



Microsoft SQL Server 데이터베이스를 클라우드로 마이그레이션하기 AWS

AWS 규범적 지침



AWS 규범적 지침: Microsoft SQL Server 데이터베이스를 클라우드로 마이그레이션하기 AWS

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

의 상표 및 브랜드 디자인은 외 제품 또는 서비스와 함께, 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon 계열사, 관련 업체 또는 Amazon의 지원 업체 여부에 상관없이 해당 소유자의 자산입니다.

Table of Contents

소개	1
개요	1
마이그레이션 전략	4
적절한 마이그레이션 전략 선택	5
온라인 및 오프라인 마이그레이션	6
마이그레이션 방법	7
네이티브 SQL 서버 백업/복원	13
로그 전달	15
데이터베이스 미러링	16
Always On 가용성 그룹	16
분산 가용성 그룹	17
트랜잭션 복제	19
AWS Migration Hub Orchestrator	20
AWS Snowball 엣지	22
동종 데이터베이스 마이그레이션	23
Amazon RDS for SQL Server	24
Amazon RDS를 선택해야 하는 시기	24
높은 가용성	25
읽기 전용 복제본	27
재해 복구	29
SQL Server용 Amazon RDS Custom	29
SQL Server용 Amazon RDS Custom를 선택해야 하는 경우	29
작동 방식	30
Amazon EC2 for SQL Server	33
Amazon EC2를 선택해야 하는 경우	33
높은 가용성	34
재해 복구	41
SQL Server용 VMware 클라우드 온 AWS	42
VMware 클라우드 온을 선택해야 하는 경우 AWS	42
이기종 데이터베이스 마이그레이션	44
도구	46
AWS SCT	46
AWS DMS	47
Babelfish	47

하이브리드 마이그레이션 시나리오	50
데이터베이스를 클라우드에 백업	50
고가용성 및 재해 복구 솔루션 확장	50
Storage Gateway	51
AWS DMS 및 AWS SCT 사용	52
SQL Server 데이터베이스 현대화	53
SQL Server 워크로드를 Windows에서 Linux로 마이그레이션	53
Linux의 고가용성	54
AWS Launch Wizard	55
Amazon RDS for SQL Server로 마이그레이션하기 위한 모범 사례	59
대상 데이터베이스 프로비저닝	59
소스 데이터베이스에서 백업	60
AWS에 데이터 덤프 파일 전송	60
대상 데이터베이스로 데이터 복원	60
마이그레이션 후 단계	60
마이그레이션 테스트	61
Amazon RDS 데이터베이스 운영 및 최적화	61
Amazon EC2와 Amazon RDS 중 선택	63
결정 매트릭스	63
공동 책임	86
SQL Server 데이터베이스 마이그레이션 패턴	88
파트너	89
추가 리소스	90
감사의 말	91
부록: SQL Server 데이터베이스 마이그레이션 설문	92
일반 정보	92
인프라	92
데이터베이스 백업	93
데이터베이스 기능	93
데이터베이스 보안	93
데이터베이스 고가용성 및 재해 복구	93
문서 기록	95
용어집	97
#	97
A	98
B	100

C	102
D	104
E	108
F	110
G	111
H	112
I	113
L	115
M	116
O	119
P	121
Q	123
R	123
S	126
T	129
U	130
V	131
W	131
Z	132
.....	cxxxiii

Microsoft SQL Server 데이터베이스를 클라우드로 마이그레이션하기 AWS

Sagar Patel, Amazon Web Services (AWS)

2024년 4월 ([문서 기록](#))

Amazon Web Services (AWS) 는 Microsoft SQL Server 데이터베이스를 안정적이고 안전한 AWS 클라우드 인프라에 배포하기 위한 포괄적인 서비스 및 도구 세트를 제공합니다. SQL Server를 기반으로 실행할 때의 AWS 이점으로는 비용 절감, 확장성, 고가용성 및 재해 복구, 성능 향상, 관리 용이성 등이 있습니다. 자세한 내용은 AWS Compute [블로그에서 Microsoft Windows Server 및 SQL Server 워크로드를 실행하기에 가장 적합한 클라우드인 이유 AWS 알아보기](#)를 참조하십시오.

이 가이드에서는 SQL Server 데이터베이스를 온프레미스에서 AWS 클라우드, Amazon RDS (관계형 데이터베이스 서비스), Amazon Elastic Compute Cloud (Amazon EC2) 또는 VMware Cloud on으로 마이그레이션하는 데 사용할 수 있는 옵션에 대해 설명합니다. AWS이 문서에서는 이러한 마이그레이션 옵션 사용에 대한 모범 사례 및 권장 사항을 자세히 설명합니다. 또한 온프레미스 SQL Server 환경과 로그 전달, 복제 AWS, Always On 가용성 그룹과 같은 기본 SQL Server 기능을 사용하여 고가용성 및 재해 복구 솔루션을 설정하는 방법에 대한 정보도 제공합니다.

이 가이드는 온프레미스 SQL Server 데이터베이스를 AWS(으)로 마이그레이션하려는 프로그램 또는 프로젝트 관리자, 제품 소유자, 데이터베이스 관리자, 데이터베이스 엔지니어, 운영 또는 인프라 관리자를 위한 것입니다.

개요

SQL Server 데이터베이스를 마이그레이션하기 전에 [관계형 데이터베이스의 마이그레이션 전략에 설명된 프레임워크를 사용하여 마이그레이션 전략을](#) 이해하고 평가해야 합니다. AWS

첫 번째 단계는 마이그레이션의 복잡성, 호환성 및 비용을 파악하여 응용 프로그램 및 SQL Server 데이터베이스 워크로드를 분석하는 것입니다. 마이그레이션을 계획할 때 고려해야 할 몇 가지 주요 사항은 다음과 같습니다.

- 데이터베이스 크기 - 데이터베이스의 현재 크기와 전체 용량 증가를 확인하십시오. 예를 들어 SQL Server 데이터베이스를 Amazon RDS 또는 Amazon RDS Custom 으로 마이그레이션하려는 경우 최대 16TiB의 스토리지가 포함된 DB 인스턴스를 생성할 수 있습니다. Support에서 [AWS 지원 티켓을 열어](#) 추가 스토리지를 요청할 수 있습니다. 최신 정보는 Amazon RDS 설명서의 [Amazon RDS DB 인스턴스 스토리지](#)를 참조하십시오.

- IOPS - 데이터베이스의 IOPS와 처리량을 결정합니다. Amazon RDS로 마이그레이션할 계획이라면 Amazon RDS DB [인스턴스의 I/O 성능을](#) 고려해 보십시오.
- 종속성 - 현재 데이터베이스 종속성을 확인하십시오. 데이터베이스가 다른 데이터베이스에 종속되어 있는 경우 기본 데이터베이스를 마이그레이션한 후 함께 마이그레이션하거나 종속성을 만들 수 있습니다.

데이터베이스가 레거시, 사용자 지정 또는 패키지 애플리케이션을 지원하는 경우 Amazon RDS Custom for SQL Server 작업이 좋은 선택일 수 있습니다. 이 서비스를 사용하면 데이터베이스 구성, 공유 파일 시스템 및 운영 체제 패치에 대한 제어를 유지할 수 있습니다.

모든 SQL Server 종속성을 인벤토리로 작성하십시오. 어떤 웹 서버 (예: 보고 서버 또는 비즈니스 인텔리전스 서버)가 SQL Server와 인터페이스하는지 알아보십시오. 마이그레이션할 시기가 되면 이 정보를 통해 영향을 받을 대상과 영향을 최소화할 수 있는 방법을 파악할 수 있습니다.

- 규정 준수 - Amazon RDS 또는 Amazon EC2로 이전한 후 현재 아키텍처와 감사 또는 규정 준수 요구 사항을 검토하여 이러한 요구 사항을 충족할 수 있는지 확인하십시오.
- HA/DR - 고가용성 (HA) 및 자동 장애 조치 기능이 필요하십니까? 프로덕션 워크로드를 실행하는 경우 고가용성 및 재해 복구 (DR)가 권장되는 모범 사례입니다.

HA/DR 요구 사항을 이해하여 다중 지역 아키텍처가 필요한지 여부를 결정하십시오. 그렇다면 SQL Server 데이터베이스를 Amazon EC2로 마이그레이션하십시오. Amazon RDS는 다중 지역 구성을 지원하지 않습니다.

- 버전 지원 - Amazon RDS for SQL Server로 전환하려는 경우 SQL Server 소프트웨어의 버전 및 에디션을 확인하세요([Amazon RDS](#) 및 [Amazon RDS](#)에서 현재 지원되는 버전 참조).
- 네트워크 연결 — 온-프레미스 환경과 환경 간의 네트워크 연결을 확인하여 온프레미스와 AWS환경 간에 데이터를 빠르게 전송할 수 있는 충분한 대역폭을 제공하는지 확인하십시오. AWS
- 마이그레이션 다운타임 - 마이그레이션 접근 방식을 계획하고 온라인 또는 오프라인 마이그레이션을 사용할지 결정할 수 있도록 마이그레이션에 사용할 수 있는 다운타임을 결정하십시오.
- RTO, RPO, SLA 요구 사항 - 기존 Database Work에 대한 목표 복구 시간 (RTO), 목표 복구 시점 (RPO), 목표 복구 시점 (RPO), 목표 복구 시점 (RPO), 목표 복구 시점 (RPO) 및 SLA (서비스 수준 계약) 요건을 파악합니다.
- 라이선스 - 라이선스 옵션을 이해하십시오. Amazon EC2 및 Amazon RDS에서 라이선스가 포함된 옵션을 선택하거나 Amazon EC2에서 [자체 라이선스 \(BYOL\)](#)를 사용할 수 있습니다.
- 기능 지원 — 애플리케이션이 사내에서 개발했던 COTS commercial-off-the-shelf (소프트웨어)에서 개발했던 상관없이 애플리케이션이 사용하는 데이터베이스 기능을 식별합니다. 이 정보는 SQL Server Enterprise 에디션에서 스탠다드 에디션으로 전환하여 라이선스 비용을 절감할 수 있는지 여

부를 판단하는 데 도움이 될 수 있습니다. 하지만 전환하기 전에 스탠다드 에디션 리소스 제한을 검토하십시오. 예를 들어 스탠다드 에디션은 128GB RAM만 지원합니다.

워크로드가 Amazon RDS for SQL Server에서 제공하는 특징 및 기능에 적합합니까? 자세한 내용은 [Amazon RDS의 SQL Server 기능](#)을 참조하십시오. 지원되지 않는 기능이 필요한 경우 Amazon EC2로 마이그레이션하는 것이 좋습니다.

SQL Server 데이터베이스 마이그레이션 전략

높은 수준에서 보면 SQL Server 데이터베이스를 온프레미스에서 AWS 클라우드로 마이그레이션하는 두 가지 옵션이 있습니다. 하나는 SQL Server를 계속 사용하는 것([동종 마이그레이션](#))이고 다른 하나는 SQL Server를 사용하지 않는 것([이기종 마이그레이션](#))입니다. 동종 마이그레이션에서는 데이터베이스 엔진을 변경하지 않습니다. 즉, 대상 데이터베이스는 SQL Server 데이터베이스이기도 합니다. 이기종 마이그레이션에서는 SQL Server 데이터베이스를 MySQL, PostgreSQL 또는 MariaDB와 같은 오픈 소스 데이터베이스 엔진이나 Amazon Aurora, Amazon DynamoDB 또는 Amazon Redshift와 같은 AWS 클라우드 네이티브 데이터베이스로 전환합니다.

SQL Server 데이터베이스를 AWS(으)로 마이그레이션하는 데는 리호스팅, 리플랫폼, 리아키텍트(리팩터링) 등 세 가지 일반적인 전략이 있습니다. 이는 [애플리케이션 마이그레이션 전략의 7R 중 일부](#)이며 다음 표에 설명되어 있습니다.

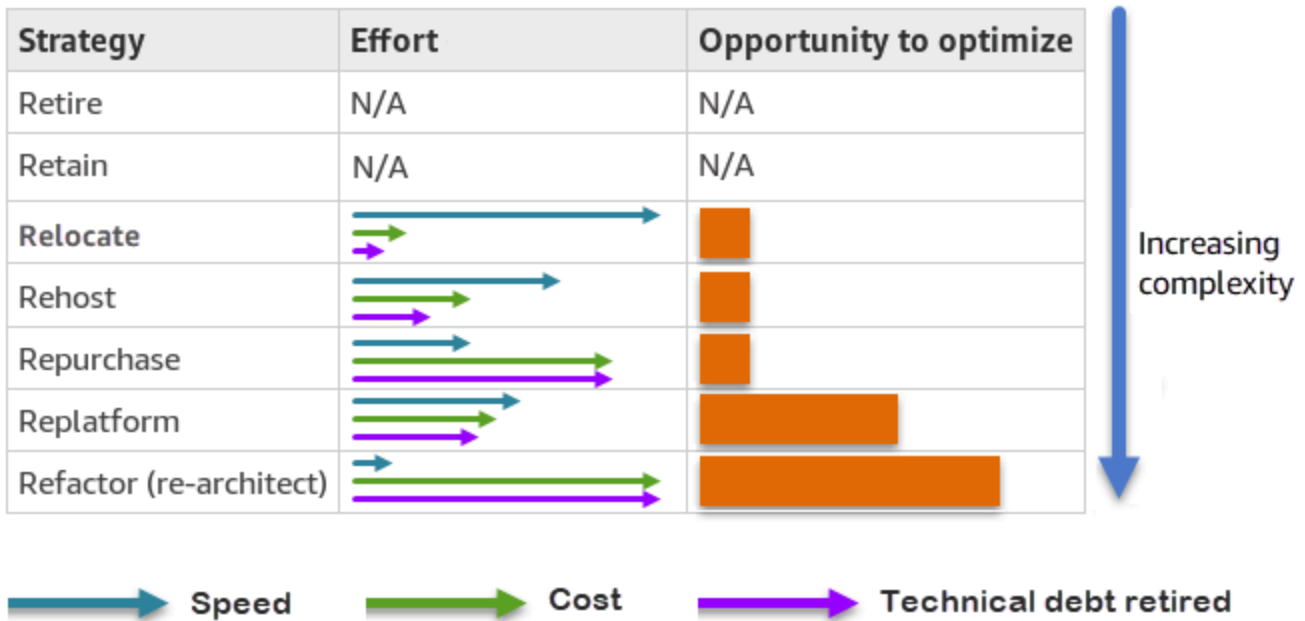
Strategy	유형	선택 시기	예
리호스팅	동종	운영 체제, 데이터베이스 소프트웨어 또는 구성을 변경하거나 변경하지 않고 SQL Server 데이터베이스를 있는 그대로 마이그레이션하려고 합니다.	SQL 서버에서 Amazon EC2로 (리호스트 패턴 찾아보기)
리플랫폼	동종	완전 관리형 데이터베이스 제품을 사용하여 데이터베이스 인스턴스 관리에 소요되는 시간을 줄이고 싶습니다.	SQL Server for SQL Server, Amazon RDS for SQL Server (리플랫폼 패턴 찾아보기)
리아키텍트(리팩터링)	이기종	오픈 소스 및 클라우드 네이티브 데이터베이스 기능을 활용하기 위해 데이터베이스와 애플리케이션을 재구성,	SQL 서버에서 Amazon Aurora PostgreSQL, MySQL 또는 MariaDB로

Strategy	유형	선택 시기	예
		재작성 및 재설계하려고 합니다.	리아키텍처 패턴 찾아보기 (리아키텍처 패턴)

SQL Server 데이터베이스를 리호스팅할지 또는 리플랫폼할지 결정하려는 경우 이 안내서의 뒷부분에 나오는 [Amazon EC2와 Amazon RDS 중 하나를 선택하여](#) 지원되는 기능을 나란히 비교하세요.

적절한 마이그레이션 전략 선택

올바른 전략을 선택하는 것은 비즈니스 요구 사항, 리소스 제약, 마이그레이션 기간 및 비용 고려 사항에 따라 달라집니다. 다음 다이어그램은 7가지 전략을 모두 포함하여 마이그레이션과 관련된 노력과 복잡성을 보여줍니다.



SQL Server 데이터베이스를 리팩터링하고 Amazon Aurora PostgreSQL 호환 에디션 또는 Aurora MySQL 호환 에디션과 같은 오픈 소스 또는 AWS 클라우드 네이티브 데이터베이스로 마이그레이션하면 데이터베이스를 현대화하고 최적화하는 데 도움이 될 수 있습니다. 오픈 소스 데이터베이스로 전환하면 값비싼 라이선스 (비용 절감), 공급업체 종속 기간 및 감사를 피할 수 있습니다. 그러나 워크로드의 복잡성에 따라 SQL Server 데이터베이스를 리팩터링하는 작업은 복잡하고 시간이 많이 걸리며 리소스 집약적인 작업이 될 수 있습니다.

복잡성을 줄이려면 데이터베이스를 한 번에 마이그레이션하는 대신 단계적 접근 방식을 고려해 볼 수 있습니다. 첫 번째 단계에서는 핵심 데이터베이스 기능에 집중할 수 있습니다. 다음 단계에서는 추가 AWS 서비스를 클라우드 환경에 통합하여 비용을 절감하고 성능, 생산성 및 규정 준수를 최적화할 수 있습니다. 예를 들어 온프레미스 SQL Server 데이터베이스를 Aurora MySQL과 호환되는 데이터베이스로 교체하는 것이 목표라면, 첫 번째 단계에서는 데이터베이스를 Amazon EC2에 리호스팅하거나 Amazon RDS for SQL Server에서 데이터베이스를 재구축하고, 이후 단계에서는 Aurora MySQL과 호환되도록 리팩터링하는 것을 고려할 수 있습니다. 이 접근 방식은 마이그레이션 단계에서 비용, 리소스 및 위험을 줄이는 데 도움이 되며 두 번째 단계에서는 최적화 및 현대화에 중점을 둡니다.

온라인 및 오프라인 마이그레이션

마이그레이션 일정과 허용할 수 있는 가동 중지 시간에 따라 두 가지 방법, 즉 오프라인 마이그레이션 또는 온라인 마이그레이션을 사용하여 온-프레미스 또는 다른 AWS 클라우드 환경에서 클라우드로 SQL Server 데이터베이스를 마이그레이션할 수 있습니다.

- 오프라인 마이그레이션: 이 방법은 애플리케이션이 계획된 다운타임을 감당할 수 있을 때 사용됩니다. 오프라인 마이그레이션에서는 마이그레이션 기간 동안 소스 데이터베이스가 오프라인 상태가 됩니다. 소스 데이터베이스가 오프라인 상태인 동안에는 AWS에 있는 대상 데이터베이스로 마이그레이션됩니다. 마이그레이션이 완료되면 소스 데이터베이스와의 데이터 일관성을 보장하기 위해 검증 및 확인 검사가 수행됩니다. 데이터베이스가 모든 유효성 검사를 통과하면 AWS에서 애플리케이션을 대상 데이터베이스에 연결하여 AWS(으)로 전환을 수행합니다.
- 온라인 마이그레이션: 이 방법은 애플리케이션의 가동 중지 시간이 거의 없거나 최소일 때 사용됩니다. 온라인 마이그레이션에서는 소스 데이터베이스가 여러 단계에 걸쳐 AWS(으)로 마이그레이션됩니다. 초기 단계에서는 소스 데이터베이스가 계속 실행되는 동안 소스 데이터베이스의 데이터가 대상 데이터베이스에 복사됩니다. 후속 단계에서는 소스 데이터베이스에서 변경된 모든 내용이 대상 데이터베이스로 전달됩니다. 원본 및 대상 데이터베이스가 동기화되면 전환할 준비가 된 것입니다. 전환하는 동안 애플리케이션은 소스 데이터베이스에 대한 연결을 유지한 채 AWS에 있는 대상 데이터베이스로 연결을 전환합니다. 에서 사용할 수 있는 AWS Database Migration Service(AWS DMS) 또는 도구 [AWS Marketplace](#)(예: Attunity)를 사용하여 원본 및 대상 데이터베이스를 동기화할 수 있습니다.

SQL Server 데이터베이스 마이그레이션 방법

SQL Server 데이터베이스를 AWS에 마이그레이션하는 방법은 여러 가지가 있습니다. 평가 및 요구 사항에 따라 AWS 서비스와 SQL Server 기본 기능 중에서 선택할 수 있습니다. 이 섹션에서는 가장 일반적인 몇 가지 방법을 설명하며, 다음 두 표에 요약되어 있습니다. 이러한 방법 중 일부에 대한 자세한 설명은 이 안내서 뒷부분의 Amazon EC2 및 Amazon RDS 섹션에 포함되어 있습니다.

