Organizations .

## 튜닝

ML 모델의 정확도를 높이기 위해 훈련 프로세스의 측면을 여러 변경하는 것입니다. 예를 들어, 레이블링 세트를 생성하고 레이블을 추가한 다음 다양한 설정에서 이러한 단계를 여러 번 반복하여 모델을 최적화하는 방식으로 ML 모델을 훈련할 수 있습니다.

## 피자 두 판 팀

피자 두 판이면 충분한 소규모 DevOps 팀. 피자 두 판 팀 규모는 소프트웨어 개발에 있어 가능한 최상의 공동 작업 기회를 보장합니다.

## U

### 불확실성

예측 ML 모델의 신뢰성을 저해할 수 있는 부정확하거나 불완전하거나 알려지지 않은 정보를 나타내는 개념입니다. 불확실성에는 두 가지 유형이 있습니다. 인식론적 불확실성은 제한적이고 불완전한 데이터에 의해 발생하는 반면, 우연한 불확실성은 데이터에 내재된 노이즈와 무작위성에 의해 발생합니다. 자세한 내용은 [Quantifying uncertainty in deep learning systems](#) 가이드를 참조하십시오.

### 차별화되지 않은 작업

애플리케이션을 만들고 운영하는 데 필요하지만 최종 사용자에게 직접적인 가치를 제공하거나 경쟁 우위를 제공하지 못하는 작업을 헤비 리프팅이라고도 합니다. 차별화되지 않은 작업의 예로는 조달, 유지보수, 용량 계획 등이 있습니다.

### 상위 환경

[환경](#)을 참조하세요.

## V

### 정리

스토리지를 회수하고 성능을 향상시키기 위해 증분 업데이트 후 정리 작업을 수행하는 데이터베이스 유지 관리 작업입니다.

### 버전 제어

리포지토리의 소스 코드 변경과 같은 변경 사항을 추적하는 프로세스 및 도구입니다.

### VPC 피어링

프라이빗 IP 주소를 사용하여 트래픽을 라우팅할 수 있게 하는 두 VPC 간의 연결입니다. 자세한 내용은 Amazon VPC 설명서의 [VPC 피어링이란?](#)을 참조하십시오.

### 취약성

시스템 보안을 손상시키는 소프트웨어 또는 하드웨어 결함입니다.

# W

## 웜 캐시

자주 액세스하는 최신 관련 데이터를 포함하는 버퍼 캐시입니다. 버퍼 캐시에서 데이터베이스 인스턴스를 읽을 수 있기 때문에 주 메모리나 디스크에서 읽는 것보다 빠릅니다.

## 웜 데이터

자주 액세스하지 않는 데이터입니다. 이런 종류의 데이터를 쿼리할 때는 일반적으로 적절히 느린 쿼리가 허용됩니다.

## 창 함수

현재 레코드와 어떤 식으로든 관련된 행 그룹에서 계산을 수행하는 SQL 함수입니다. 창 함수는 이동 평균을 계산하거나 현재 행의 상대적 위치를 기반으로 행 값에 액세스하는 등의 태스크를 처리하는 데 유용합니다.

## 워크로드

고객 대면 애플리케이션이나 백엔드 프로세스 같이 비즈니스 가치를 창출하는 리소스 및 코드 모음입니다.

## 워크스트림

마이그레이션 프로젝트에서 특정 작업 세트를 담당하는 직무 그룹입니다. 각 워크스트림은 독립적이지만 프로젝트의 다른 워크스트림을 지원합니다. 예를 들어, 포트폴리오 워크스트림은 애플리케이션 우선순위 지정, 웨이브 계획, 마이그레이션 메타데이터 수집을 담당합니다. 포트폴리오 워크스트림은 이러한 자산을 마이그레이션 워크스트림에 전달하고, 마이그레이션 워크스트림은 서버와 애플리케이션을 마이그레이션합니다.

## WORM

[Write Once, Read Many\(WORM\)](#)를 참조하세요.

## WQF

[AWS Workload Qualification Framework](#)를 참조하세요.

## Write Once Read Many(WORM)

데이터를 한 번 쓰고 데이터가 삭제되거나 수정되지 않도록 하는 스토리지 모델입니다. 권한 있는 사용자는 필요한 만큼 여러 번 데이터를 읽을 수 있지만 데이터를 변경할 수는 없습니다. 이 데이터 스토리지 인프라는 [변경 불가능](#)한 항목으로 간주됩니다.

## Z

### 제로데이 익스플로잇

[제로데이 취약성](#)을 악용하는 공격(일반적으로 맬웨어)입니다.

### 제로데이 취약성

프로덕션 시스템의 명백한 결함 또는 취약성입니다. 위협 행위자는 이러한 유형의 취약성을 사용하여 시스템을 공격할 수 있습니다. 개발자는 공격의 결과로 취약성을 인지하는 경우가 많습니다.

### 제로샷 프롬프팅

태스크를 수행하기 위해 [LLM](#)에 명령을 제공하지만 안내에 도움이 되는 예제(샷)는 제공하지 않습니다. LLM은 사전 훈련된 지식을 사용하여 태스크를 처리해야 합니다. 제로샷 프롬프팅의 효과는 태스크의 복잡성과 프롬프트의 품질에 따라 달라집니다. [퓨샷 프롬프팅](#)도 참조하세요.

### 좀비 애플리케이션

평균 CPU 및 메모리 사용량이 5% 미만인 애플리케이션입니다. 마이그레이션 프로젝트에서는 이러한 애플리케이션을 사용 중지하는 것이 일반적입니다.

---

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.