AWS 서비스

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
AWS DMS	Amazon EC2 Amazon RDS Amazon RDS Custom Amazon Aurora	<ul style="list-style-type: none"> 전체 로드 및 CDC 지원 모든 크기의 데이터베이스 지원 	AWS DMS 섹션
AWS Migration Hub Orchestrator	Amazon EC2 Amazon RDS	<ul style="list-style-type: none"> 사전 정의된 단계별 워크플로 템플릿 제공 기본 백업 및 복원 자동화 모든 SQL Server 에디션과 버전 지원 한 번에 하나 이상의 데이터베이스에 적용 가능 모든 크기의 데이터베이스 지원 	AWS Migration Hub Orchestrator 섹션
AWS Application Migration Service	Amazon EC2	<ul style="list-style-type: none"> 고도로 자동화된 리프트 앤드 시프트 솔루션 에이전트 기반, 블록 수준 복제 	이 가이드에서 다루지 않음(애플리케이션 마이그레이션 서비스 설명서 참조)

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
AWS Snowball 엣지	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"> 초대형 데이터베이스 지원(최대 210TB) 데이터 저장 및 복원에 Amazon Simple Storage Service(S3) 사용 	Snowball Edge 섹션

SQL 서버 기본 방법

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
기본 백업 및 복원	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"> 한 번에 하나 이상의 데이터베이스에 적용 가능 가동 중지 필요 모든 크기의 데이터베이스 지원 	기본 SQL Server 백업/복원 섹션 (AWS Migration Hub Orchestrator을(를) 기본 백업 및 복원을 자동화하는 데 사용할 수 있음)
로그 전달	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"> 데이터베이스별로 적용 지연될 수 있음 	로그 전달 섹션
데이터베이스 미러링	Amazon EC2	<ul style="list-style-type: none"> 데이터베이스별로 적용 SQL Server 에디션 을 기반으로 동기식 또는 비동기식일 수 있습니다. 보조 데이터베이스 는 읽을 수 없으며 	데이터베이스 미러링 섹션

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
		<p>대기 데이터베이스로 작동합니다.</p> <ul style="list-style-type: none"> • 자동 및 수동 장애 조치 모두 지원 	
<p>Always On 가용성 그룹</p>	<p>Amazon EC2 Amazon RDS Custom</p>	<ul style="list-style-type: none"> • 사용자 데이터베이스 세트에 적용 • 동기식 또는 비동기식 가능 • 보조 데이터베이스를 읽을 수 있음(SQL Server Enterprise 에디션만 해당) • 자동 및 수동 장애 조치 모두 지원 • 데이터베이스 그룹 수준에서 한 번에 여러 데이터베이스에 대한 장애 조치를 시작할 수 있습니다. 	<p>Always On 가용성 그룹 섹션</p>

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
Basic Always On 가용성 그룹	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> • SQL Server Standard 에디션에서 지원 • 가용성 그룹당 단일 사용자 데이터베이스에 적용 • 동기식 또는 비동기식 가능 • 자동 및 수동 장애 조치 모두 지원 • 가용성 그룹 수준에서 장애 조치를 시작할 수 있습니다. • 온프레미스와 AWS의 하이브리드 환경으로 사용할 수 있습니다. 	이 안내서에서는 다루지 않음(Microsoft 설명서의 단일 데이터베이스에 대한 기본 Always On 가용성 그룹 참조)

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
분산 가용성 그룹	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> • 다중 리전 SQL Server 배포에 사용할 수 있습니다. • 최신 버전의 SQL Server로 장애 조치할 수 있습니다. • Windows Server 장애 조치 클러스터링(WSC)을 대상 AWS 환경으로 확장할 필요가 없습니다. • Windows 기반(원본) 및 Linux 기반(대상) SQL Server 데이터베이스 간에 사용할 수 있습니다. • 온프레미스와 AWS 간의 하이브리드 SQL Server 배포로 사용할 수 있습니다. 	분산 가용성 그룹 섹션

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
트랜잭션 복제	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"> • 객체 세트(테이블, 뷰, 저장 프로시저)의 마이그레이션 지원 • 거의 실시간 데이터로 비동기 복제를 지원합니다. • 구독자 데이터베이스를 읽을 수 있습니다. • 복제를 수행하는 SQL Server 복제 작업을 면밀히 모니터링해야 합니다. 	트랜잭션 복제 섹션
일괄 복사 프로그램 (bcp)	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> • 소규모 데이터베이스 지원 • 가동 중지 필요 • 스키마는 대상에 미리 생성됩니다. • 데이터 이동에 사용되지만 메타데이터에는 사용되지 않습니다. 	이 안내서에서는 다루지 않음(Amazon RDS 설명서의 다른 방법을 사용한 SQL Server 데이터 가져오기 및 내보내기 , 대량 복사 섹션 참조)
분리 및 연결	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> • 백업 필요 없음 • 가동 중지 필요 • 중지, 분리, 파일 복사, Amazon EC2에 연결 포함 	이 안내서에서는 다루지 않음(Microsoft 설명서의 데이터베이스 분리 및 연결 참조)

마이그레이션 방법	대상	특성 및 제한 사항	추가 정보
가져오기/내보내기	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> 소규모 데이터베이스 지원 가동 중지 필요 스키마는 대상에 미리 생성됩니다. 데이터 이동에 사용되지만 메타데이터에는 사용되지 않습니다. 	이 안내서에서는 다루지 않음(Amazon RDS 설명서의 다른 방법을 사용한 SQL Server 데이터 가져오기 및 내보내기 참조)

네이티브 SQL 서버 백업/복원

Amazon RDS는 차등 및 전체 백업 파일(.bak 파일)을 사용하여 Microsoft SQL Server 데이터베이스에 기본 백업 및 복원을 할 수 있도록 지원합니다. 또한 Amazon RDS for SQL Server DB 인스턴스 또는 Amazon EC2 SQL Server 인스턴스에서 차등 복원 및 로그 복원 옵션을 지원하여 애플리케이션의 가동 중지 시간을 최소화합니다.

Note

Amazon RDS for SQL Server에서 전체, 차등 복원 작업 및 로그 복원 작업을 수행할 수 있습니다. 하지만 지금은 전체 및 차등 백업만 수행할 수 있습니다(로그 백업 제외).

기본 .bak 파일을 사용하는 것은 SQL Server 데이터베이스를 백업 및 복원하는 가장 간단한 방법입니다. 이 방법을 사용하여 데이터베이스를 Amazon RDS로, 또는 Amazon RDS에서 데이터베이스를 마이그레이션할 수 있습니다. DB 인스턴스 전체가 아닌 데이터베이스 하나를 백업 및 복원합니다. 또한 Amazon RDS for SQL Server DB 인스턴스 간에 데이터베이스를 이동할 수도 있습니다.

Amazon RDS를 사용하면 재해 복구를 위한 추가 보호 계층을 위해 Amazon Simple Storage Service(S3)에 백업 파일을 저장 및 전송할 수 있습니다. 예:

- 로컬 서버에서 데이터베이스의 전체 백업을 생성하고, 이를 S3 버킷으로 복사한 후, 기존 Amazon RDS SQL Server DB 인스턴스에서 복원할 수 있습니다.

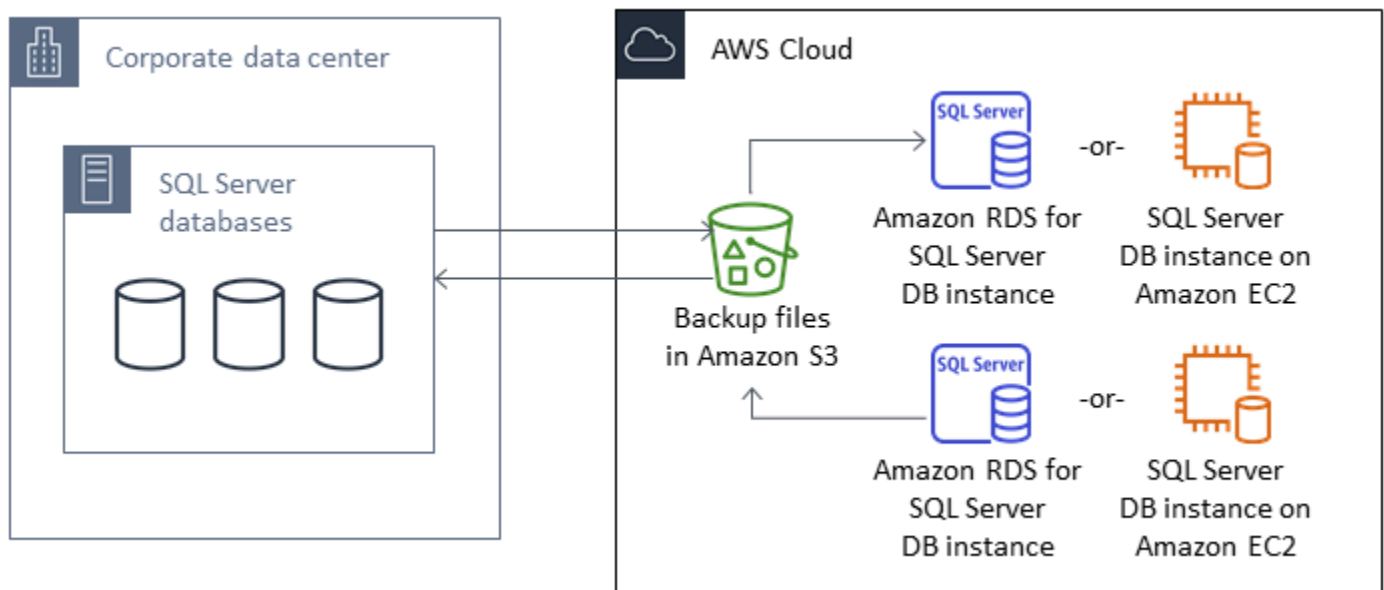
- Amazon RDS for SQL Server DB 인스턴스에서 백업을 생성하고, Amazon S3에 저장한 후, 어디든 원하는 곳에서 복원할 수 있습니다.
- [Amazon S3 Lifecycle](#) 구성 규칙을 구현하여 장기 백업을 보관하거나 삭제할 수 있습니다.

Amazon RDS for SQL Server는 SQL Server 기본 백업을 읽기 전용 복제본이 구성된 SQL Server DB 인스턴스로 복원하는 기능을 지원합니다. 즉, 기본 백업 파일을 Amazon RDS for SQL Server DB 인스턴스로 복원하기 전에 읽기 전용 복제본을 제거할 필요가 없습니다.

Note

Migration Hub Orchestrator를 사용하면 기본 백업 및 복원을 사용하여 Amazon EC2 또는 Amazon RDS로의 SQL Server 데이터베이스 마이그레이션을 자동화하고 조정할 수 있습니다. 자세한 내용은 [AWS Migration Hub Orchestrator 섹션](#)을 참조하세요.

다음 다이어그램은 기본 SQL Server 백업/복원 프로세스를 보여 줍니다. Migration Hub Orchestrator를 사용하여 이 프로세스를 자동화할 수 있습니다. 이 프로세스를 사용하여 SQL Server 데이터베이스를 Amazon EC2에 백업하고 복원할 수도 있습니다.



백업 및 복원을 자동화하려면 [Migration Hub Orchestrator 설명서](#)를 참조하세요.

Amazon S3를 사용하여 기본 백업/복원을 설정하려면 [Amazon RDS 설명서](#)를 참조하세요.

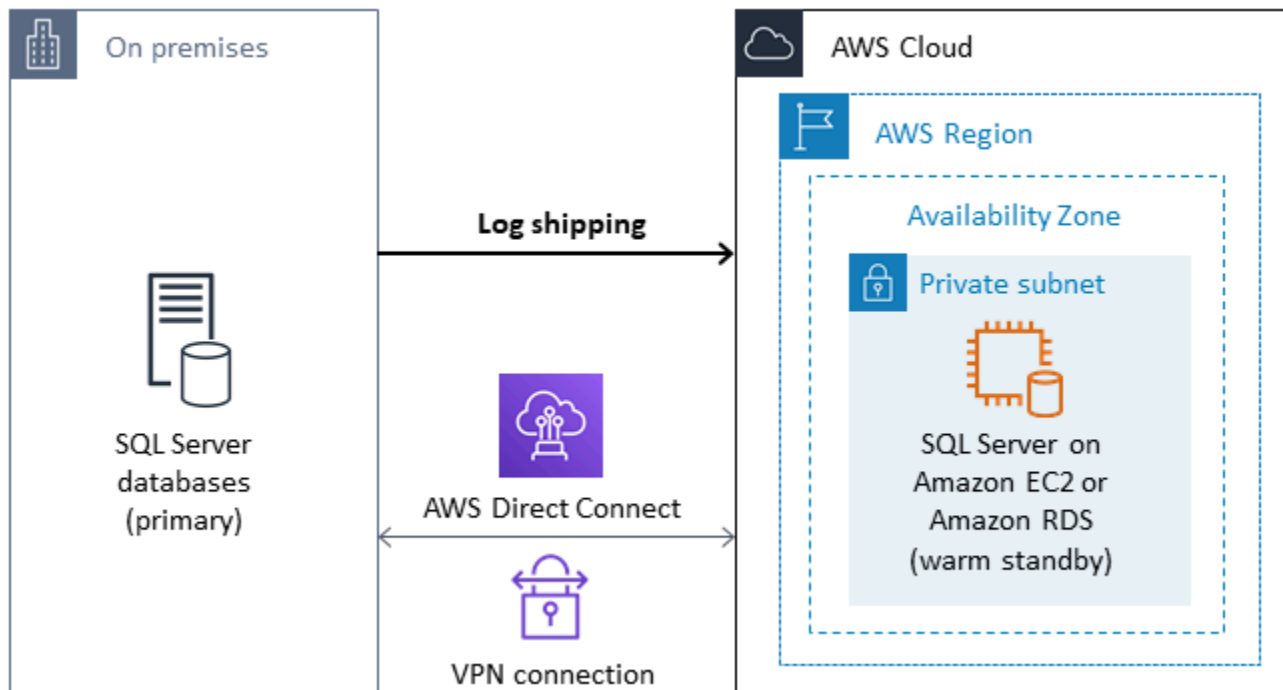
SQL Server 기본 백업 및 복원을 사용할 때의 제한 사항은 Amazon RDS [설명서의 제한 및 권장 사항을](#) 참조하세요.

로그 전달

로그 전달을 사용하여 기본 온프레미스 SQL Server 데이터베이스에서 AWS 클라우드의 EC2 인스턴스 또는 Amazon RDS for SQL Server DB 인스턴스에 배포된 하나 이상의 보조(웜 스탠바이) SQL Server 데이터베이스로 트랜잭션 로그 백업을 보낼 수 있습니다. Amazon RDS for SQL Server에서 로그 전달을 설정하려면 자체 사용자 지정 스크립트를 사용해야 합니다.

이 시나리오는 EC2 인스턴스 또는 Amazon RDS for SQL Server DB 인스턴스에 웜 스탠바이 SQL Server 데이터베이스를 구성하고 AWS 클라우드의 온프레미스 데이터베이스와 웜 스탠바이 서버 간에 트랜잭션 로그 백업을 비동기적으로 전송합니다. 그러면 트랜잭션 로그 백업이 웜 스탠바이 데이터베이스에 적용됩니다. 모든 로그가 적용되면 수동 장애 조치를 수행하고 클라우드로 전환할 수 있습니다.

이 옵션은 SQL Server의 모든 버전 및 에디션을 지원합니다. 데이터베이스를 AWS 클라우드로 마이그레이션한 후에는 고가용성 및 복원력을 위해 Always On 가용성 그룹을 사용하여 보조 복제본을 추가할 수 있습니다.



이 방법을 사용하여 Amazon EC2에서 SQL Server 데이터베이스의 고가용성, 데이터 보호 및 재해 복구를 달성하는 방법에 대한 자세한 내용은 Amazon EC2 for SQL Server 섹션의 [로그 전달](#)을 참조하세요.

데이터베이스 미러링

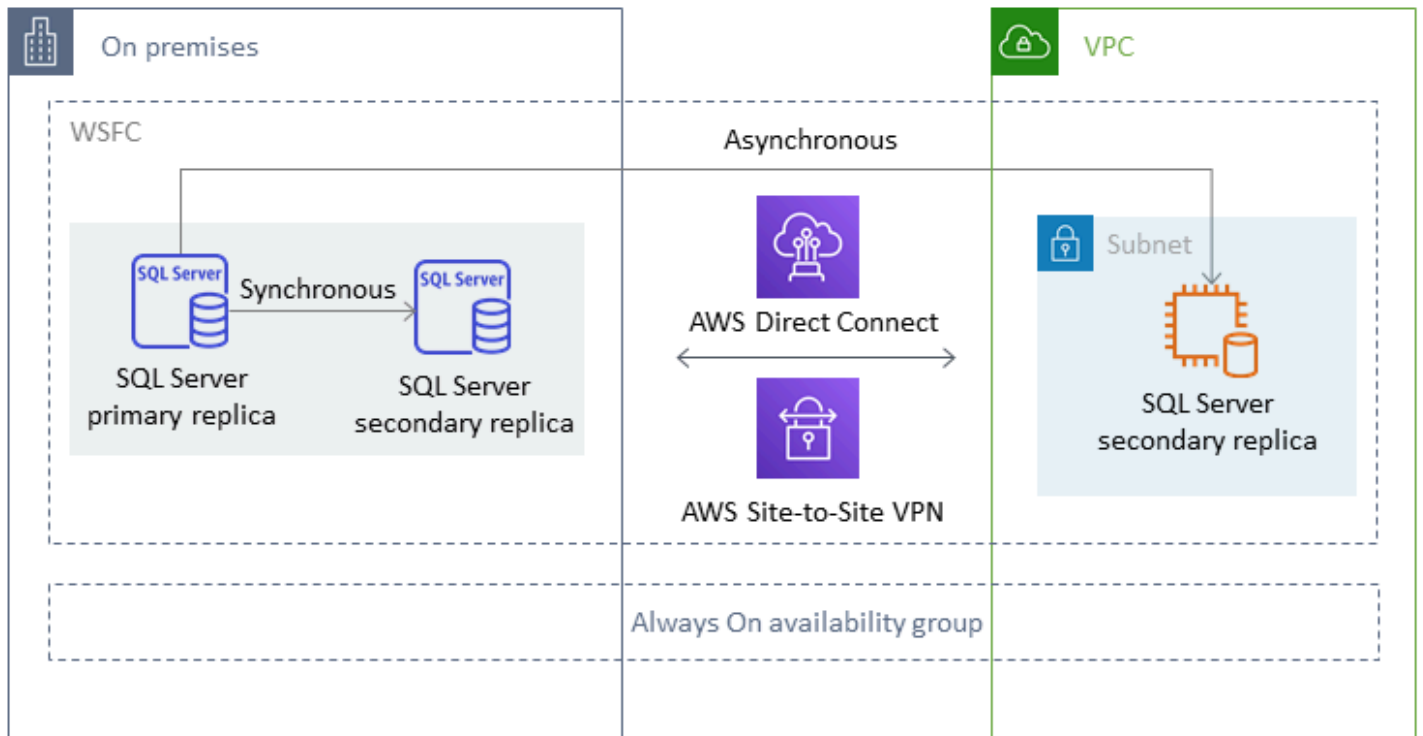
데이터베이스 미러링을 사용하여 SQL Server 데이터베이스를 위한 하이브리드 클라우드 환경을 설정할 수 있습니다. 이 옵션을 사용하려면 SQL Server Enterprise 에디션이 필요합니다. 이 시나리오는 주요 SQL Server 데이터베이스가 온프레미스에서 실행되고 클라우드에 읽기 스탠바이 데이터베이스를 생성합니다. 데이터를 비동기적으로 복제하고, 전환 준비가 되면 수동 장애 조치를 수행합니다. 데이터베이스를 AWS 클라우드로 마이그레이션한 후에는 고가용성 및 복원력을 위해 Always On 가용성 그룹을 사용하여 보조 복제본을 추가할 수 있습니다.

이 방법을 사용하여 Amazon EC2에서 SQL Server 데이터베이스의 고가용성, 데이터 보호 및 재해 복구를 달성하는 방법에 대한 자세한 내용은 Amazon EC2 for SQL Server 섹션의 [데이터베이스 미러링](#)을 참조하세요.

Always On 가용성 그룹

SQL Server Always On 가용성 그룹은 고가용성 및 재해 복구 솔루션을 제공하기 위한 고급 엔터프라이즈급 기능입니다. SQL Server 2014 이상 버전을 사용하는 경우 이 기능을 사용할 수 있습니다. 또한 Always On 가용성 그룹을 사용하여 온프레미스 SQL Server 데이터베이스를 AWS의 Amazon EC2으로 마이그레이션할 수 있습니다. 이 접근 방식을 사용하면 가동 중지 시간을 최소화하거나 전혀 없이 데이터베이스를 마이그레이션할 수 있습니다.

SQL Server Always On 가용성 그룹의 기존 온프레미스 배포가 있는 경우 기본 복제본과 보조 복제본이 가용성 그룹 내에서 데이터를 동기적으로 복제합니다. 따라서 데이터베이스를 AWS 클라우드로 마이그레이션하려면 Windows Server 장애 조치 클러스터링(WSFC) 클러스터를 클라우드로 확장됩니다. 이는 일시적으로, 마이그레이션 용도로만 사용할 수 있습니다. 그런 다음, 다음 다이어그램과 같이 AWS 클라우드에서 보조 복제본을 생성하고 비동기식 복제를 사용합니다. 보조 복제본이 기본 온프레미스 데이터베이스와 동기화된 후에는 전환 준비가 되면 언제든지 수동 장애 조치를 수행할 수 있습니다.



이 방법을 사용하여 Amazon EC2에서 SQL Server 데이터베이스의고가용성, 데이터 보호 및 재해 복구를 달성하는 방법에 대한 자세한 내용은 Amazon EC2 for SQL Server 섹션의 [Always On 가용성 그룹](#)을 참조하세요.

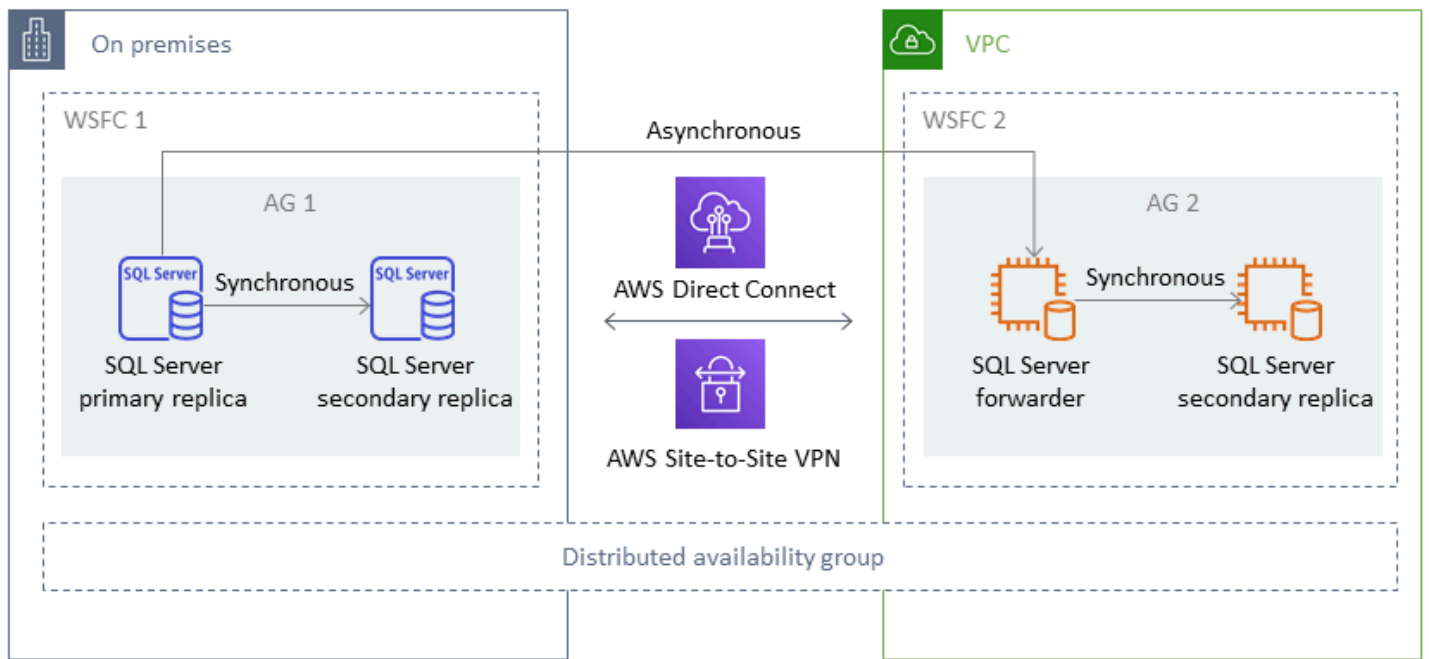
분산 가용성 그룹

분산 가용성 그룹은 두 개의 개별 가용성 그룹에 걸쳐 있습니다. 가용성 그룹으로 구성된 가용성 그룹이라고 생각하시면 됩니다. 기본 가용성 그룹은 서로 다른 두 WSFC 클러스터에 구성됩니다. 분산 가용성 그룹에 참여하는 가용성 그룹은 동일한 위치를 공유할 필요가 없습니다. 물리적 또는 가상, 온프레미스 또는 퍼블릭 클라우드에 있을 수 있습니다. 분산 가용성 그룹의 가용성 그룹은 동일한 버전의 SQL Server를 실행할 필요가 없습니다. 대상 DB 인스턴스는 원본 DB 인스턴스보다 최신 버전의 SQL Server를 실행할 수 있습니다.

분산 가용성 그룹 아키텍처를 사용하면 업무상 중요한 SQL Server 인스턴스 또는 데이터베이스를 AWS에 유연하게 리호스팅할 수 있습니다. 이는 중요한 SQL Server 데이터베이스를 AWS에 리프트 앤드 시프트(또는 리프팅 및 변환)할 수 있는 하이브리드 솔루션을 제공합니다.

분산 가용성 그룹 아키텍처를 사용하는 것이 기존 온프레미스 WSFC 클러스터를 AWS로 확장하는 것보다 더 효율적입니다. 데이터는 온프레미스 기본 복제본에서 AWS 복제본 중 하나(전달자)로만 전송됩니다. 전달자는 AWS 내의 다른 보조 읽기 전용 복제본으로 데이터를 전송하는 역할을 합니다.

다음 다이어그램에서 첫 번째 WSFC 클러스터(WSFC 1)는 온프레미스로 호스팅되며 온프레미스 가용성 그룹(AG 1)이 있습니다. 두 번째 WSFC 클러스터(WSFC 2)는 AWS에 호스팅되며 AWS 가용성 그룹(AG 2)이 있습니다. [AWS Direct Connect](#)은(는) 온프레미스 환경과 AWS 간의 전용 네트워크 연결로 사용됩니다. 온프레미스 가용성 그룹(AG 1)에는 두 개의 복제본(노드)이 있습니다. 노드 간 데이터 전송은 동기식이며 자동 장애 조치가 적용됩니다. 마찬가지로 AWS 가용성 그룹(AG 2)에도 두 개의 복제본이 있으며 두 복제본 간의 데이터 전송은 동기식으로 이루어지고 자동 장애 조치가 적용됩니다. 분산 가용성 그룹은 데이터베이스를 비동기 방식으로 동기화된 상태로 유지합니다. 데이터는 AG 1의 SQL Server 기본 복제본(온프레미스)에서 AG 2(AWS)의 기본 복제본(전달자)으로 전송됩니다. 전달자는 AWS의 다른 읽기 전용 복제본으로 데이터를 전송하고 이를 최신 상태로 유지하는 역할을 합니다. 온프레미스와 AWS 데이터베이스가 동기화된 후에는 분산 가용성 그룹을 AWS로 수동 장애 조치할 수 있습니다. AWS 데이터베이스는 애플리케이션의 읽기/쓰기 액세스를 위한 기본 데이터베이스가 됩니다.



Note

어느 시점이든 쓰기 작업에 사용할 수 있는 데이터베이스는 하나뿐입니다. 나머지 보조 복제본은 읽기 작업에 사용할 수 있습니다. 읽기 워크로드를 스케일 아웃하려면 AWS의 여러 가용 영역에 읽기 전용 복제본을 추가할 수 있습니다.

분산 가용성 그룹에 대한 자세한 내용은 다음을 참조하세요.

- [Microsoft SQL Server 설명서](#)
- AWS 데이터베이스 블로그의 [분산 가용성 그룹을 사용하여 하이브리드 Microsoft SQL Server 솔루션을 설계하는 방법](#)
- AWS 권장 가이드 웹 사이트의 [분산 가용성 그룹을 사용하여 SQL Server를 AWS\(으\)로 마이그레이션](#)

트랜잭션 복제

트랜잭션 복제는 두 데이터베이스 간의 변경 내용을 복제하는 데 사용되는 SQL Server 기술입니다. 이러한 변경에는 테이블(프라이머리 키 필요), 저장 프로시저, 뷰, 데이터와 같은 데이터베이스 개체가 포함될 수 있습니다. 복제 프로세스에는 게시자(데이터를 게시하는 기본 데이터베이스), 구독자(복제된 데이터를 수신하는 보조 데이터베이스) 및 배포자(트랜잭션 복제를 위해 메타데이터와 트랜잭션을 저장하는 서버)가 포함됩니다. Amazon EC2의 SQL Server 및 Amazon RDS for SQL Server 인스턴스에서 트랜잭션 복제를 사용할 수 있습니다.

트랜잭션 복제는 온프레미스(게시) 데이터베이스에 있는 객체 및 데이터의 스냅샷을 생성하여 구독자 데이터베이스로 전송합니다. 스냅샷이 구독자에게 적용된 후 게시자에서 이루어진 모든 후속 데이터 변경 및 스키마 수정 사항은 발생하는 즉시 구독자에게 전송됩니다. 그런 다음 게시자에서 발생한 것과 동일한 순서로 데이터 변경 내용이 구독자에게 계속 적용됩니다.

동기화가 완료되면 대상 SQL Server DB 인스턴스에서 검증을 수행합니다. 두 데이터베이스가 동기화되면 온프레미스 데이터베이스의 활동을 중지하고 복제가 완료되었는지 확인한 다음, 대상 SQL Server DB 인스턴스로 전환을 수행합니다. 그런 다음 푸시 구독을 중지하고 삭제한 다음 Amazon RDS for SQL Server를 사용할 수 있습니다.

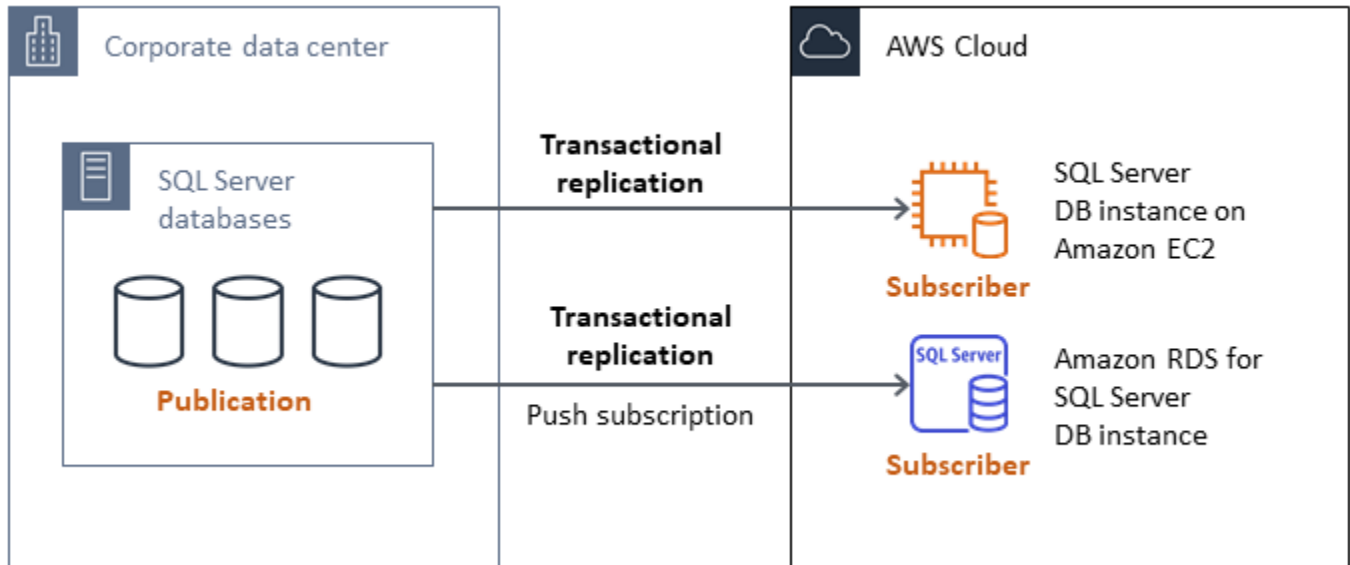
구독자 데이터베이스를 읽기 전용 데이터베이스로도 사용할 수 있습니다. 동기화 작업을 기록하는 배포자는 별도의 서버에 두는 것이 좋습니다. 대상 데이터베이스가 Amazon RDS for SQL Server에 있는 경우 푸시 구독을 설정하여 변경 내용을 구독자에게 전파할 수 있습니다.

다음과 같은 경우에는 트랜잭션 복제를 사용하는 것이 좋습니다.

- Amazon RDS 또는 Amazon EC2로 데이터를 일회성 마이그레이션.
- 스키마 수준 또는 테이블 수준 객체를 AWS(으)로 마이그레이션.
- 데이터베이스의 일부를 AWS(으)로 마이그레이션.
- 구독자를 추가하고 기존 SQL Server 복제 전략을 사용하여 가동 중지 기간을 최소화하면서 마이그레이션.

Amazon RDS for SQL Server로 데이터를 한 번 마이그레이션하기 위해 트랜잭션 복제를 사용하려는 경우 복제를 위한 단일 AZ 구성을 설정하는 것이 좋습니다. 복제 프로세스가 완료되면 환경을 다중 AZ 아키텍처로 변환하여 고가용성을 확보할 수 있습니다.

다음 다이어그램은 Amazon RDS와 Amazon EC2의 데이터베이스에 대한 트랜잭션 복제 프로세스를 보여 줍니다.



트랜잭션 복제에 대한 자세한 내용은 [Microsoft SQL Server 설명서](#) 및 AWS 데이터베이스 블로그의 [트랜잭션 복제를 사용하여 Amazon RDS for SQL Server로 마이그레이션하는 방법](#) 게시물을 참조하세요.

AWS Migration Hub Orchestrator

AWS Migration Hub Orchestrator은(는) SQL Server 데이터베이스를 Amazon EC2 또는 Amazon RDS로 마이그레이션하는 작업을 오케스트레이션하고 자동화하는 데 도움을 줍니다. AWS Migration Hub의 이 기능을 사용하면 모범 사례를 기반으로 구축된 사전 정의된 워크플로 템플릿을 사용하여 빠르게 시작할 수 있습니다. Migration Hub Orchestrator는 환경 준비 상태 및 연결 확인 등 마이그레이션 프로세스 과정에서 오류가 발생하기 쉬운 수동 작업을 자동화합니다. 또한 Migration Hub Orchestrator를 사용하여 SQL Server 데이터베이스 외에도 .NET 애플리케이션, SAP 워크로드, 가상 머신 이미지에 대한 마이그레이션을 오케스트레이션하고 가속화할 수 있습니다. [Migration Hub Orchestrator 콘솔](#)을 통해 이 도구에 액세스할 수 있습니다.

SQL Server 마이그레이션의 경우 Migration Hub Orchestrator는 다음 세 가지 사용 사례를 지원합니다.

- Amazon EC2에서 SQL Server 리호스팅 Migration Hub Orchestrator의 자동화된 기본 백업 및 복원을 사용하여 특정 SQL 서버를 선택하고 Amazon EC2에서 리호스팅할 수 있습니다. 자세한 내용은 Migration Hub Orchestrator 설명서에서 [Amazon EC2의 SQL 서버 리호스팅](#)을 참조하세요.
- Amazon RDS의 SQL Server 리플랫폼 Migration Hub Orchestrator에서 자동화된 기본 백업 및 복원을 사용하여 특정 SQL Server 데이터베이스를 선택하고 Amazon RDS에서 리플랫폼할 수 있습니다. 자세한 내용은 Migration Hub Orchestrator 설명서에서 [Amazon RDS의 SQL 서버 리플랫폼](#)을 참조하세요.
- Amazon EC2에서 Windows 및 SQL Server 애플리케이션 리호스팅 Amazon EC2 템플릿의 리호스팅 애플리케이션을 사용하여 .NET 및 SQL Server를 실행하는 Windows 서버를 Amazon EC2로 리프트 앤드 시프트할 수 있습니다. 자세한 내용은 Migration Hub Orchestrator 설명서에서 [Amazon EC2의 애플리케이션 리호스팅](#)을 참조하세요.

Migration Hub Orchestrator를 사용하면 SQL Server 마이그레이션 시 일정 및 예산 초과를 방지할 수 있습니다. 기타 주요 이점은 다음과 같습니다.

- 규범적 방법론을 사용하여 애플리케이션을 마이그레이션합니다. 검증된 마이그레이션 모범 사례를 기반으로 하는 사전 정의된 워크플로 템플릿을 사용하여 빠르게 시작할 수 있습니다. 필요에 따라 단계를 추가, 재정렬 및 제거하여 마이그레이션 워크플로를 사용자 지정할 수도 있습니다. 예를 들어 전환 승인을 위한 단계를 추가할 수 있습니다.
- 수동 단계를 자동화합니다. Migration Hub Orchestrator는 에이전트 설치, 온프레미스 이미지 가져오기, AWS의 대상 환경 프로비저닝, 원본 및 대상 환경 확인과 같은 수동 작업을 자동화합니다. 자동화를 통해 오류를 줄이면서 시간과 비용을 절약할 수 있습니다.
- 마이그레이션 워크플로를 조율합니다. Migration Hub Orchestrator는 인벤토리 메타데이터, 구성 사양 및 환경 컨텍스트를 재사용하여 마이그레이션 단계에서 사용되는 도구를 조율하며, 이러한 도구에 필요한 입력 수를 최소화합니다.

자세한 내용은 다음 리소스를 참조하세요.

- [Migration Hub Orchestrator 콘솔](#)
- [Amazon EC2에서 애플리케이션 리호스팅](#)(Migration Hub Orchestrator 사용 설명서)
- [Amazon RDS의 SQL 서버 리플랫폼](#)(Migration Hub Orchestrator 사용 설명서)
- [마이그레이션 워크플로](#)(Migration Hub Orchestrator 사용 설명서)
- [Migration Hub Orchestrator를 사용한 Microsoft SQL Server 마이그레이션의 간소화 및 가속화](#)(AWS 블로그 게시물)

- [AWS Migration Hub Orchestrator을\(를\) 사용한 Windows Server 이미지 마이그레이션의 간소화](#)
(AWS 블로그 게시물)

AWS Snowball 엣지

AWS Snowball Edge를 사용하여 최대 210TB의 매우 큰 데이터베이스를 마이그레이션할 수 있습니다. Snowball에는 온프레미스 서버에 연결하여 모든 데이터베이스 백업 또는 데이터를 Snowball 디바이스에 저장하는 10Gb 이더넷 포트가 있습니다. 데이터를 Snowball에 복사한 후 지정된 S3 버킷에 배치할 어플라이언스를 AWS(으)로 전송합니다. 그런 다음 Amazon S3에서 백업을 다운로드하고 EC2 인스턴스의 SQL Server에 복원하거나 `rds_restore_database` 저장 프로시저를 실행하여 데이터베이스를 Amazon RDS로 복원할 수 있습니다. 최대 8TB 크기의 데이터베이스에는 [AWS Snowcone](#)을 사용할 수 있습니다. 자세한 내용은 Amazon RDS 설명서의 [AWS Snowball Edge 설명서](#) 및 [SQL Server 데이터베이스 가져오기 및 내보내기](#), 데이터베이스 복원 섹션을 참조하세요.

SQL Server를 위한 동종 데이터베이스 마이그레이션

AWS 클라우드 환경에서 SQL Server 데이터베이스를 실행할 수 있는 기능을 제공합니다. 개발자와 데이터베이스 관리자의 경우 AWS 클라우드에서 SQL Server 데이터베이스를 실행하는 것은 데이터 센터에서 SQL Server 데이터베이스를 실행하는 것과 매우 유사합니다. 이 섹션에서는 온프레미스 환경 또는 데이터 센터에서 AWS 클라우드로 SQL Server 데이터베이스를 마이그레이션하는 옵션에 대해 설명합니다.

AWS 는 다음 표에 설명된 대로 SQL Server를 AWS 실행할 수 있는 세 가지 옵션을 제공합니다.

옵션	하이라이트	추가 정보
Amazon RDS 기반 SQL Server	관리형 서비스는 프로비저닝과 라이선스가 간편하고 비용 효율적이며 설치, 관리 및 유지 관리가 쉽습니다.	Amazon RDS for SQL Server 섹션
Amazon RDS Custom 기반 SQL Server	관리형 서비스이지만 데이터베이스 및 기본 운영 체제에 대한 관리 권한은 보유하고 있습니다.	SQL Server용 Amazon RDS Custom 섹션
Amazon EC2 기반 SQL Server	자체 관리형이며 완전한 제어 및 유연성을 제공합니다.	SQL Server용 Amazon EC2 섹션
VMware 클라우드상의 SQL 서버 AWS	VMware 클라우드에서 SQL Server 워크로드를 설정, 확장 및 운영하고 액티브 디렉터리 커넥터 AWS 및 Amazon AWS Directory Service S3와 통합할 수 있습니다.	SQL 서버를 AWS 위한 VMware 클라우드 온 섹션

애플리케이션 요구 사항, 데이터베이스 특징, 기능, 성장 용량 및 전반적인 아키텍처 복잡성에 따라 선택할 옵션이 결정됩니다. 여러 SQL Server 데이터베이스를 로 마이그레이션하는 경우 Amazon RDS에 적합한 데이터베이스도 있고 Amazon EC2에서 직접 실행하기에 더 적합한 데이터베이스도 있을 수 있습니다. AWS SQL Server 엔터프라이즈 에디션에서 실행되지만 SQL Server 스탠다드 에디션에 적합한 데이터베이스가 있을 수 있습니다. 비용과 라이선스를 절약하기 위해 Windows에서 실행되는 SQL Server 데이터베이스를 Linux 운영 체제에서 실행하도록 현대화할 수도 있습니다. 많은 AWS 고

객이 Amazon RDS, Amazon EC2 및 VMware Cloud에서 여러 SQL Server 데이터베이스 워크로드를 실행합니다. AWS

Note

Migration Hub Orchestrator를 사용하면 기본 백업 및 복원을 사용하여 Amazon EC2 또는 Amazon RDS로의 SQL Server 데이터베이스 마이그레이션을 자동화하고 조정할 수 있습니다. 자세한 내용은 [AWS Migration Hub Orchestrator 섹션](#)을 참조하십시오.

Amazon RDS for SQL Server

Amazon RDS for SQL Server는 AWS기반 SQL Server의 프로비저닝 및 관리를 간소화하는 관리형 데이터베이스 서비스입니다. Amazon RDS를 사용하면 클라우드에서 SQL Server 배포를 쉽게 설정, 운영 및 조정할 수 있습니다. Amazon RDS를 사용하면 비용 효율적이고 크기 조정이 가능한 컴퓨팅 파워를 갖춘 여러 버전의 SQL Server (2014, 2016, 2017, 2019, 2022) 및 에디션 (익스프레스, 웹, 스탠다드 및 엔터프라이즈 포함)을 몇 분 만에 배포할 수 있습니다. 범용 SSD 또는 프로비저닝된 IOPS SSD 스토리지를 사용하여 Amazon RDS for SQL Server DB 인스턴스를 프로비저닝할 수 있습니다. (자세한 내용은 AWS 설명서의 [Amazon RDS 스토리지 유형](#)을 참조하십시오.) 프로비저닝된 IOPS SSD는 빠르고 예측 가능하며 일관된 I/O 성능을 제공하도록 설계되었으며 I/O 집약적인 트랜잭션(OLTP) 데이터베이스 워크로드에 최적화되어 있습니다.

Amazon RDS는 프로비저닝, 백업, 소프트웨어 패치, 모니터링, 하드웨어 규모 조정 등 시간이 많이 걸리는 데이터베이스 관리 작업을 관리하므로 애플리케이션 개발에 집중할 수 있습니다. 또한 Amazon RDS for SQL Server는 다중 AZ 배포 및 읽기 전용 복제본(SQL Server 엔터프라이즈 에디션용)을 제공하여 프로덕션 워크로드에 대한 높은 가용성, 성능, 확장성 및 안정성을 제공합니다.

SQL Server에서 Amazon RDS로 마이그레이션하는 방법에 대한 자세한 내용은 AWS 규범적 지침 웹 [사이트의 플랫폼 변경 패턴](#)을 참조하십시오.

Amazon RDS를 선택해야 하는 시기

Amazon RDS for SQL Server는 다음과 같은 시기의 마이그레이션 옵션입니다.

- 비즈니스와 애플리케이션에 집중하고 데이터베이스 프로비저닝, 백업 및 복구 작업 관리, 보안 패치 관리, 사소한 SQL Server 버전 업그레이드, 스토리지 관리 등 차별화되지 않은 무거운 작업을 AWS 처리하기를 원합니다.

- 고가용성 데이터베이스 솔루션이 필요하고 데이터베이스 미러링, 장애 조치 클러스터 또는 Always On 가용성 그룹을 수동으로 설정하고 유지 관리할 필요 없이 Amazon RDS에서 제공하는 푸시 버튼 방식의 동기식 다중 AZ 복제를 활용하고자 합니다.
- 대규모 선불 투자를 하는 대신 SQL Server 라이선스 비용을 인스턴스 비용의 일부로 시간당 지불하고자 합니다.
- 데이터베이스 크기와 IOPS 요구 사항은 Amazon RDS for SQL Server에서 지원됩니다. 현재 최대 한도는 AWS 설명서의 [Amazon RDS DB 인스턴스 스토리지를](#) 참조하십시오.
- 데이터베이스의 백업이나 point-in-time 복구를 관리하고 싶지는 않습니다.
- 데이터베이스를 매일 관리하는 대신 성능 조정 및 스키마 최적화와 같은 높은 수준의 작업에 집중하고 싶습니다.
- 라이선스 복잡성에 대해 걱정할 필요 없이 워크로드 패턴에 따라 인스턴스 유형을 늘리거나 줄이고 싶습니다.

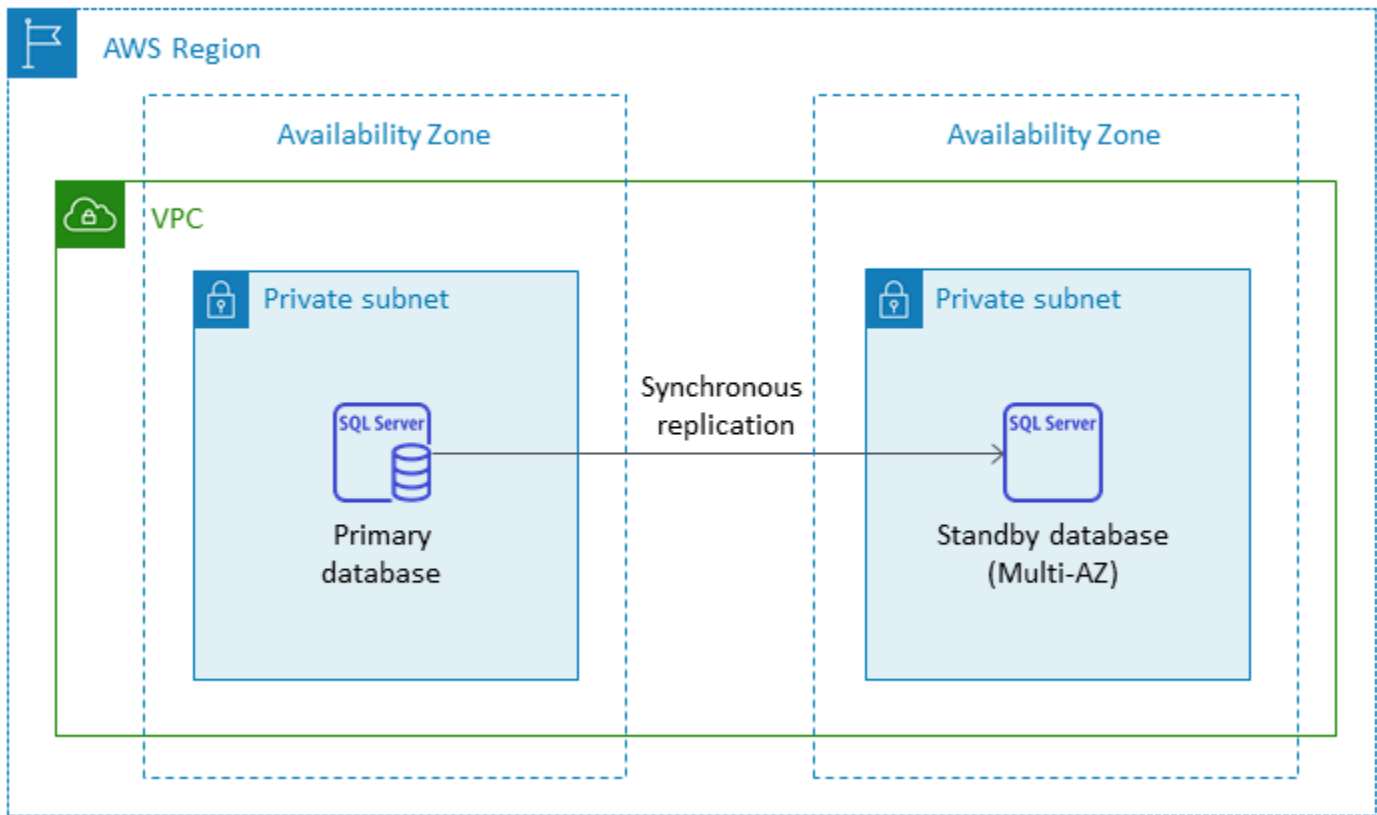
데이터베이스 및 프로젝트 요구 사항을 평가한 후 Amazon RDS for SQL Server로 마이그레이션하기로 결정했다면 다음 섹션에 제공된 세부 정보를 참조하고 이 안내서의 뒷부분에서 설명하는 [마이그레이션 모범 사례](#)를 검토하십시오.

현재 지원되는 SQL Server 기능, 버전 및 옵션에 대해서는 웹 사이트의 [SQL Server용 Amazon RDS](#) 기능, 이 안내서의 뒷부분에 나오는 [Amazon EC2와 Amazon RDS 중 선택 및](#) 설명서의 Amazon RDS의 [Microsoft SQL Server on Amazon AWS RDS](#)를 참조하십시오. AWS Amazon RDS Custom으로 옮기려면 [Amazon RDS Custom for SQL Server에 대한 요구 사항 및 제한 사항](#)을 검토해야 합니다.

높은 가용성

Amazon RDS는 다중 AZ 옵션으로 배포된 데이터베이스에 대한 고가용성 및 장애 조치 지원을 제공합니다. 다중 AZ 옵션으로 데이터베이스를 프로비저닝하는 경우 Amazon RDS는 자동으로 서로 다른 가용 영역에 동기식 대기 인스턴스를 프로비저닝하고 유지합니다. 기본 데이터베이스는 데이터를 대기 인스턴스에 동기적으로 복제합니다. 문제가 발생하는 경우 Amazon RDS는 비정상 인스턴스를 자동으로 복구하고 동기화를 다시 설정합니다. 인프라 장애 또는 가용 영역 중단이 발생하는 경우 Amazon RDS는 대기 인스턴스로 자동 장애 조치를 수행합니다. 대기 및 기본 데이터베이스가 완벽히 동기화되어 있는 경우에만 장애 조치가 이루어집니다. 기본 인스턴스와 대기 인스턴스의 엔드포인트는 동일하게 유지되므로 수동 개입 없이 장애 조치가 완료되는 즉시 데이터베이스 작업을 재개할 수 있습니다. 장애 조치 시간은 복구 프로세스 완료에 걸리는 시간에 따라 달라집니다. 트랜잭션이 크면 장애 조치 시간이 늘어납니다.

다음 다이어그램은 Amazon RDS for SQL Server 다중 AZ 배포 옵션을 보여줍니다.



다중 AZ 구성으로 SQL Server를 설정하면 Amazon RDS는 배포하는 SQL Server 버전에 따라 데이터베이스 미러링 또는 Always On 가용성 그룹을 사용하여 대기 데이터베이스 인스턴스를 자동으로 구성합니다. 특정 SQL Server 버전 및 에디션은 [Amazon RDS](#) 설명서에 나와 있습니다.

다중 AZ 배포에서는 가용성을 높이기 위해 기본 인스턴스의 자동 장애 조치 이전에 인스턴스 조정 또는 운영 체제(OS) 패치와 같은 시스템 업그레이드와 같은 작업이 대기 인스턴스에 먼저 적용됩니다.

SQL Server의 장애 조치 최적화로 인해 특정 워크로드를 특히 데이터베이스 미러링 배포에서 기본 인스턴스보다 대기 인스턴스에 더 많은 I/O 부하를 생성할 수 있습니다. 이 기능으로 인해 대기 인스턴스의 IOPS가 높아질 수 있습니다. Amazon RDS for SQL Server DB 인스턴스의 스토리지 유형 및 IOPS를 프로비저닝할 때는 기본 인스턴스와 예비 인스턴스 모두의 최대 IOPS 요구 사항을 고려하는 것이 좋습니다. 또한 클라이언트 드라이버가 이를 지원하는 경우 `MultiSubnetFailover=True`를 지정하여 장애 조치 시간을 크게 줄일 수 있습니다.

제한 사항

- 다중 AZ 옵션은 SQL Server Express 및 웹 에디션에는 사용할 수 없습니다. 이 기능은 SQL Server Standard 및 Enterprise Edition에만 사용할 수 있습니다.
- 대기 DB 인스턴스가 데이터베이스 읽기 작업을 허용하도록 구성할 수 없습니다.

- 교차 리전 다중 AZ는 지원되지 않습니다.
- Amazon RDS에서는 독립 실행형 DB 인스턴스에 중지 명령을 실행하여 인스턴스를 중지된 상태로 유지하여 컴퓨팅 요금이 발생하지 않도록 할 수 있습니다. 다중 AZ 구성에서는 Amazon RDS for SQL Server DB 인스턴스를 중지할 수 없습니다. 대신 인스턴스를 종료하고 종료 전에 최종 스냅샷을 찍고 필요할 때 스냅샷에서 새 Amazon RDS 인스턴스를 다시 만들 수 있습니다. 또는 다중 AZ 구성을 먼저 제거한 다음 인스턴스를 중지할 수 있습니다. 7일이 지나면 중지된 인스턴스가 다시 시작되므로 보류 중인 유지 관리를 적용할 수 있습니다.

추가적인 제한 사항은 Amazon RDS 설명서의 [Microsoft SQL Server 다중 AZ 배포 참고 및 권장 사항을](#) 참조하십시오.

읽기 전용 복제본

읽기 전용 복제본은 확장성과 로드 밸런싱을 제공합니다. SQL Server 읽기 전용 복제본은 읽기 전용으로 사용되는 Amazon RDS for SQL Server DB 인스턴스의 물리적 사본입니다. Amazon RDS는 읽기 전용 워크로드를 읽기 전용 복제본 DB 인스턴스로 오프로드하여 기본 DB 인스턴스의 부하를 줄이는데 도움이 됩니다. 기본 DB 인스턴스에 적용된 업데이트는 읽기 전용 복제본 인스턴스에 비동기식으로 복사됩니다.

읽기 전용 복제본을 요청할 때는 Amazon RDS가 소스 DB 인스턴스의 스냅샷을 캡처하여 이 스냅샷을 읽기 전용 복제본으로 전송합니다. 읽기 전용 복제본을 생성하고 삭제하는 동안에는 중단이 없습니다. Amazon RDS for SQL Server는 유지 관리 기간과 상관없이 읽기 전용 복제본을 업그레이드한 후 즉시 기본 데이터베이스를 업그레이드합니다. 모든 읽기 전용 복제본에는 읽기 전용 복제본 데이터베이스에 연결하는 데 사용하는 별도의 엔드포인트가 있습니다.

Amazon RDS for SQL Server를 사용하면 Always On 가용성 그룹을 구성하고 기본 DB 인스턴스와 읽기 전용 복제본 간의 안전한 네트워크 연결을 유지하여 읽기 전용 복제본을 쉽게 만들 수 있습니다.

기본 데이터베이스와 동일한 AWS 지역에 읽기 전용 복제본을 설정할 수 있습니다. Amazon RDS for SQL Server에서는 리전 간 읽기 전용 복제본을 지원하지 않습니다. 원본 DB 인스턴스 하나에 대해 최대 5개까지 읽기 전용 복제본을 생성할 수 있습니다.

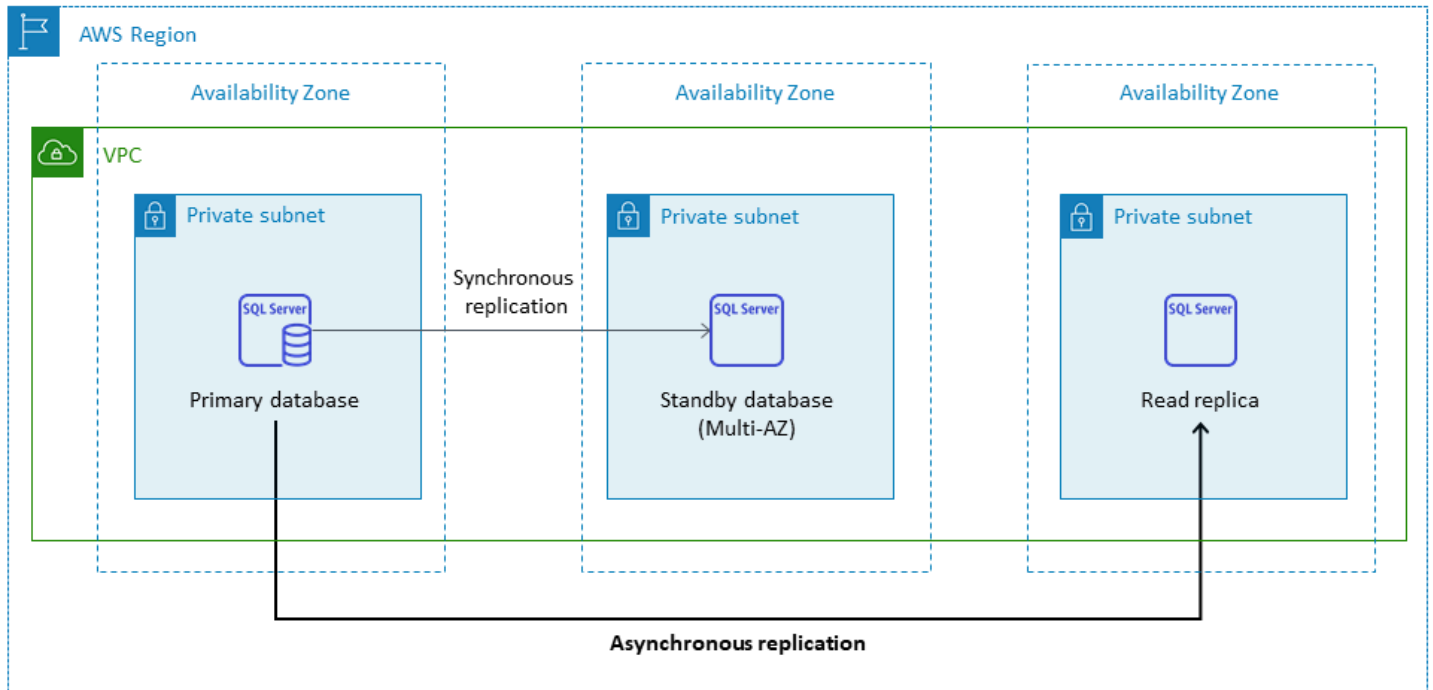
Note

읽기 전용 복제본은 다음 SQL Server 버전에 한하여 사용할 수 있습니다.

- SQL Server 2017 Enterprise Edition 14.00.3049.1 이상
- SQL Server 2016 Enterprise Edition 13.00.5216.0 이상

다중 AZ 환경을 위한 데이터베이스 미러링을 지원하는 SQL Server 버전 및 에디션은 읽기 전용 복제본을 제공하지 않습니다.

다음 다이어그램은 동일한 지역 내 다른 가용 영역에 읽기 전용 복제본이 있는 다중 AZ 환경의 Amazon RDS SQL Server DB 인스턴스를 보여줍니다. AWS 모든 AWS 지역이 2개 이상의 가용 영역을 제공하는 것은 아니므로 [이 전략을 채택하기 전에 사용하려는 지역을 확인해야](#) 합니다.



SQL Server 읽기 전용 복제본은 쓰기 작업을 허용하지 않습니다. 하지만 읽기 전용 복제본을 쓰기 가능하도록 승격할 수 있습니다. 승격한 후에는 다시 읽기 전용 복제본으로 되돌릴 수 없습니다. 이 인스턴스는 원래의 기본 데이터베이스 인스턴스와 관계가 없는 독립형 단일 DB 인스턴스가 됩니다. 승격된 읽기 전용 복제본의 데이터는 승격 요청이 이루어진 시점까지 원본 DB 인스턴스의 데이터와 일치합니다. 소스 DB 인스턴스의 SQL Server DB 엔진 버전과 모든 읽기 전용 복제본은 동일합니다.

효율적인 복제를 위해 다음 작업을 수행하는 것이 좋습니다.

- 읽기 전용 복제본도 각각 원본 DB 인스턴스와 동일한 컴퓨팅 및 스토리지 리소스를 갖도록 설정합니다.
- 백업 보존 기간을 0이 아닌 다른 값으로 설정하여 원본 DB 인스턴스의 자동 백업을 활성화해야 합니다.

- 원본 DB 인스턴스는 Always On 가용성 그룹(AG)이 있는 다중 AZ 환경에서 배포되어야 합니다.

SQL Server 버전 지원, 에디션 및 제한 사항에 대한 내용은 Amazon RDS 설명서의 [SQL Server에서의 읽기 전용 복제본 제한](#)을 참조하십시오.

읽기 전용 복제본 사용에 대한 자세한 내용은 설명서에서 [읽기 전용 복제본](#) 사용 및 [Amazon RDS용 SQL Server 읽기 전용 복제본](#) 사용을 참조하십시오. AWS 데이터 전송 요금에 대한 자세한 내용은 [Amazon RDS 요금](#)을 참조하십시오.

재해 복구

Amazon RDS for SQL Server를 사용하면 신뢰할 수 있는 지역 간 재해 복구(DR) 전략을 만들 수 있습니다. DR 솔루션을 만드는 주된 이유는 비즈니스 연속성과 규정 준수입니다.

- 효과적인 DR 전략을 세우면 재해 발생 시 시스템 종단을 최소화하거나 전혀 중단하지 않고 시스템을 계속 가동하고 운영할 수 있습니다. 안정적이고 효과적인 지역 간 DR 전략을 사용하면 지역 전체가 오프라인 상태가 되더라도 비즈니스를 계속 운영할 수 있습니다.
- 지역 간 DR 솔루션은 감사 및 규정 준수 요구 사항을 충족하는 데 도움이 됩니다.

Recovery Point Objective(RPO), Recovery Time Objective(RTO) 및 비용은 DR 전략을 개발할 때 고려해야 할 세 가지 주요 지표입니다. 리전 간 복제본을 제공하기 위한 다른 옵션에 대해서는 [AWS Marketplace](#)을 참조하십시오. 이러한 접근 방식에 대한 자세한 내용은 데이터베이스 블로그의 [SQL Server용 Amazon RDS의 지역 간 재해 복구를](#) 참조하십시오. AWS

SQL Server용 Amazon RDS Custom

타사 애플리케이션의 사용자 지정 요구 사항 때문에 Amazon RDS와 같은 완전 관리형 서비스로 전환할 수 없는 경우 SQL Server용 Amazon RDS Custom으로 마이그레이션할 수 있습니다. Amazon RDS Custom을 사용하면 데이터베이스 및 기본 운영 체제에 대한 관리 권한을 보유하여 종속 애플리케이션을 활성화할 수 있습니다.

SQL Server용 Amazon RDS Custom를 선택해야 하는 경우

Amazon RDS Custom for SQL Server는 다음과 같은 경우에 적합한 마이그레이션 옵션입니다.

- 기본 운영 체제 및 데이터베이스 환경에 액세스해야 하는 레거시, 커스텀 및 패키지 애플리케이션이 있습니다.

- 공급업체 기반 애플리케이션 배포 요구 사항을 충족하려면 관리자 액세스 권한이 필요합니다.
- 설정을 구성하고 패치를 설치하며 종속 애플리케이션의 요구 사항을 충족하도록 기본 기능을 활성화하려면 기본 운영 체제에 액세스가 필요합니다.
- 데이터베이스 및 애플리케이션 요구 사항에 맞게(사용자 지정 데이터베이스 패치를 적용하거나 OS 패키지를 수정하여) 데이터베이스 환경에 액세스하고 사용자 지정하고자 합니다.

작동 방식

Amazon RDS Custom for SQL Server를 사용하려면 Amazon RDS Custom for SQL Server에 있는 [요구 사항을](#) 검토하십시오. [Amazon RDS 설명서](#)에 설명된 대로 먼저 SQL Server용 Amazon RDS Custom에 대한 환경을 설정해야 합니다. 환경을 설정한 후에는 다음 다이어그램에 나와 있는 다음 단계를 따르십시오.

1. Amazon RDS Custom에서 제공하는 SQL Server 엔진 버전에서 SQL Server DB 인스턴스용 Amazon RDS Custom을 생성합니다.

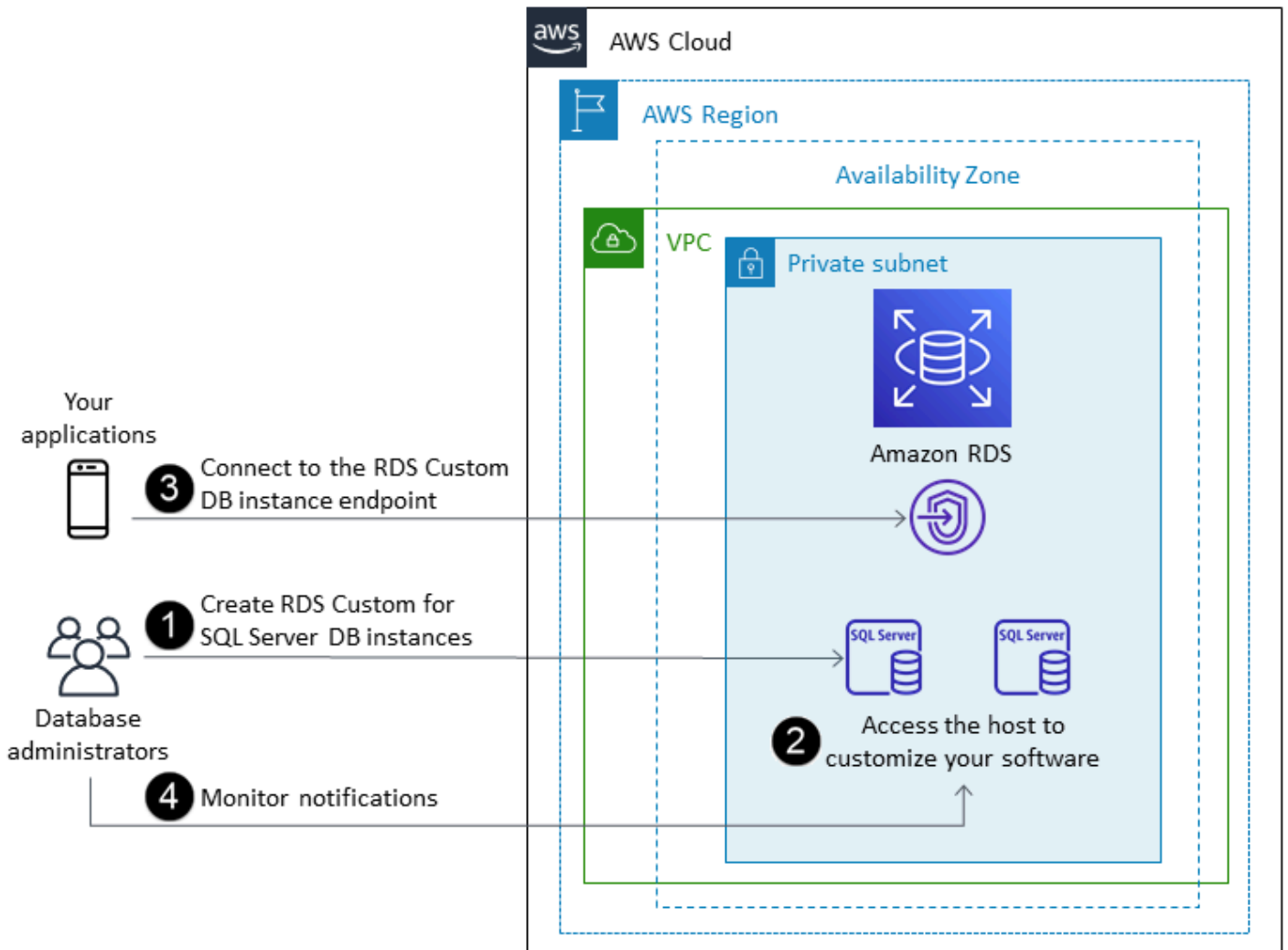
SQL Server용 Amazon RDS Custom은 현재 설명서에 나열된 [지원되는 DB 인스턴스 클래스](#)를 사용하여 SQL Server 2022 또는 2019 엔터프라이즈, 스탠다드 또는 개발자 에디션을 지원합니다. 자세한 내용은 [RDS Custom for SQL Server DB 인스턴스 생성](#)을 참조하십시오.

2. 애플리케이션을 Amazon RDS Custom DB 인스턴스 엔드포인트에 연결합니다.

자세한 내용은 RDP를 [AWS Systems Manager를 사용하여 RDS Custom DB 인스턴스에 연결](#) 및 [RDP를 사용하여 RDS Custom DB 인스턴스에 연결](#)을 참조하십시오.

3. (선택 사항) 호스트에 액세스하여 소프트웨어를 커스터마이징합니다.
4. Amazon RDS Custom 자동화에서 생성된 알림 및 메시지를 모니터링합니다.

이러한 단계에 대한 자세한 내용은 [Amazon RDS Custom 설명서](#)를 참조하십시오.



Amazon RDS Custom for SQL Server에서는 소프트웨어를 설치하여 사용자 지정 애플리케이션 및 에이전트를 실행할 수 있습니다. 호스트에 대한 액세스 권한이 있으므로 레거시 애플리케이션을 지원하도록 파일 시스템을 수정할 수 있습니다. 또한 Amazon RDS Custom DB 인스턴스에서 커스텀 데이터베이스 패치를 적용하거나 OS 패키지를 수정할 수 있습니다.

Amazon RDS Custom은 모니터링, 백업 및 인스턴스 복구를 자동으로 제공하고 DB 인스턴스가 지원되는 AWS 인프라, 운영 체제 및 데이터베이스를 사용하도록 합니다. 인스턴스를 사용자 지정하려는 경우 Amazon RDS Custom 자동화를 최대 24시간 동안 일시 중지했다가 사용자 지정 작업이 완료되면 다시 시작할 수 있습니다. 자동화를 일시 중지하면 Amazon RDS 자동화가 사용자 지정을 직접 방해하는 것을 방지할 수 있습니다.

자동화를 재개하면 [지원 경계에](#) 따라 데이터베이스 또는 운영 체제 환경의 사용자 지정이 Amazon RDS Custom 자동화에 방해가 되는지 아니면 중단되는지가 결정됩니다. Amazon RDS Custom은 변경 사항으로 인해 DB 인스턴스가 지원 경계를 벗어나지 않는 한 호스트 및 데이터베이스 환경의 사

용자 지정을 지원합니다. 지원 경계 검사는 기본적으로 30분마다 수행되며, 스냅샷 삭제 또는 DB 인스턴스를 모니터링하는 Amazon RDS Custom 에이전트 제거와 같은 이벤트 이후에도 수행됩니다. Amazon RDS Custom 에이전트는 Amazon RDS Custom 기능을 보장하는 데 중요한 구성 요소입니다. 에이전트를 제거하면 Amazon RDS Custom은 1분 후에 지원 경계 검사를 실행하고 DB 인스턴스를 지원 경계 밖으로 이동합니다.

Amazon RDS Custom for SQL Server DB 인스턴스를 설정하는 경우 소프트웨어 라이선스가 포함됩니다. 즉 SQL Server 라이선스를 별도로 구매할 필요가 없습니다. 라이선스에 대한 자세한 내용은 [AWS 서비스 약관](#)의 섹션 10.5를 참조하십시오. 활성 AWS 프리미엄 지원 계정이 있는 경우 Amazon RDS 사용자 지정 AWS 프리미엄 지원에 문의하여 SQL Server 관련 문제를 문의할 수 있습니다.

SQL Server용 Amazon RDS 사용자 지정은 제한된 수의 AWS 리전 DB 인스턴스 클래스에서만 지원됩니다. 이러한 제한 및 기타 제한 사항은 SQL Server용 Amazon RDS Custom 설명서에서 [요구 사항 및 제한](#) 페이지를 참조하십시오.

온프레미스 SQL Server 데이터베이스를 사용하는 경우 [Amazon RDS 설명서](#)에 설명된 프로세스에 따라 기본 백업 및 복원 유틸리티를 사용하여 Amazon RDS Custom for SQL Server로 데이터베이스를 마이그레이션할 수 있습니다.

자세한 내용은 다음 리소스를 참조하십시오.

- [신규 — SQL Server용 Amazon RDS 사용자 지정 정식 출시](#) (AWS 뉴스 블로그)
- [SQL 서버용 Amazon RDS Custom과 SQL Server용 Amazon RDS 간의 SQL 서버 복제 구성](#) (데이터베이스 블로그)
- [사용자 지정 로그 전달을 사용하여 SQL Server 마이그레이션을 위해 온프레미스 또는 Amazon EC2 SQL Server를 Amazon RDS로 마이그레이션하는 작업을 자동화합니다](#) (데이터베이스 블로그)
- [SQL Server용 Amazon RDS 사용자 지정의 상시 가동 가용성 그룹을 사용하여고가용성을 AWS 구성하십시오](#) (데이터베이스 블로그)
- [AWS CloudFormation 템플릿을 사용하여 SQL Server용 Amazon RDS 사용자 지정 시작하기](#) (네트워크 설정) (AWS 데이터베이스 블로그)
- [분산된 가용성 그룹을 사용하여 온프레미스 SQL Server 워크로드를 SQL Server용 Amazon RDS Custom으로 마이그레이션합니다](#) (AWS 데이터베이스 블로그)
- [SQL Server용 Amazon RDS 커스텀 \(데이터베이스 블로그\) 에서 BYOM \(Bring Your Own Media\) 을 사용하여 SQL Server 비용을 최적화하십시오](#). AWS

Amazon EC2 for SQL Server

Amazon EC2는 자체 관리형 SQL Server 데이터베이스를 지원합니다. 즉, 인프라 및 데이터베이스 환경 설정을 완전히 제어할 수 있습니다. Amazon EC2에서 데이터베이스를 실행하는 것은 자체 서버에서 데이터베이스를 실행하는 것과 매우 유사합니다. 데이터베이스 및 운영 체제 수준 액세스를 완전히 제어할 수 있으므로 원하는 도구를 사용하여 운영 체제, 데이터베이스 소프트웨어, 패치, 데이터 복제, 백업 및 복원을 관리할 수 있습니다. 이 마이그레이션 옵션을 사용하려면 아키텍처 모범 사례에 따라 EC2 인스턴스, 스토리지 볼륨, 확장성, 네트워킹, 보안을 비롯한 모든 구성 요소를 설정, 구성, 관리 및 조정해야 합니다. AWS 동일 지역 또는 다른 AWS 지역의 인스턴스 전반에서 데이터 복제 및 복구를 책임져야 합니다.

SQL Server에서 Amazon EC2로 마이그레이션하는 방법에 대한 자세한 내용은 규범 지침 웹 [사이트의 재호스팅](#) 패턴을 참조하십시오. AWS

Amazon EC2를 선택해야 하는 경우

Amazon EC2는 다음과 같은 경우 SQL Server 데이터베이스에 적합한 마이그레이션 옵션입니다.

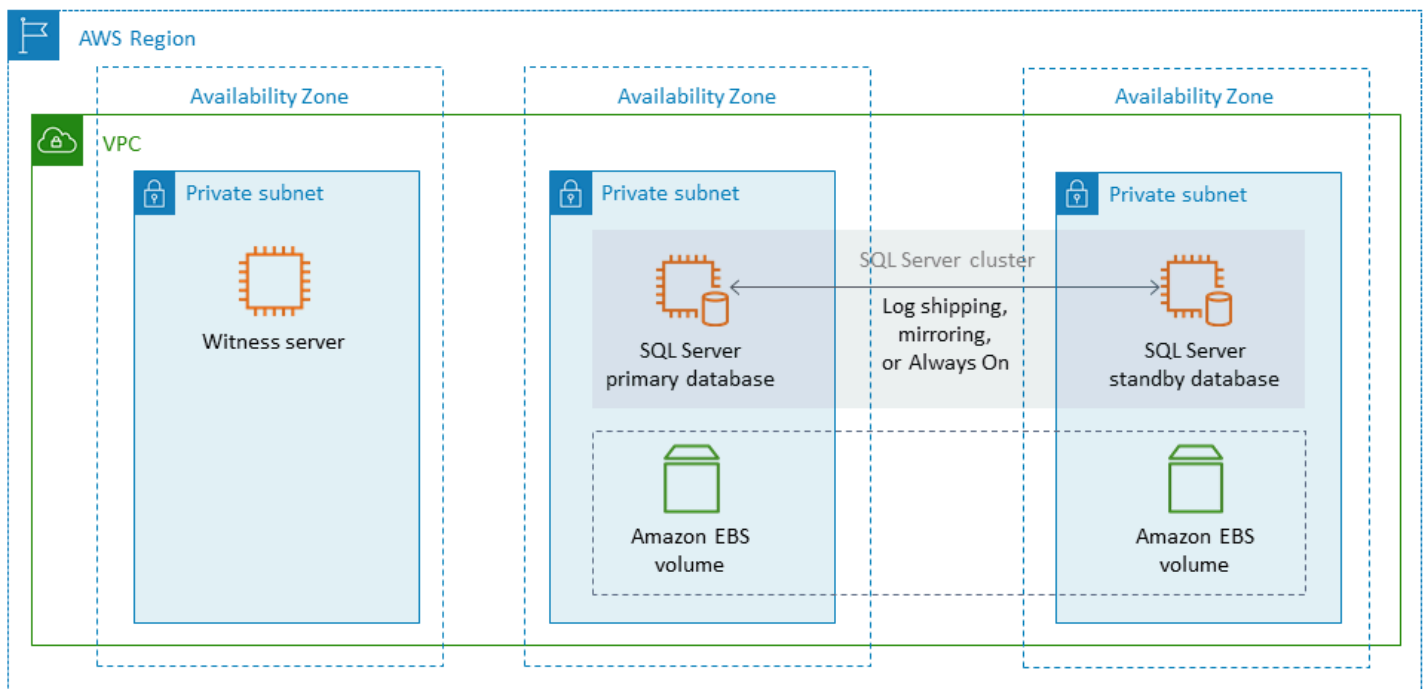
- 데이터베이스를 완전히 제어하고 기본 운영 체제, 데이터베이스 설치 및 구성에 액세스할 수 있어야 합니다.
- 백업 및 복구, 운영 체제 및 데이터베이스 패치 적용, 운영 체제 및 데이터베이스 매개 변수 조정, 보안 관리,고가용성 또는 복제 구성 등 데이터베이스를 관리하고자 합니다.
- 현재 Amazon RDS에서 지원하지 않는 기능과 옵션을 사용하고 싶습니다. 자세한 내용은 Amazon RDS 설명서의 [지원되지 않는 기능 및 지원이 제한된 기능](#)을 참조하십시오.
- Amazon RDS에서 지원하지 않는 특정 SQL Server 버전이 필요합니다. 지원되는 버전 및 에디션 목록은 Amazon RDS 설명서의 [Amazon RDS 기반 SQL Server 버전을](#) 참조하십시오.
- 데이터베이스 크기 및 성능 요구 사항은 현재 Amazon RDS for SQL Server 서비스를 초과합니다. 자세한 내용은 Amazon RDS 설명서의 [Amazon RDS DB 인스턴스 스토리지를](#) 참조하십시오.
- 애플리케이션과 호환되지 않을 수 있는 자동 소프트웨어 패치는 피하는 것이 좋습니다.
- Amazon RDS for SQL Server 라이선스 포함 모델을 사용하는 대신 자체 라이선스를 사용하는 것이 좋습니다.
- 현재 한도보다 높은 IOPS와 스토리지 용량을 달성하고자 합니다. 자세한 내용은 Amazon RDS 설명서의 [Amazon RDS DB 인스턴스 스토리지를](#) 참조하십시오.

Amazon EC2에서 현재 지원되는 SQL Server 기능 및 버전 목록은 이 안내서 뒷부분에 나오는 [Amazon EC2와 Amazon RDS 중 선택을](#) 참조하십시오.

높은 가용성

Amazon EC2 기반 SQL Server 데이터베이스와 함께 SQL Server가 지원되는 모든 복제 기술을 사용하여 고가용성, 데이터 보호 및 재해 복구를 달성할 수 있습니다. 일반적인 솔루션으로는 로그 전달, 데이터베이스 미러링, Always On 가용성 그룹, Always On 장애 조치 클러스터 인스턴스 등이 있습니다.

다음 다이어그램은 단일 AWS 지역 내 여러 가용 영역에서 Amazon EC2에서 SQL Server를 사용하는 방법을 보여줍니다. 기본 데이터베이스는 읽기-쓰기 데이터베이스이고, 보조 데이터베이스는 로그 전달, 데이터베이스 미러링 또는 고가용성을 위한 Always On 가용성 그룹으로 구성됩니다. 기본 데이터베이스의 모든 트랜잭션 데이터는 전송되며 로그 전달의 경우 보조 데이터베이스에 비동기적으로, Always On 가용성 그룹 및 미러링의 경우 비동기적으로 적용할 수 있습니다.



로그 전달

로그 전달을 사용하면 기본 데이터베이스 인스턴스에서 별도의 DB 인스턴스에 있는 하나 이상의 보조 데이터베이스(웜 스탠바이라고도 함)로 트랜잭션 로그 백업을 자동으로 전송할 수 있습니다. 로그 전달은 SQL Server 에이전트 작업을 사용하여 트랜잭션 로그 백업의 백업, 복사 및 적용 프로세스를 자동화합니다. 로그 전달은 일반적으로 재해 복구 기능으로 간주되지만 기본 DB 인스턴스에 장애가 발생할 경우 보조 DB 인스턴스를 승격시켜 고가용성을 제공할 수도 있습니다. RTO와 RPO가 유연하거나 데이터베이스가 업무상 매우 중요하다고 간주되지 않는 경우 SQL Server 데이터베이스의 가용성을 높이기 위해 로그 전달을 사용해 보십시오.

로그 전달은 필요할 때 기본 데이터베이스의 읽기 전용 복사본으로 사용할 수 있도록 보조 데이터베이스에 대한 액세스를 제공함으로써 데이터베이스의 가용성을 높입니다. 래그 지연(더 긴 지연 시간)을 구성하여 이러한 변경 내용이 보조 데이터베이스로 전달되기 전에 기본 데이터베이스에서 실수로 변경된 데이터를 복구할 수 있습니다.

기본 및 보조 DB 인스턴스를 별도의 가용 영역에서 실행하고 모니터 인스턴스를 배포하여 로그 전달의 모든 세부 정보를 추적하는 것이 좋습니다. 로그 전달 그룹의 백업, 복사, 복원 및 실패 이벤트는 모니터 인스턴스에서 사용할 수 있습니다. 로그 전달 구성은 기본 서버에서 보조 서버로 자동 장애 조치되지 않습니다. 하지만 기본 데이터베이스를 사용할 수 없게 되면 보조 데이터베이스를 수동으로 온라인 상태로 만들 수 있습니다.

로그 전달은 대개 재해 복구 솔루션으로 사용되지만 애플리케이션 요구 사항에 따라 고가용성 솔루션으로도 사용될 수 있습니다. 다음과 같은 경우 로그 전달을 사용하십시오.

- RTO 및 RPO 요구 사항은 유연합니다. 로그 전달은 분 단위의 RPO와 몇 분에서 몇 시간의 RTO를 제공합니다.
- 보조 데이터베이스로의 자동 장애 조치는 필요하지 않습니다.
- 보조 데이터베이스에서 읽고 싶지만 복원 작업 중에는 가독성이 필요하지 않습니다.

로그 전달에 대한 자세한 정보는 [Microsoft SQL Server 설명서](#)를 참조하십시오.

데이터베이스 미러링

데이터베이스 미러링은 EC2 인스턴스에 있는 데이터베이스를 가져와 별도의 DB 인스턴스에 전체 또는 거의 완전한 읽기 전용 사본(미러)을 제공합니다. Amazon RDS는 데이터베이스 미러링을 사용하여 Amazon RDS for SQL Server에 대한 다중 AZ 지원을 제공합니다. 이 기능은 데이터베이스의 가용성과 보호를 향상시키고 업그레이드 중에도 데이터베이스를 계속 사용할 수 있는 메커니즘을 제공합니다.

Note

[Microsoft 설명서에](#) 따르면 데이터베이스 미러링은 향후 SQL Server 버전에서 제거될 예정입니다. 대신 Always On 가용성 그룹을 사용할 계획을 세워야 합니다.

데이터베이스 미러링에서 SQL 서버는 다음 세 가지 역할 중 하나를 수행할 수 있습니다.

- 기본 서버는 데이터베이스의 기본 읽기/쓰기 버전을 호스팅합니다.

- 주요 데이터베이스의 복사본을 호스팅하는 미러 서버.
- 선택적 미러링 모니터 서버. 이 서버는 보안 강화 모드에서만 사용할 수 있습니다. 데이터베이스 미러 상태를 모니터링하고 기본 데이터베이스에서 미러 데이터베이스로의 장애 조치를 자동화합니다.

기본 서버와 미러 서버 간에 미러링 세션이 설정됩니다. 미러링 중에는 주요 데이터베이스에서 수행된 모든 데이터베이스 변경 내용이 미러 데이터베이스에서도 수행됩니다. 데이터베이스 미러링은 동기 또는 비동기 작업일 수 있습니다. 이는 보호 우선 모드와 고성능 모드의 두 가지 미러링 작동 모드로 결정됩니다.

- 안전 강화 모드: 이 모드는 동기 작업을 사용합니다. 이 모드에서 데이터베이스 미러링 세션은 주요 데이터베이스의 삽입, 업데이트 및 삭제 작업을 최대한 빨리 미러 데이터베이스와 동기화합니다. 데이터베이스가 동기화되는 즉시 주요 데이터베이스와 미러 데이터베이스 모두에서 트랜잭션이 커밋됩니다. 미러 데이터베이스가 같거나 다른 가용 영역에 있지만 같은 AWS 지역 내에서 호스팅되는 경우 이 운영 모드를 사용하는 것이 좋습니다.
- 고성능 모드: 이 모드는 비동기 작업을 사용합니다. 이 모드에서 데이터베이스 미러링 세션은 주요 데이터베이스의 삽입, 업데이트 및 삭제 작업을 미러 데이터베이스와 동기화하지만 주요 데이터베이스가 트랜잭션을 커밋하는 시간과 미러 데이터베이스가 트랜잭션을 커밋하는 시간 사이에 지연이 발생할 수 있습니다. 미러 데이터베이스가 서로 다른 AWS 지역에 있는 경우 이 모드를 사용하는 것이 좋습니다.

다음과 같은 경우 데이터베이스 미러링을 사용하십시오.

- RTO 및 RPO 요구 사항이 엄격하며 기본 데이터베이스와 보조 데이터베이스 간에 지연이 있어서는 안 됩니다. 데이터베이스 미러링은 0초 동기 커밋 포함)의 RPO와 초~분의 RTO를 제공합니다.
- 보조 데이터베이스에서 읽을 필요는 없습니다.
- 미러링 모니터 서버가 동기화 모드로 구성되어 있을 때 자동 장애 조치를 수행하려고 합니다.
- 기본 옵션인 Always On 가용성 그룹은 사용할 수 없습니다.

제한:

- one-to-one 페일오버만 지원됩니다. 여러 데이터베이스 대상을 기본 데이터베이스와 동기화할 수 없습니다.

미러링에 대한 자세한 정보는 [Microsoft SQL Server 설명서](#)를 참조하십시오.

Always On 가용성 그룹

SQL Server Always On 가용성 그룹은 SQL Server 데이터베이스를 위한고가용성 및 재해 복구 솔루션을 제공합니다. 가용성 그룹은 함께 장애 조치되는 사용자 데이터베이스 집합으로 구성됩니다. 여기에는 기본 읽기/쓰기 데이터베이스 세트 하나와 관련된 보조 데이터베이스의 여러 세트(1~8개)가 포함됩니다. 보조 데이터베이스를 애플리케이션 계층에서 기본 데이터베이스의 읽기 전용 복사본으로 사용할 수 있도록 만들어(SQL Server Enterprise 에디션만 해당) 읽기 워크로드를 위한 스케일 아웃 아키텍처를 제공할 수 있습니다. 보조 데이터베이스를 백업 작업에 사용할 수도 있습니다.

SQL Server Always On 가용성 그룹은 동기 및 비동기 커밋 모드를 모두 지원합니다. 동기 모드에서 기본 복제본은 변경 사항이 보조 복제본의 로그에 커밋되거나 기록된 후 데이터베이스 트랜잭션을 커밋합니다. 이 모드를 사용하면 복제본이 동기화된 경우 계획된 수동 장애 조치와 자동 장애 조치를 수행할 수 있습니다. 동일한 환경 내의 SQL Server 인스턴스 간에 동기 커밋 모드를 사용할 수 있습니다(예: 모든 인스턴스가 온프레미스이거나 모든 인스턴스가 온프레미스에 있는 경우). AWS

비동기 커밋 모드에서 기본 복제본은 보조 복제본을 기다리지 않고 데이터베이스 트랜잭션을 커밋합니다. 서로 다른 환경에 있는 SQL Server 인스턴스 간에 비동기 커밋 모드를 사용할 수 있습니다(예: 온프레미스와 내부 인스턴스가 있는 경우). AWS

고가용성 또는 재해 복구를 위해 Always On 가용성 그룹을 사용할 수 있습니다. 다음과 같은 경우 이 방법을 사용하십시오.

- 엄격한 RTO 및 RPO 요구 사항이 있습니다. Always On 가용성 그룹은 초 단위의 RPO와 초에서 분 단위의 RTO를 제공합니다.
- 데이터베이스 그룹을 관리하고 장애 조치하려고 합니다. Always On 가용성 그룹은 SQL Server 2019의 동기 커밋 모드에서 0-4개의 보조 복제본을 지원합니다.
- 동기식 커밋 모드에서 자동 장애 조치를 사용하려는 경우 미러링 모니터 서버가 필요하지 않습니다.
- 보조 데이터베이스에서 읽고 싶습니다.
- 여러 데이터베이스 대상을 기본 데이터베이스와 동기화하려고 합니다.

SQL Server 2016 SP1부터 SQL Server Standard 에디션은 가용성 그룹별로 읽을 수 없는 단일 보조 데이터베이스 및 리스너에 대한 기본적인 고가용성을 제공합니다. 또한 가용성 그룹당 최대 두 개의 노드를 지원합니다.

Always On 장애 조치 클러스터 인스턴스

SQL Server Always On 장애 조치 클러스터 인스턴스(FCI)는 Windows Server 장애 조치 클러스터링(WSFC)을 사용하여 서버 인스턴스 수준에서 고가용성을 제공합니다. FCI는 SQL Server의 전체 설치

에 고가용성을 제공하기 위해 WSFC 노드에 설치되는 SQL Server의 단일 인스턴스입니다. 기본 노드에 하드웨어, 운영 체제, 애플리케이션 또는 서비스 장애가 발생하는 경우 SQL Server 인스턴스 내의 모든 항목이 다른 WSFC 노드로 이동됩니다. 여기에는 시스템 데이터베이스, SQL Server 로그인, SQL Server 에이전트 작업 및 인증서가 포함됩니다.

FCI는 일반적으로 다음과 같은 경우 Always On 가용성 그룹보다 선호됩니다.

- 엔터프라이즈 에디션 대신 SQL Server 스탠다드 에디션을 사용하고 있습니다.
- 인스턴스당 많은 수의 작은 데이터베이스가 있습니다.
- SQL Server 에이전트 작업, 로그인 등과 같은 인스턴스 수준 개체를 지속적으로 수정하고 있습니다.
- SQL Server 인스턴스를 실행하는 데 드는 비용(Amazon EC2 인프라 비용 및 SQL Server 라이선스 비용 모두)을 최적화하고자 합니다. FCI를 사용하면 Amazon Elastic Block Store (Amazon EBS) 처리량에 대한 SQL Server 호스트 EC2 인스턴스의 종속성이 제거됩니다. 따라서 I/O 성능에 영향을 주지 않으면서 SQL Server EC2 인스턴스의 크기를 적절하게 조정하고(즉, 오버프로비저닝할 필요 없이) Amazon EC2 및 SQL Server 라이선스 비용을 줄일 수 있습니다.
- SQL Server 인스턴스의 데이터베이스 성능을 최적화하려고 합니다. FCI를 사용하면 데이터 복제가 데이터베이스 수준이 아닌 블록 수준에서 수행되므로 일반적인 워크로드의 성능이 약 30% 향상됩니다.

FCI에는 복원력과 고가용성을 제공하기 위해 Amazon EBS 프로비저닝된 IOPS SSD (io2) 볼륨, SAN (스토리지 영역 네트워크) 상의 디스크, SMB (서버 메시지 블록) 의 파일 공유 또는 스토리지 스페이스 다이렉트 (S2D), SIOS 데이터키퍼 또는 Amazon FSx를 사용하는 로컬 연결 스토리지 등 특정 형태의 공유 스토리지가 필요합니다.

FCI를 배포할 수 있는 네 AWS가지 옵션이 있습니다.

- Amazon EBS 멀티어태치 (영구 예약 포함)
- Amazon FSx for Windows File Server
- ONTAP용 아마존 NetApp FSx
- 파트너가 제공하는 솔루션 AWS

영구 예약과 함께 Amazon EBS 멀티어태치 사용

[NVMe 예약을 사용하는 Amazon EBS 다중 연결](#)은 Windows Server 장애 조치 클러스터의 공유 스토리지로 Amazon EBS io2 볼륨을 사용하여 SQL Server FCI를 생성할 수 있도록 지원합니다. 이 기능을 사용하면 Amazon EBS 볼륨을 사용하여 장애 조치 클러스터를 구축할 수 있으므로 장애 조치 클러

스터 설정 프로세스가 간소화됩니다. io2 이러한 볼륨은 동일한 가용 영역에 있는 인스턴스에만 연결할 수 있습니다. Amazon EBS io2 볼륨을 사용하여 Windows Server 장애 조치 클러스터를 배포하려면 최신 AWS NVMe 드라이버를 사용해야 합니다.

Amazon EBS 볼륨 및 인스턴스 스토어 볼륨은 [Nitro 기반 인스턴스](#)에서 NVMe 블록 디바이스로 표시됩니다. Amazon EBS io2 볼륨을 사용하여 WSFC 및 SQL Server FCI를 구성할 때 구성된 [SCSI 영구 예약 기능과](#) 함께 [AWS NVMe 드라이버가](#) 설치되어 있어야 합니다.

이 기능에 대한 자세한 내용은 Windows Server에 [Amazon EBS Multi-Attach를 사용하여 SQL Server 장애 조치 클러스터를 배포하는 방법에 대한 AWS](#) 블로그 게시물을 참조하십시오.

Windows File Server용 Amazon FSx 사용

[Windows File Server용 Amazon FSx](#)는 완전 관리형 공유 파일 스토리지를 제공합니다. 두 가용 영역에 스토리지를 동기식으로 자동 복제하여 고가용성을 제공합니다. Windows File Server용 FSx를 파일 스토리지로 사용하면 Amazon EC2에서 SQL Server 고가용성 배포를 단순화하고 최적화하는 데 도움이 됩니다.

Microsoft SQL Server를 사용하면 일반적으로 WSFC의 여러 데이터베이스 노드에 고가용성이 배포되며 각 노드는 공유 파일 스토리지에 액세스할 수 있습니다. Windows File Server용 FSx를 SQL Server 고가용성 배포를 위한 공유 스토리지로 사용할 수 있으며, 두 가지 방법, 즉 활성 데이터 파일을 위한 스토리지로 사용하고 SMB 파일 공유 감시자로 사용할 수 있습니다.

Windows File Server용 FSx를 사용하여 SQL Server FCI 배포 실행의 복잡성과 비용을 줄이는 방법에 대한 자세한 내용은 Windows File Server용 [Amazon FSx를 사용하여 Microsoft SQL Server 고가용성 배포](#)를 단순화하는 블로그 게시물을 참조하십시오. 블로그 게시물에서는 Amazon FSx 다중 AZ 파일 시스템을 공유 스토리지 솔루션으로 사용하여 SQL Server FCI를 배포하는 step-by-step 방법에 대한 지침도 제공합니다. 자세한 내용은 [Windows File Server용 Amazon FSx 설명서](#)를 참조하십시오.

ONTAP용 아마존 NetApp FSx 사용

Amazon FSx NetApp for ONTAP는 ONTAP 파일 시스템에 구축된 안정성과 확장성이 뛰어나고 성능이 뛰어나고 기능이 풍부한 파일 스토리지를 제공하는 완전 관리형 서비스입니다. NetApp FSx for ONTAP는 파일 시스템의 친숙한 기능, 성능, 기능 및 API 작업을 완전 관리형 NetApp 서비스의 민첩성, 확장성 및 단순성과 결합합니다. AWS

FSx for ONTAP는 Windows 및 Linux 시스템용 NFS, SMB 및 iSCSI 프로토콜을 통해 데이터에 대한 멀티 프로토콜 액세스를 제공합니다. ONTAP용 [Amazon FSx를 사용한 SQL Server 고가용성 배포에 자세히 설명된 대로 가용성이 높은 SQL Server Always On FCI](#) 아키텍처를 구축할 수 있습니다. NetApp FSx for ONTAP는 또한 RTO (복구 시간 목표) 및 RPO (복구 시점 목표) 요구 사항을 충족하기 위해

SQL Server 환경을 AWS 리전 다른 환경으로 페일오버하는 빠른 방법을 제공합니다. 자세한 내용은 [ONTAP용 FSx를 사용하여 SQL Server 상시 작동 장애 조치 클러스터 인스턴스용 HA 및 DR 구현](#) 블로그 게시물을 참조하십시오.

Always On 가용성 그룹 및 AWS Launch Wizard 단일 노드 배포를 지원하는 SQL Server 솔루션을 배포하는 데에도 사용할 수 있습니다. AWS Launch Wizard는 ONTAP용 FSx를 공유 스토리지로 사용하여 Amazon EC2의 SQL Server Always on FCI에 배포할 수 있도록 지원합니다. 이 서비스는 복잡한 수동 배포 프로세스를 공유 스토리지를 사용하는 온프레미스 SQL Server 워크로드의 마이그레이션을 가속화하는 안내식 콘솔 기반 마법사로 대체하여 시간과 노력을 절약합니다. Launch Wizard를 사용하여 몇 시간 내에 SQL Server FCI를 프로비저닝하고 구성하는 방법에 대한 자세한 내용은 [AWS Launch Wizard Amazon FSx를 사용하여 SQL Server Always On 배포를 단순화하는](#) 블로그 게시물을 참조하십시오. 또한 Launch Wizard는 [Windows File Server용 Amazon FSx를 공유 스토리지 솔루션으로 사용하여 SQL Server Always On FCI를 배포할 수 있도록](#) 지원합니다.

파트너의 솔루션 사용 AWS

- [DataKeeperSIOS](#)는 가용 영역 전반에 걸쳐 AWS 리전 고가용성 클러스터 페일오버 지원을 제공합니다. DataKeeper SIOS는 에서 사용할 수 있습니다. [AWS Marketplace](#)
- [DxEnterpriseDH2i](#)를 사용하면 쿠버네티스에 있는 SQL Server 가용성 그룹의 완전 자동 페일오버와 Windows 및 Linux용 통합 인스턴스 페일오버가 가능합니다. [AWS Marketplace](#)D2HI는 에서 사용할 수 있습니다.

FSx for Windows File Server

Windows File Server용 FSx는 SMB(서버 메시지 블록) 프로토콜을 사용하여 액세스할 수 있는 안정성과 확장성이 뛰어난 완전 관리형 파일 스토리지를 제공합니다. Windows Server를 기반으로 구축되었으며 사용자 할당량, 최종 사용자 파일 복원 및 Microsoft Active Directory(AD) 통합과 같은 광범위한 관리 기능을 제공합니다. 단일 AZ 및 다중 AZ 배포 옵션, 완전 관리형 백업, 저장된 데이터 및 전송 중인 데이터의 암호화를 제공합니다. 솔리드 스테이트 드라이브(SSD) 및 하드 디스크 드라이브(HDD) 스토리지 옵션을 사용하여 워크로드의 비용 및 성능을 최적화할 수 있으며, 언제든지 스토리지를 확장하고 파일 시스템의 처리 성능을 변경할 수 있습니다. Amazon FSx 파일 스토리지는 AWS온프레미스에서 실행되는 Windows, Linux 컴퓨팅 인스턴스에서 액세스할 수 있습니다.

Amazon FSx는 지속적 가용성(CA) 파일 공유 및 더 작은 파일 시스템을 지원하므로 고가용성 SQL Server 배포를 위한 공유 Windows 스토리지를 더 쉽게 배포할 수 있습니다. 이 옵션은 다음과 같은 사용 사례에 적합합니다.

- WSFC 인스턴스의 SQL Server 노드에서 사용하는 공유 스토리지로 사용됩니다.

- WSFC가 있는 모든 SQL Server 클러스터에서 사용할 수 있는 SMB 파일 공유 감시자로 사용할 수 있습니다.

Amazon FSx는 파일 시스템당 최대 2GB/초의 기준 처리량, 수십만 IOPS, 1밀리초 미만의 일관된 지연 시간으로 빠른 성능을 제공합니다.

SQL 인스턴스에 적합한 성능을 제공하기 위해 파일 시스템 크기와 무관한 처리량 수준을 선택할 수 있습니다. 처리량 용량 수준이 높을수록 파일 서버가 액세스하는 SQL Server 인스턴스에 제공할 수 있는 IOPS 수준도 높아집니다.

스토리지 용량에 따라 저장할 수 있는 데이터의 양뿐만 아니라 스토리지에서 수행할 수 있는 IOPS 수도 결정됩니다. 각 기가바이트의 스토리지는 3IOPS를 제공합니다. 각 파일 시스템의 크기를 최대 64TB까지 프로비저닝할 수 있습니다.

Amazon FSx를 구성하고 사용하여 SQL Server 고가용성 배포의 복잡성과 비용을 줄이는 방법에 대한 자세한 내용은 스토리지 블로그에서 [Windows File Server용 FSx를 사용하여 Microsoft SQL Server 고가용성 배포를 단순화하기](#)를 참조하십시오. AWS 새 CA 공유를 생성하는 방법에 대한 자세한 내용은 [Windows File Server용 FSx 설명서](#)를 참조하십시오.

재해 복구

많은 조직에서 SQL Server 데이터베이스의 고가용성을 구현하지만, 진정한 IT 복원력이 필요한 조직에는 이 정도로는 충분하지 않습니다. 업무상 중요한 데이터베이스의 데이터 손실과 다운타임을 방지하려면 재해 복구 솔루션을 구현하는 것이 좋습니다. SQL Server 배포에 다중 지역 재해 복구 아키텍처를 채택하면 다음과 같은 이점을 얻을 수 있습니다.

- 비즈니스 연속성 달성
- 지리적으로 분산된 고객층의 지연 시간을 개선하세요
- 감사 및 규제 요구 사항 충족

재해 복구 옵션에는 [로그 전송](#), [Always On 가용성 그룹](#), [Amazon S3에 저장되어 여러 AWS 지역에 복제되는 Amazon EBS 스냅샷](#), [Always On 가용성 그룹과 결합된 상시 작동 장애 조치 클러스터 인스턴스 \(FCI\)](#), 분산 가용성 그룹 등이 있습니다.

분산 가용성 그룹

가용성 그룹이 분산된 아키텍처는 다중 지역 SQL Server 배포에 가장 적합한 접근 방식입니다. 분산 가용성 그룹은 두 개의 개별 가용성 그룹에 걸쳐 있는 특수한 유형의 가용성 그룹입니다. 가용성 그룹

으로 구성된 가용성 그룹이라고 생각하시면 됩니다. 기본 가용성 그룹은 서로 다른 두 WSFC 클러스터에 구성됩니다.

분산 가용성 그룹은 느슨하게 결합되어 있으므로 단일 WSFC 클러스터가 필요하지 않고 SQL Server에서 유지 관리합니다. WSFC 클러스터는 개별적으로 유지 관리되고 전송은 주로 두 가용성 그룹 간에 비동기로 이루어지므로 다른 사이트에서 재해 복구를 구성하기가 더 쉽습니다. 각 가용성 그룹의 기본 복제본은 자체 보조 복제본을 동기화합니다.

분산 가용성 그룹은 현재 수동 장애 조치만 지원합니다. 데이터가 손실되지 않도록 글로벌 기본 데이터베이스, 즉 기본 가용성 그룹의 데이터베이스에서 모든 트랜잭션을 중지하십시오. 그런 다음 분산 가용성 그룹을 동기 커밋으로 설정합니다.

SQL Server용 VMware 클라우드 온 AWS

[VMware Cloud AWS on](#)은 VMware가 공동으로 개발한 AWS 통합 클라우드 서비스입니다. SQL Server는 AWS기반 VMware Cloud와 쉽게 통합됩니다. 이 마이그레이션 옵션을 사용하면 가상화에서의 기존 투자를 최대한 활용할 수 있습니다.

시간당, 온디맨드 방식으로 또는 구독 형태로 VMware AWS Cloud에 액세스할 수 있습니다. 여기에는 vSphere Hypervisor(ESXi), Virtual SAN(vSAN), NSX 네트워크 가상화 플랫폼 등 데이터 센터에서 실행하는 것과 동일한 핵심 VMware 기술이 포함되며 SQL Server 데이터베이스를 관리하기 위한 효율적이고 원활한 환경을 제공하도록 설계되었습니다. 몇 분 안에 AWS 기반 VMware Cloud의 SQL Server 데이터베이스의 스토리지, 컴퓨팅 및 메모리를 확장할 수 있습니다.

VMware Cloud on은 물리적 하드웨어에서 직접 AWS 실행되지만 AWS 보안 우선 인프라 모델을 지원하도록 설계된 네트워크 및 하드웨어 기능을 활용합니다. 즉, VMware 가상화 스택은 중첩된 가상화를 사용할 필요 없이 AWS 인프라에서 실행됩니다.

VMware Cloud on을 AWS 사용하면 SQL Server 데이터베이스 워크로드를 쉽게 설정, 확장 및 운영할 수 있습니다. AWS고가용성 솔루션을 제공하고, 온프레미스 Active Directory와 통합되며, AD Connector, Amazon Route 53 CloudWatch, Amazon AWS Directory Service for Microsoft Active Directory 및 Amazon S3와 같은 AWS 서비스에 대한 액세스를 제공합니다. Amazon S3에 백업을 저장하고 재해 복구 프로세스를 현대화하고 단순화할 수 있습니다.

VMware 클라우드 온을 선택해야 하는 경우 AWS

VMware 클라우드 AWS 온은 다음과 같은 경우에 SQL Server 데이터베이스를 위한 옵션입니다.

- SQL Server 데이터베이스는 vSphere 가상화 환경의 온프레미스 데이터 센터에서 이미 실행되고 있습니다.

- 데이터베이스 수가 많고 마이그레이션 팀의 추가 작업 없이 다음과 같은 이유 중 하나로 클라우드로 빠르게 마이그레이션(예: 몇 시간)해야 합니다.
- 데이터 센터 확장. 가상 데스크톱을 실행하거나, 애플리케이션을 게시하거나, 개발/테스트 환경을 제공하려면 온디맨드 용량이 필요합니다.
- 재해 복구. 새 재해 복구 시스템을 설치하거나 기존 시스템을 교체하려고 합니다.
- 클라우드 마이그레이션. 전체 데이터 센터를 클라우드로 마이그레이션하거나 인프라를 새로 고치고 싶습니다.

SQL Server 데이터베이스에 80K 이상의 IOPS가 필요한 경우 vSAN을 사용할 수 있습니다.

자세한 내용은 AWS 뉴스 블로그의 [작업 중 — VMware Cloud를 참조하고 웹 사이트의 AWS VMware AWS Cloud에 Microsoft SQL Server 배포를 참조하십시오.](#) AWS

SQL Server를 위한 이기종 데이터베이스 마이그레이션

오픈 소스 데이터베이스 및 AWS와 같은 클라우드 컴퓨팅 플랫폼의 혁신과 개선으로 인해 많은 조직이 SQL Server와 같은 독점(온라인 트랜잭션 처리 또는 OLTP) 데이터베이스 엔진에서 오픈 소스 엔진으로 전환하고 있습니다. SQL Server 데이터베이스는 모든 조직의 업무상 중요한 시스템이지만 특정 공급업체에 종속되는 것은 위험하고 비용이 많이 드는 상황입니다. 기본 데이터베이스 기술을 오픈 소스 또는 AWS 클라우드 네이티브 데이터베이스로 전환하는 것을 고려해야 하는 매력적인 이유는 낮은 운영 비용과 무료 라이선스 비용입니다.

SQL Server에서 마이그레이션하는 다른 이유로는 공급업체 종속 기간, 라이선스 감사, 값비싼 라이선스, 비용 등이 있습니다. 이러한 이유로 많은 조직에서는 AWS로 마이그레이션할 때 SQL Server 데이터베이스를 오픈 소스 데이터베이스(예: PostgreSQL, MySQL 또는 MariaDB) 또는 AWS 클라우드 네이티브 데이터베이스(예: Amazon Aurora 또는 Amazon DynamoDB)로 마이그레이션하기로 선택합니다.

또한 SQL Server 데이터 웨어하우스 데이터베이스를 빠르고 완벽하게 관리되는 클라우드 데이터 웨어하우스인 Amazon Redshift로 마이그레이션할 수 있습니다. Amazon Redshift는 데이터 레이크와 통합되어 다른 데이터 웨어하우스보다 최대 3배 빠른 성능을 제공하며 다른 클라우드 데이터 웨어하우스보다 최대 75% 저렴한 비용을 제공합니다. 자세한 내용은 AWS 권장 가이드 웹 사이트의 [AWS DMS를 사용하여 온프레미스 Microsoft SQL Server 데이터베이스를 Amazon Redshift로 마이그레이션하는 패턴](#)을 참조하세요.

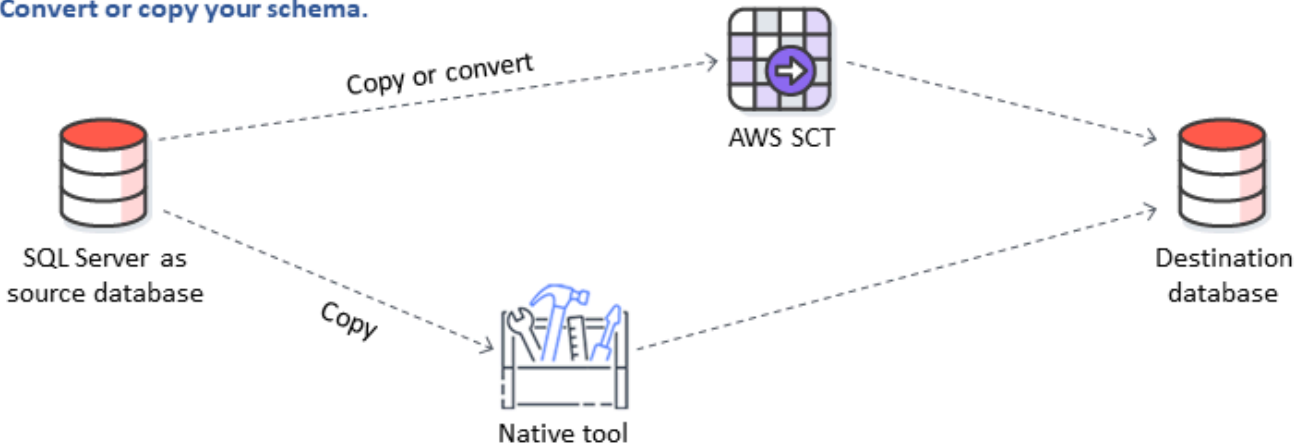
오픈 소스 또는 AWS 클라우드 네이티브 데이터베이스로 마이그레이션하려면 보유한 데이터 유형, 액세스 모델, 확장성, 애플리케이션 실용성 및 복잡성에 따라 적절한 데이터베이스를 선택하십시오. SQL Server에서 PostgreSQL 및 기타 오픈 소스 데이터베이스로 마이그레이션하는 작업은 종종 어렵고 시간이 많이 걸리며 신중한 평가, 계획 및 테스트가 필요합니다.

가동 중지 시간을 최소화하면서 상용 데이터베이스를 AWS의 오픈 소스 데이터베이스로 마이그레이션하는 데 도움이 되는 AWS Database Migration Service(AWS DMS) 및 AWS Schema Conversion Tool(AWS SCT) 같은 서비스를 사용하면 이 프로세스가 더 쉬워집니다.

이기종 데이터베이스 마이그레이션에서는 SQL Server에서 Aurora로 또는 SQL Server에서 MariaDB로 마이그레이션하는 경우와 같이 소스와 대상 데이터베이스 엔진이 다릅니다. 소스와 대상 데이터베이스의 스키마 구조, 데이터 유형 및 데이터베이스 코드는 매우 다를 수 있으므로 데이터 마이그레이션을 시작하기 전에 스키마와 코드를 변환해야 합니다. 이러한 이유로 이기종 마이그레이션은 다음과 같은 2단계 프로세스입니다.

- 1단계. 소스 스키마와 코드를 대상 데이터베이스의 것과 일치하도록 변환하십시오. 이 변환에 AWS SCT를 사용할 수 있습니다.
- 2단계. 원본 데이터베이스의 데이터를 원본 데이터베이스에서 대상 데이터베이스로 마이그레이션합니다. 이 프로세스에 AWS DMS를 사용할 수 있습니다.

1 Convert or copy your schema.



2 Copy your data.



AWS DMS은 마이그레이션 중에 주요 데이터 유형 변환을 자동으로 처리합니다. 원본 데이터베이스는 AWS 외부의 자체 프리미스에 있거나, EC2 인스턴스에서 실행되는 데이터베이스이거나, Amazon RDS 데이터베이스일 수 있습니다(AWS DMS 설명서의 [데이터 마이그레이션 소스](#) 참조). 대상은 Amazon EC2, Amazon RDS 또는 Aurora에 있는 데이터베이스일 수 있습니다. MySQL을 대상 데이터베이스로 사용하는 방법에 대한 자세한 내용은 AWS 데이터베이스 블로그의 [MySQL 호환 데이터베이스 엔진으로 SQL Server 데이터베이스 마이그레이션](#)을 참조하십시오.

AWS 기반 SQL Server 데이터베이스를 리팩터링하는 방법에 대한 자세한 내용은 AWS 권장 가이드 웹 사이트의 [재설계 패턴](#)을 참조하십시오.

이기종 데이터베이스 마이그레이션을 위한 도구

다음 차트는 SQL Server에서 다른 데이터베이스 엔진으로 마이그레이션하는 데 사용할 수 있는 도구 목록을 제공합니다.

마이그레이션 도구	대상 데이터베이스 지원	사용 대상
AWS SCT	Amazon RDS for MySQL Amazon RDS for PostgreSQL Amazon Aurora MySQL Amazon Aurora PostgreSQL	스키마 변환
AWS DMS	Amazon RDS for MySQL Amazon RDS for PostgreSQL Amazon Aurora MySQL Amazon Aurora PostgreSQL	데이터 마이그레이션
Babelfish	Amazon Aurora PostgreSQL	데이터 액세스 및 마이그레이션

다음 하위 섹션에서는 각 도구에 대한 자세한 정보를 제공합니다.

AWS SCT

[AWS Schema Conversion Tool\(AWS SCT\)](#)는 기존의 상용 데이터베이스 스키마를 오픈 소스 엔진 또는 AWS 클라우드 네이티브 데이터베이스로 변환합니다. AWS SCT는 소스 데이터베이스 스키마와 대부분의 데이터베이스 코드 개체(뷰, 저장된 프로시저, 함수 등)를 대상 데이터베이스와 호환되는 형식으로 자동 변환하여 이기종 데이터베이스 마이그레이션을 예측할 수 있도록 합니다.

한 엔진에서 다른 엔진으로 데이터베이스 스키마를 변환할 때는 이전 데이터베이스 엔진 대신 새 데이터베이스 엔진과 상호 작용하도록 응용 프로그램의 SQL 코드도 업데이트해야 합니다. 또한 AWS SCT은 C++, C#, Java 또는 기타 애플리케이션 코드의 SQL 코드를 변환합니다. 자동으로 변환할 수 없는 모든 객체는 수동 변환용으로 명확하게 표시됩니다. 또한 AWS SCT은 애플리케이션 소스 코드에서 내

장된 SQL 문을 스캔하여 데이터베이스 스키마 변환 프로젝트의 일부로 변환할 수 있습니다. 자세한 내용은 AWS 설명서의 [AWS SCT에 대한 소스로 Microsoft SQL Server 사용](#)을 참조하세요.

AWS DMS

[AWS Database Migration Service\(AWS DMS\)](#)는 데이터를 빠르고 안전하게 AWS로 마이그레이션합니다. 마이그레이션하는 동안 원본 데이터베이스는 완벽하게 작동하여 애플리케이션 다운타임을 최소화합니다. AWS DMS은 하나의 SQL Server 데이터베이스에서 다른 SQL Server 데이터베이스로 데이터를 마이그레이션하는 것과 같은 동종 마이그레이션을 지원합니다. 또한 SQL Server 데이터베이스를 오픈 소스 데이터베이스 또는 AWS 클라우드 네이티브 데이터베이스로 마이그레이션하는 등 서로 다른 데이터베이스 플랫폼 간의 이기종 마이그레이션도 지원합니다. AWS DMS는 원본 데이터베이스에서 발생하는 데이터 변경 사항을 대상 데이터베이스에 자동으로 복제하는 등 마이그레이션 프로세스의 복잡성을 관리합니다. 데이터베이스 마이그레이션이 완료된 후 대상 데이터베이스는 사용자가 선택하는 기간 동안 원본과 동기 상태를 유지합니다. 따라서 데이터베이스를 편리한 시점에 전환할 수 있습니다. 자세한 내용은 AWS 설명서의 [Microsoft SQL Server 데이터베이스를 AWS DMS에 대한 소스로 사용](#)을 참조하십시오.

Babelfish

Babelfish는 Amazon Aurora에 내장된 기능입니다. Babelfish for Aurora PostgreSQL를 사용하면 Aurora PostgreSQL 호환 버전 데이터베이스에서 Microsoft SQL Server용으로 작성된 애플리케이션의 명령을 이해할 수 있습니다. SQL Server의 전용 SQL 언어인 Transact-SQL(T-SQL)로 작성된 SQL Server 데이터베이스 코드가 있는 SQL Server 애플리케이션을 수정하려면 많은 노력이 필요하고 시간이 많이 걸립니다. [Babelfish for Aurora PostgreSQL](#)를 사용하면 이 프로세스를 더 간단하고 쉽게 수행할 수 있습니다. Babelfish를 사용하면 애플리케이션 코드를 변경할 필요가 없습니다. 대신 Babelfish for Aurora PostgreSQL를 사용하여 SQL Server 데이터베이스를 Aurora PostgreSQL 호환 DB 클러스터로 마이그레이션할 수 있습니다.

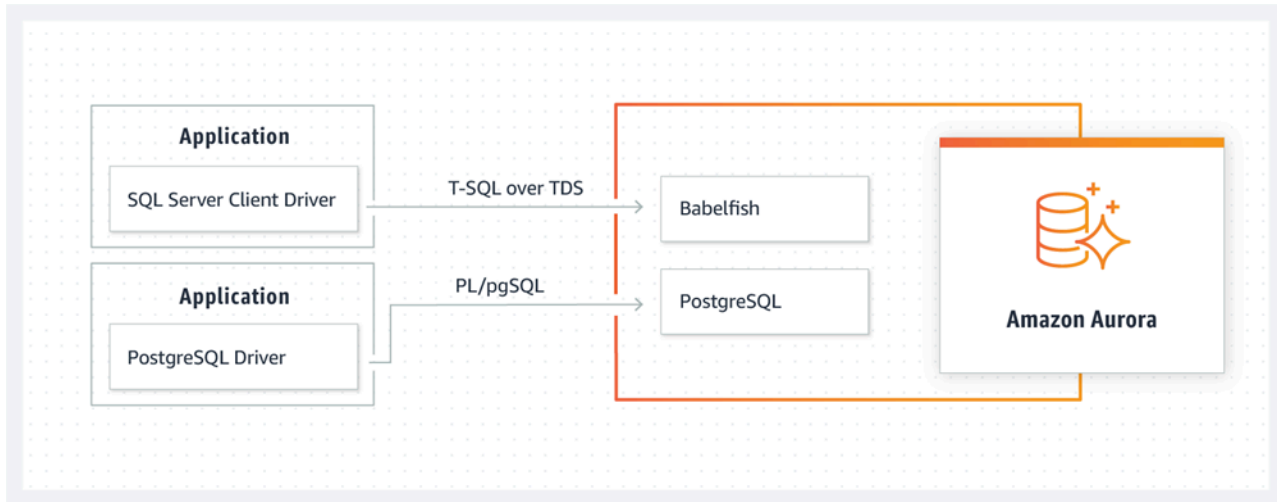
Babelfish를 사용하면 Aurora PostgreSQL이 T-SQL을 이해하고 동일한 통신 프로토콜을 지원하므로 데이터베이스 드라이버를 전환하거나 애플리케이션 쿼리를 다시 작성할 필요가 없습니다. 원래 SQL Server용으로 작성된 애플리케이션을 이제 코드 변경 횟수를 줄이면서 Aurora와 연동할 수 있습니다. 이를 통해 SQL Server 또는 최신 버전에서 실행되는 애플리케이션을 수정하고 Aurora로 이동하는 데 필요한 노력이 줄어들어 마이그레이션 속도가 빨라지고 위험도가 낮으며 비용 효율적으로 마이그레이션할 수 있습니다.

기존 SQL Server 데이터베이스에서 마이그레이션하는 경우 기본 PostgreSQL API를 사용하여 구축한 새로운 기능과 함께 Babelfish를 사용하여 SQL Server 코드를 실행할 수 있습니다. Babelfish는 Aurora PostgreSQL가 일반적으로 사용되는 SQL Server 도구, 명령 및 드라이버와 함께 작동할 수 있도록 합니다.

또한 Babelfish는 기본 PostgreSQL 연결을 사용하여 데이터에 대한 액세스를 제공합니다. 기본적으로 Babelfish에서 지원하는 두 SQL 언어는 다음 포트에서 기본 와이어 프로토콜을 통해 사용할 수 있습니다.

- SQL Server 언어(T-SQL)의 경우 포트 1433에 연결합니다.
- PostgreSQL 언어(PL/pgSQL)의 경우 포트 5432에 연결합니다.

Babelfish를 사용하면 SQL Server 또는 PostgreSQL 포트에서 연결을 제공하여 광범위한 코드 재작성 없이 기존 SQL Server 애플리케이션이 Aurora와 통신할 수 있습니다. 다음은 이러한 아키텍처를 나타낸 다이어그램입니다.



Amazon RDS 관리 콘솔을 사용하여 Aurora 클러스터에서 Babelfish를 활성화할 수 있습니다. 자세한 지침은 Amazon RDS 설명서의 [Babelfish for Aurora PostgreSQL DB 클러스터 생성](#)을 참조하십시오.

마이그레이션에 대한 자세한 내용은 Aurora 설명서의 [SQL Server 데이터베이스를 Babelfish for Aurora PostgreSQL로 마이그레이션하기](#)를 참조하십시오.

자세한 내용은 다음 리소스를 참조하세요.

- [Babelfish for Aurora PostgreSQL 시작하기](#)(AWS 데이터베이스 블로그)
- [Babelfish를 사용하여 SQL Server에서 Amazon Aurora로 마이그레이션](#)(AWS 데이터베이스 블로그)
- [SSIS 및 Babelfish를 사용하여 SQL Server에서 Aurora PostgreSQL로 마이그레이션](#)(AWS 데이터베이스 블로그)
- [SQL Server에서 Babelfish for Aurora PostgreSQL로 SSIS 패키지를 수정](#)(AWS 데이터베이스 블로그)

-
- [Babelfish for Aurora PostgreSQL에 대한 SQL Server 보고 서비스 보고서 실행\(AWS 데이터베이스 블로그\)](#)
 - [AWS SCT 평가 보고서로 Babelfish 마이그레이션 준비\(AWS 데이터베이스 블로그\)](#)

SQL Server 하이브리드 마이그레이션 시나리오

AWS를 포함하는 하이브리드 환경에서 SQL Server 워크로드를 실행할 수도 있습니다. 예를 들어 온프레미스 또는 같은 위치에 있는 데이터 센터에서 이미 SQL Server를 실행하고 있지만 AWS 클라우드를 사용하여 아키텍처를 개선하여 고가용성 또는 재해 복구 솔루션을 제공하려고 할 수 있습니다. 또한 하이브리드 솔루션을 사용하여 AWS에 장기 SQL Server 백업을 저장하거나, 문제 발생 시 마이그레이션을 롤백하거나, AWS 클라우드의 SQL Server Always On 가용성 그룹을 사용하여 보조 복제본을 실행할 수 있습니다. SQL Server에는 고가용성 및 재해 복구 솔루션을 제공하는 여러 복제 기술이 있습니다.

SQL Server 데이터베이스를 AWS 클라우드에 백업

Amazon Simple Storage Service(Amazon S3)를 사용하면 클라우드 스토리지의 유연성과 요금을 활용할 수 있습니다. 이를 통해 SQL Server 데이터베이스를 안전하고 가용성이 높으며 내구성이 뛰어나고 안정적인 스토리지 시스템에 백업할 수 있습니다. Amazon S3에 SQL Server 백업을 안전하게 저장할 수 있습니다. 또한 Amazon S3 수명 주기 정책을 사용하여 백업을 장기간 저장할 수도 있습니다. Amazon S3를 사용하면 매우 저렴한 비용으로 대량의 데이터를 저장할 수 있습니다. [AWS DataSync](#)를 사용하여 백업 파일을 Amazon S3로 전송할 수 있습니다.

Storage Gateway를 사용하여 온프레미스 SQL Server 백업을 저장하고 Amazon S3 또는 Amazon S3 Glacier에 데이터를 아카이빙할 수 있습니다. 캐시된 스토리지 볼륨을 생성하여 온프레미스 백업 애플리케이션 서버에서 Small Computer System Interface(iSCSI) 디바이스로 마운트할 수 있습니다. 모든 데이터는 SSL을 통해 AWS로 안전하게 전송되며 Amazon S3에 암호화된 형식으로 저장됩니다. 게이트웨이 캐싱 볼륨을 사용하면 비용이 많이 드는 스토리지 하드웨어를 온프레미스로 유지 관리하고 확장하는 데 드는 초기 비용을 절감할 수 있습니다. 기본 데이터 또는 백업을 온프레미스로 보관하려는 경우 게이트웨이 저장 볼륨을 사용하여 해당 데이터를 로컬에 보관하고 데이터를 오프사이트로 Amazon S3에 백업할 수 있습니다.

고가용성 및 재해 복구 솔루션 확장

SQL Server의 기본 로그 전달 기능을 사용하여 기존 온프레미스 고가용성 사례를 확장하고 AWS에서 재해 복구 솔루션을 제공할 수 있습니다. SQL Server 트랜잭션 로그를 온프레미스 또는 같은 위치에 있는 데이터 센터에서 EC2 인스턴스 또는 Virtual Private Cloud(VPC)의 Amazon RDS for SQL Server DB 인스턴스에서 실행 중인 SQL Server 인스턴스로 전송할 수 있습니다. AWS Direct Connect를 사용하여 전용 네트워크 연결을 통해 해당 데이터를 안전하게 전송하거나 보안 VPN 터널을 통해 전송할 수 있습니다. 트랜잭션 로그 백업은 EC2 인스턴스로 전송되며 보조 데이터베이스 인스턴스에 적용됩니다.

AWS 클라우드를 사용하면 온프레미스 데이터 센터와 Amazon EC2 간의 SQL Server Always On 가용성 그룹을 사용하여 더 높은 수준의 고가용성과 재해 복구를 제공할 수 있습니다. 이는 AWS Direct Connect와 같은 전용 네트워크 연결을 사용하거나 이 두 환경 간에 보안 VPN 터널을 설정하여 데이터 센터를 AWS의 VPC로 확장함으로써 수행할 수 있습니다.

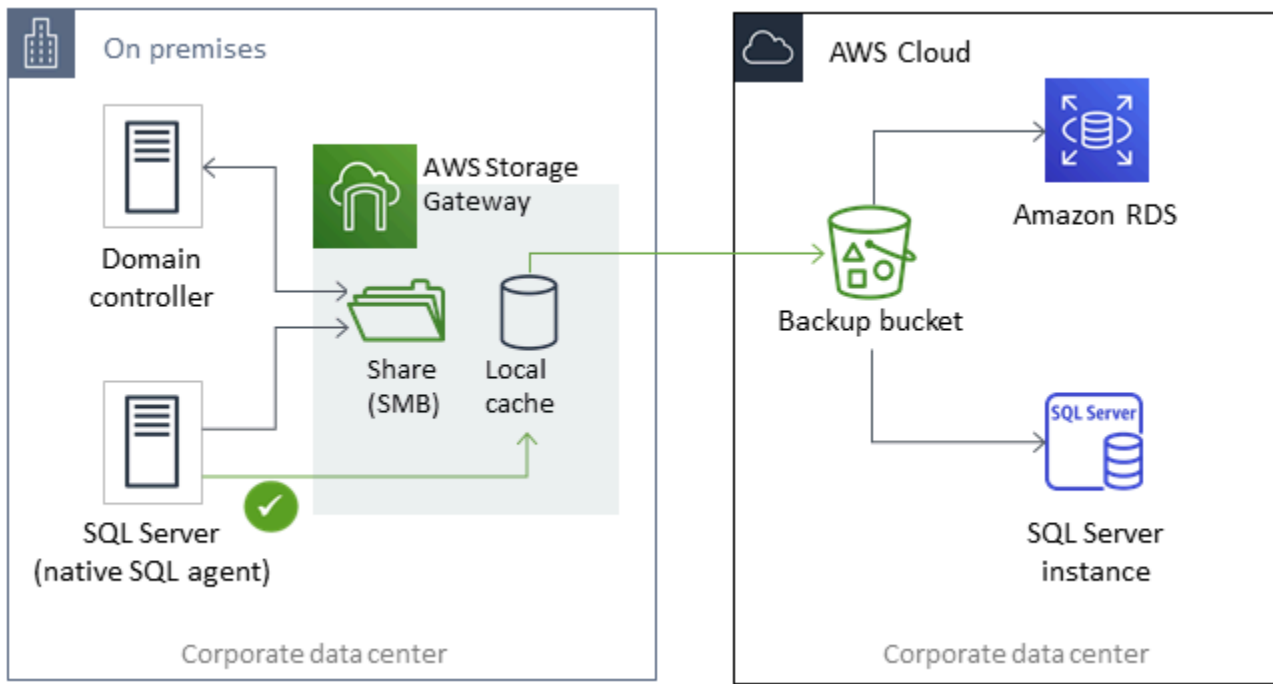
SQL Server Always On 가용성 그룹의 하이브리드 구현을 계획할 때 고려해야 할 몇 가지 사항은 다음과 같습니다.

- AWS Direct Connect 또는 VPN을 통해 온프레미스 환경과 AWS 간에 안전하고 안정적이며 일관된 네트워크 연결을 구축하세요.
- Amazon Virtual Private Cloud(VPC) 서비스를 사용하여 VPC를 생성하세요. Amazon VPC 라우팅 테이블과 보안 그룹을 사용하여 두 환경 간에 적절한 통신이 가능하도록 하세요.
- 도메인 컨트롤러를 EC2 인스턴스로 배포하거나 AWS Directory Service for Microsoft Active Directory를 사용하여 Active Directory 도메인을 VPC로 확장하세요. Amazon RDS for SQL Server에 AWS Managed Microsoft AD를 사용할 수도 있습니다. 자세한 내용은 [Amazon RDS 설명서](#)를 참조하세요.

Storage Gateway

Storage Gateway를 사용하면 Windows용 Server Message Block(SMB) 공유를 사용하여 파일을 저장하고 검색할 수 있습니다. 스토리지 게이트웨이를 온프레미스 Active Directory 도메인에 조인할 수 있습니다. SQL Server 데이터베이스와 스토리지 게이트웨이를 동일한 도메인에 두면 백업을 로컬에 저장한 다음 네트워크 공유에 업로드하는 대신 SMB 네트워크 공유로 직접 가져올 수 있습니다. 스토리지 게이트웨이는 S3 버킷을 사용하도록 구성되어 있으므로 모든 백업을 AWS의 S3 버킷에서 사용할 수 있습니다. 백업 파일을 EC2 인스턴스의 SQL Server로 다운로드하여 데이터베이스를 복원하거나 Amazon RDS에 데이터베이스를 직접 복원할 수 있습니다.

다음 다이어그램은 Storage Gateway와 Amazon S3를 사용하여 백업을 저장하고 액세스하는 방법을 보여줍니다. 자세한 내용은 [Storage Gateway 설명서](#)를 참조하세요.



AWS DMS 및 AWS SCT 사용

하이브리드 SQL Server 환경에서 AWS DMS를 사용하여 온프레미스 데이터베이스의 데이터를 클라우드로 마이그레이션하거나 그 반대로 마이그레이션할 수 있습니다. AWS SCT와 함께 AWS DMS를 사용하여 SQL Server 데이터베이스를 MySQL 또는 PostgreSQL로 마이그레이션할 수 있습니다. 마이그레이션 단계는 [AWS SCT 설명서](#)를 참조하세요. 데이터를 마이그레이션하기 전에 필요할 수 있는 추가 수동 작업에 플래그를 지정하는 [마이그레이션 평가 보고서](#)를 실행할 수 있습니다.

지속적인 복제(변경 데이터 캡처 또는 CDC)에도 AWS DMS를 사용할 수 있습니다. 자세한 내용은 AWS DMS 설명서의 [SQL Server 소스에서 지속적 복제\(CDC\) 사용](#)을 참조하세요.

SQL Server 데이터베이스 현대화

이 섹션에서는 Windows 운영 체제에서 Linux로 전환하여 AWS에서 SQL Server 워크로드를 현대화하는 방법에 대해 설명합니다. 이러한 변경을 통해 시스템 아키텍처를 크게 변경하거나 사용자를 재교육하지 않고도 오픈 소스 기술을 활용하고 Windows 라이선스 비용을 절감할 수 있습니다.

SQL Server 워크로드를 Windows에서 Linux로 마이그레이션

SQL Server 2017부터 SQL Server를 Linux 운영 체제에서 실행할 수 있습니다. SQL Server 워크로드를 Linux로 이전하면 비용을 절감하고 성능을 개선할 수 있습니다.

Microsoft Windows에서 사용하는 거의 모든 SQL Server 함수, 응용 프로그램, 명령문 및 스크립트는 Linux에서도 지원됩니다. 또한 SQL 서버 관리 스튜디오 (SSMS), SQL 서버 데이터 도구 (SSDT) 및 PowerShell 모듈 (sqlps) 과 같은 도구를 사용하여 Windows 인스턴스에서 Linux의 SQL 서버를 관리할 수 있습니다.

다음 세 가지 옵션 중 하나를 사용하여 SQL Server 워크로드를 Linux로 마이그레이션할 수 있습니다.

- 기본 SQL Server 백업 및 복원 기능 ([Microsoft SQL Server 설명서](#) 참조)
- 분산 가용성 그룹(AWS(으)로 마이그레이션하는 동안 운영 체제를 변경하기 위해)
- PowerShell 기반 스크립팅 도구인 AWS 리플랫폼 어시스턴트

AWS 리플랫폼 어시스턴트는 기존 SQL Server 워크로드를 Windows에서 Linux 운영 체제로 마이그레이션할 수 있도록 도와줍니다. 소스 SQL Server 데이터베이스에서 리플랫폼 어시스턴트를 위한 PowerShell 스크립트를 실행하면 Windows 인스턴스가 데이터베이스를 암호화된 Amazon S3 스토리지 버킷에 백업합니다. 그런 다음 백업을 EC2 Linux 인스턴스의 새 SQL Server 데이터베이스 또는 기존 SQL Server 데이터베이스로 복원합니다. 원본 SQL Server 데이터베이스가 온라인 상태인 상태에서 데이터베이스를 복제하고 애플리케이션을 테스트할 수 있습니다. 테스트가 끝나면 애플리케이션 가동 중단 시간을 예약하고 PowerShell 백업 스크립트를 다시 실행하여 최종 전환을 수행할 수 있습니다.

리플랫폼 어시스턴트를 사용하는 방법에 대한 자세한 내용은 AWS 데이터베이스 블로그의 [온프레미스 SQL Server Windows 워크로드를 Amazon EC2 Linux로 마이그레이션](#)하고 [Amazon EC2 설명서](#)를 참조하세요.

Linux의 고가용성

SQL Server 2017은 Windows와 Linux 간의 Always On 가용성 그룹을 지원하므로 고가용성 없이 읽기 규모의 워크로드를 생성할 수 있습니다. 안타깝게도 크로스 플랫폼 구성을 관리할 수 있는 클러스터 솔루션이 없기 때문에 Windows와 Linux 간에는 고가용성을 달성할 수 없습니다.

Always On 가용성 그룹에서 고가용성을 사용하려면 Windows Server 장애 조치 클러스터 (WSFC) 또는 Linux용 Pacemaker를 사용하는 것이 좋습니다. 이 솔루션은 Windows용 SQL Server에서 Linux로 또는 Linux에서 Linux로 또는 그 반대로 마이그레이션하거나 수동 장애 조치를 사용한 재해 복구에 적합합니다. 이 시나리오에 대한 자세한 내용은 AWS 데이터베이스 블로그의 [Amazon EC2 Windows와 Amazon Linux 2 인스턴스 간의 Always On 가용성 그룹 배포를](#) 참조하세요.

SQL 서버용 AWS Launch Wizard

AWS Launch Wizard는 Amazon EC2에서 Microsoft SQL Server의 크기 조정, 구성 및 배포 과정을 안내하는 서비스입니다. SQL Server 단일 인스턴스와 Amazon EC2에서의 고가용성 (HA) 배포를 모두 지원합니다.

Launch Wizard는 무료 서비스입니다. Amazon EC2, Amazon EBS, Amazon VPC AWS 리소스와 같이 애플리케이션을 실행하기 위해 프로비저닝된 리소스에 대해서만 비용을 지불합니다.

Launch Wizard 콘솔에서 성능, 노드 수, 연결을 비롯한 애플리케이션 요구 사항을 입력하면 됩니다. Launch Wizard는 SQL Server 응용 프로그램을 배포하고 실행하는 데 적합한 AWS 리소스를 식별합니다. 또한 예상 배포 비용도 제공되므로 리소스를 수정하고 업데이트된 비용 평가를 즉시 확인할 수 있습니다. 선택을 확인하고 배포를 시작하면 Launch Wizard가 몇 시간 내에 선택한 리소스를 프로비저닝 및 구성하여 완벽하게 작동하고 프로덕션에 바로 사용할 수 있는 SQL Server 응용 프로그램을 만듭니다. Amazon EC2 콘솔에서 배포된 SQL Server 애플리케이션에 액세스할 수 있습니다.

SQL Server용 Launch Wizard를 사용하여 얻을 수 있는 몇 가지 이점은 다음과 같습니다.

- 간단한 배포 — 요구 사항에 따라 질문에 답하여 AWS에서 SQL Server 리소스의 프로비저닝을 단순화할 수 있습니다. Launch Wizard 배포는 수동 배포보다 빠르므로 AWS에서 애플리케이션을 프로비저닝하고 구성하는 데 시간을 들이지 않아도 됩니다.
- 자동 크기 조정 및 비용 추정 — Launch Wizard는 요구 사항에 따라 기본 제공되는 인스턴스 선택 기능을 제공합니다. SQL Server 요구 사항에 가장 적합한 인스턴스 유형, EBS 볼륨 및 기타 리소스를 선택합니다. Launch Wizard는 AWS 리소스를 프로비저닝하기 전에 예상 비용도 제공합니다.
- 반복 가능한 자동화 템플릿으로 시간 절약 — Launch Wizard에서 만든 재사용 가능한 AWS CloudFormation 템플릿을 사용하여 SQL Server를 재배포할 수 있습니다. 이러한 템플릿은 기존 역할을 하므로 시간을 절약할 수 있습니다.

Launch Wizard는 다음 운영 체제, SQL Server 버전 및 기능을 지원합니다. 최신 정보는 [AWS Launch Wizard 설명서](#)를 참조하세요.

범주	사용 사례	Launch Wizard 지원	퀵 스타트 지원	Amazon EC2 콘솔 지원
윈도우에 배포	단일 SQL 노드 배포		 0: 필요	
	HA 배포: 상시 작동 가용성 그룹			 0: 필요
	HA 배포: Windows File Server용 FSx를 사용하는 올웨이즈 온 페일오버 클러스터 인스턴스 (FCI)			 0: 필요
	HA 배포: 전용 호스트			 0: 필요
	재사용 가능한 코드 템플릿			 0: 필요
	Linux에 배포	단일 SQL 노드 배포		 0: 필요

범주	사용 사례	Launch Wizard 지원	퀵 스타트 지원	Amazon EC2 콘솔 지원	
	Ubuntu에 HA 배포		0:  니요	0:  니요	아
	재사용 가능한 코드 템플릿		0:  니요	0:  니요	아
크기 조정	인스턴스 유형 권장 사항		0:  니요	0:  니요	아
	비용 추정		0:  니요	0:  니요	아
구성	자동으로 생성된 AWS Systems Manager 리소스 그룹		0:  니요	0:  니요	아
	원클릭 Amazon SNS 알림		0:  니요	0:  니요	아
	원클릭 Amazon CloudWatch 모니터링		0:  니요	0:  니요	아

범주	사용 사례	Launch Wizard 지원	퀵 스타트 지원	Amazon EC2 콘솔 지원
	기존 액티브 디렉터리에 연결 (온프레미스 및 관리형 모두)		 니요	 니요
	조기 입력 유효성 검사		 니요	 니요
	관리형 IAM 정책		 니요	 니요

SQL Server용 Launch Wizard에 대한 자세한 내용은 다음 사항을 참조하세요.

- [SQL 서버용 AWS Launch Wizard 설명서](#)
- [AWS Launch Wizard와 Amazon FSx 블로그 게시물을 사용하여 SQL Server Always On 배포를 간소화합니다](#)
- [AWS Launch Wizard를 통한 SQL Server 올웨이즈 온 배포 가속화 블로그 게시물](#)

Amazon RDS for SQL Server로 마이그레이션하기 위한 모범 사례

데이터베이스 평가 및 프로젝트 요건을 바탕으로 Amazon RDS for SQL Server로 마이그레이션하는 것이 목표라면, 이 섹션의 모범 사례에 따라 대상 데이터베이스를 제공하고, 마이그레이션을 수행하고, Amazon RDS for SQL Server 데이터베이스를 테스트, 운영 및 최적화하십시오.

Important

데이터베이스를 마이그레이션하기 전에 롤백 계획이 있는지 확인하세요.

Note

Migration Hub Orchestrator를 사용하면 기본 백업 및 복원을 사용하여 Amazon EC2 또는 Amazon RDS로의 SQL Server 데이터베이스 마이그레이션을 자동화하고 조정할 수 있습니다. 자세한 설명은 [AWS Migration Hub Orchestrator 섹션](#)을 참조하세요.

대상 데이터베이스 프로비저닝

데이터베이스 마이그레이션 전략의 평가, 계획 및 준비를 마친 후에는 Amazon RDS for SQL Server 데이터베이스를 프로비저닝할 때 다음 모범 사례를 따르십시오.

- CPU, 메모리, IOPS 및 스토리지 타입에 대한 요건에 근거한 Amazon RDS for SQL Server DB 인스턴스의 크기를 적절하게 조정합니다. (SQL Server 표준판을 사용하는 경우, 표준판의 제한 내에서 CPU와 메모리를 프로비저닝하십시오.)
- 시간대와 데이터 정렬을 올바르게 설정하십시오.
- 올바른 Virtual Private Cloud(VPC)에서 Amazon RDS를 시작해야 합니다.
- 올바른 포트와 IP 주소를 사용하여 보안 그룹을 생성하십시오.
- Amazon RDS 데이터베이스를 프라이빗 서브넷에 프로비저닝하여 보안을 유지합니다.
- 가능하면 SQL Server 인스턴스를 최신 버전의 SQL Server로 프로비저닝하십시오.
- 각 Amazon RDS 데이터베이스에 대해 별도의 옵션 그룹과 파라미터 그룹을 생성하십시오.
- 마이그레이션을 위한 로그인, 사용자, 역할을 수집하고 추출합니다.

- 마이그레이션해야 하는 응용 프로그램과 준비에 대한 SQL Server 에이전트 작업을 검토하십시오.

소스 데이터베이스에서 백업

SQL Server 데이터베이스를 Amazon RDS for SQL Server용 데이터베이스로 마이그레이션하기 위한 많은 도구가 있습니다. 요건에 따라 가동 중단이 허용되는 경우, 가장 일반적으로 사용되는 방법은 SQL Server 기본 백업 및 복원을 사용하는 것입니다.

가동 중단 시간이 제한된 경우, 기본 SQL Server 백업/복원을 차등 백업 및 로그 백업과 함께 사용할 수 있습니다. 또는 전체 로드, 전체 로드 및 CDC 또는 CDC 전용의 세 가지 옵션을 제공하는 AWS DMS를 사용할 수도 있습니다.

AWS에 데이터 덤프 파일 전송

- 온프레미스 환경과 AWS 간에 고대역폭 연결을 제공하는 AWS Direct Connect를 사용하는 경우, SQL Server 백업을 Amazon S3로 복사하고 [Amazon S3 통합](#)을 설정할 수 있습니다.
- AWS Direct Connect를 통과하는 대역폭이 높지 않은 경우, 대용량 데이터베이스 백업 파일을 전송하는 데 AWS Snowball을 사용하십시오. 복제가 필요한 경우, AWS DMS를 사용하여 데이터를 전송할 수도 있습니다.

대상 데이터베이스로 데이터 복원

- 초대형 데이터베이스를 마이그레이션하는 경우, 데이터 로드 속도를 높이려면 마이그레이션 기간 동안 처음에 더 큰 [Amazon RDS 인스턴스 타입](#)을 프로비저닝하는 것이 좋습니다.
- 다AZ 비활성화. (마이그레이션 후 다시 활성화할 수 있습니다.)
- 백업 보존을 비활성화합니다. (마이그레이션 후 다시 활성화할 수 있습니다.)
- 기본 SQL Server 복원 명령을 사용하여 데이터베이스를 복원합니다.
- 필요한 경우, 로그인 및 사용자를 생성하고 연결이 끊긴 사용자를 수정합니다.
- 필요에 따라 SQL Server 에이전트 작업을 생성하고 일정을 검토하십시오.

마이그레이션 후 단계

마이그레이션이 완료되면 다음을 수행할 수 있습니다.

- DB 인스턴스를 적절한 크기의 인스턴스 타입으로 변경합니다.

- 다AZ 및 백업 보존을 활성화합니다.
- 모든 작업이 보조 노드에서 생성되었는지 확인하십시오(다AZ 구성의 경우).
- Amazon CloudWatch Logs에 SQL Server 오류 및 에이전트 로그를 게시하고 CloudWatch를 통해 지표를 보고 경보를 생성합니다. 자세한 설명은 [Amazon RDS 설명서](#)를 참조하십시오.
- [고급 모니터링](#)을 통해 DB 인스턴스에 대한 지표를 실시간으로 가져옵니다.
- 알림을 위해 Amazon Simple Notification Service(Amazon SNS) 항목을 설정하십시오.

마이그레이션 테스트

새 Amazon RDS for SQL Server 데이터베이스에서 애플리케이션을 검증하려면 다음 테스트를 수행하는 것이 좋습니다.

- 기능 테스트를 수행하십시오.
- 소스 및 대상 데이터베이스의 SQL 쿼리 성능을 비교하고 필요에 따라 쿼리를 조정합니다. 일부 쿼리는 대상 데이터베이스에서 더 느리게 수행될 수 있으므로 소스 데이터베이스에서 SQL 쿼리의 기준을 캡처하는 것이 좋습니다.

Proof of Concept(POC) 단계에서 추가 검증을 받으려면 다음과 같은 추가 테스트를 수행하는 것이 좋습니다.

- 성능 테스트를 실행하여 비즈니스 기대치를 충족하는지 확인하세요.
- 데이터베이스 페일오버, 복구 및 복원을 테스트하여 RPO 및 RTO 요건을 충족하는지 확인하십시오.
- 모든 중요한 작업과 보고서를 나열하고 Amazon RDS에서 실행하여 서비스 수준에 관한 계약(SLA)에 따라 성능을 평가하세요.

Amazon RDS 데이터베이스 운영 및 최적화

AWS에 데이터베이스가 있을 때는 모니터링, 알림, 백업, 클라우드에서의 고가용성 등의 영역에서 모범 사례를 따르고 있는지 확인하십시오. 예:

- CloudWatch 모니터링을 설정하고 세부 모니터링을 활성화합니다.
- [Amazon RDS Performance Insights](#)와 [SentryOne](#) 또는 [SQL Server용 Foglight](#)와 같은 기타 타사 모니터링 솔루션을 사용하여 데이터베이스를 모니터링하십시오.
- SNS 항목을 사용하여 알림을 설정합니다.

- [AWS Backup](#) 또는 네이티브 SQL Server 백업을 사용하여 자동 백업을 설정하고 Amazon S3에 복사합니다.
- 고가용성을 위해서는 Amazon RDS 다AZ 기능을 사용하십시오.
- 읽기 전용 데이터베이스가 필요한 경우, 필요에 따라 동일한 지역 내에 또는 여러 AWS 지역에 [읽기 전용 복제본을 설정](#)하십시오.

Amazon EC2와 Amazon RDS 중 선택





Amazon EC2와 Amazon RDS는 특정 사용 사례에 유용할 수 있는 고유한 혜택을 제공합니다. 필요에 따라 SQL Server 데이터베이스에 하나 또는 두 서비스를 모두 유연하게 사용할 수 있습니다. 이 단원에서는 선택에 도움이 되는 세부 정보를 제공합니다.













결정 매트릭스

다음 표는 Amazon RDS, SQL Server용 Amazon RDS Custom 및 Amazon EC2에서 지원되는 SQL Server 기능을 side-by-side 비교한 것입니다. 이 정보를 사용하여 차이점을 이해하고 사용 사례에 가장 적합한 접근 방식을 선택할 수 있습니다.


Amazon RDS에 대한 최신 정보는 설명서에서 Amazon RDS의 [Microsoft SQL 서버를](#) 참조하십시오.
AWS

Development




개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
버퍼 풀 확장	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
BULK INSERT				Amazon RDS 설명서의 Amazon RDS for SQL Server DB 인스턴스와 Amazon S3 통합 을 참조하십시오.







개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
CDC(변경 데이터 캡처)	 se Edition: 모든 버전, Standard Edition: 2016 SP1 이상)			Amazon RDS 설명서의 변경 데이터 캡처 사용 을 참조하십시오.
변경 내용 추적				
Columnstore 인덱스	 se Edition: 2014 이상)	 (Enterprise Edition: 2019)	 se Edition: 2014 이상)	
데이터 품질 서비스	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.













개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
데이터베이스 메일		0: 	0: 	<p>블로그 게시물 Amazon RDS for SQL Server에 데이터베이스 메일 사용을 참조하십시오.</p> <p>높은 수준의 전송 가능성을 보장하려면 Amazon Simple Email Service (Amazon SES)를 사용하여 AWS 리소스에서 출발하는 아웃바운드 이메일을 보내는 것이 좋습니다.</p>
데이터베이스 엔진 튜닝 관리자		0: 	0: 	
DB 이벤트 알림		0: 	0:  니요(DB 이벤트를 수동으로 추적 및 관리)	<p>Amazon RDS 설명서의 Amazon RDS 이벤트 알림 사용을 참조하십시오.</p>










개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
DDL S3 이벤트 알림	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
지연된 트랜잭션 내구성(지연 커밋)	 Server 2014 이상)	 Server 2019)	 Server 2014 이상)	
분산된 쿼리	 Server 대상)	 (SQL Server 대상)	 (SQL Server 대상)	Amazon RDS for SQL Server 를 사용하여 연결된 서버 구현 블로그 게시물을 참조하십시오.
추가 이벤트				
xp_cmdshe 11을 포함한 저장 절차 확장	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
















개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
파일 테이블	 필요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
FILESTREAM	 필요			FILESTREAM은 Amazon RDS와 호환되지 않습니다. 그러나 인 메모리 데이터베이스를 구성할 수 있습니다.
전체 텍스트 검색	 (만틱 검색 제외)			
인 메모리 데이터베이스	 Server 2014 이상)	 Server 2019)	 Server 2014 이상)	
















개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
연결된 서버	 Server 및 Oracle 대상)	0: 	0: 	Amazon RDS 설명서에서 Amazon RDS for SQL Server 를 사용한 연결 서버 구현 블로그 게시물 및 Amazon RDS for SQL Server의 Oracle OLEDB를 사용한 연결 서버 지원 을 참조하십시오.

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
기계 학습 서비스(R 스크립트 사용)				<p>기계 학습 서비스는 Windows 또는 Linux 시스템에 별도로 설치해야 합니다. SQL Server 2019 이상의 Always On 장애 조치 클러스터 인스턴스(FCI)에서만 지원됩니다.</p> <p>R은 Amazon RDS에서 지원되지 않지만, 에서 사용할 수 있습니다 AWS (블로그 게시물 R on AWS 참조).</p>
유지 관리 계획	 니요			<p>Amazon RDS는 데이터베이스의 백업 및 복구를 용이하게 하는 별도의 기능 세트를 제공합니다. 백업의 경우 자동 백업을 구성할 수 있습니다.</p>













개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
마스터 데이터 서비스	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
MSDTC(Microsoft Distributed Transaction Coordinator)				블로그 게시물 도메인에 가입된 Amazon RDS for SQL Server 인스턴스에 대한 분산 트랜잭션 지원 활성화 를 참조하십시오.
OPENROWSET				
부분적으로 포함된 데이터베이스	 Server 2014 이상)	 Server 2019)	 Server 2014 이상)	

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
성능 데이터 수집기	 니요			Amazon RDS에서는 Amazon CloudWatch, AWS CloudTrail, 및 Performance Insights를 사용하여 SQL 서버 성능을 모니터링할 수 있습니다 (Amazon RDS 설명서의 Amazon RDS 모니터링 개요 참조).
정책 기반 관리	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
PolyBase	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.







개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
사전 구성된 파라미터		 0: 필요	 0: 필요	0:
리소스 거버너	 0: 필요	 0:	 0:	0: 워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
안전한 CLR	 0: Server 2014 및 2016)	 0:	 0:	0:
시퀀스	 0: Server 2014 이상)	 0: Server 2019)	 0: Server 2014 이상)	0:
서버 수준 트리거	 0: 필요	 0:	 0:	0: 워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
서비스 브로커 (드포인트 제외)				
공간 및 위치 기능				
SQL Server 에이전트				
SSAS(SQL Server Analysis Services) (Server 2016 이상)				Amazon RDS 문서에서 Amazon RDS for SQL Server의 SSAS 지원을 참조하십시오.
SSIS(SQL Server Integration Services) (Server 2016 이상)				Amazon RDS 설명서에서 Amazon RDS for SQL Server의 SSIS 지원을 참조하십시오.
SSMS(SQL Server Management Studio)				

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
SQL Server Migration Assistant (SSMA)				
SQL Server 프로파일러	 버측 추적 및 클라이언트측 추적)			
SSRS(SQL Server Reporting Services)	 Server 2016 이상)			Amazon RDS 설명서에서 Amazon RDS for SQL Server의 SSRS 지원 을 참조하십시오.
sqlcmd				
Stretch 데이터베이스	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
THROW 명령문	 Server 2014 이상)	 Server 2019)	 Server 2014 이상)	
Transact-SQL 엔드포인트	 니요			Amazon RDS에서는 CREATE ENDPOINT를 사용하는 모든 작업을 사용할 수 없습니다. 이러한 작업을 수행하려면 EC2 인스턴스에 SQL Server를 설치하는 것이 좋습니다.
UTF-16 지원	 Server 2014 이상)		 Server 2014 이상)	
WCF 데이터 서비스	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.

HA/DR

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
Always On 가용성 그룹		0:  기 및 비동기 모 두)	0: 	자체 관리형 Always On 가용성 그룹이 필요한 경우 EC2 인스턴스에서의 SQL Server HA AWS Launch Wizard 배포를 간소화하는데 사용하는 것이 좋습니다. 설명서의 SQL Server에 AWS Launch Wizard 대한 내용을 참조하십시오. AWS
Always On 장애 조치 클러스터 인스턴스(FCI)	 니요	0: 	0: 	를 AWS Launch Wizard 사용하여 Amazon EC2에서의 SQL Server FCI 배포를 간소화할 수 있습니다. 설명서의 SQL Server에 AWS Launch Wizard 대한 내용을 참조하십시오. AWS

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
Amazon S3로 백업				Amazon RDS는 리포지토리로서 전체 백업 파일 (.bak 파일) 및 Amazon S3를 사용하여 SQL Server 데이터베이스에 기본 백업 및 복원을 할 수 있도록 지원합니다. Amazon RDS 설명서의 SQL Server 데이터베이스 가져오기 및 내보내기 를 참조하십시오.
BACKUP 명령	 니요			SQL Server를 실행하는 Amazon RDS DB 인스턴스의 네이티브 백업을 수행하려면 어떻게 해야 하나요? 를 참조하십시오. AWS 지식 센터에서.
데이터베이스 미러링	 중 AZ)			








개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
데이터베이스 복제	 니요(제한된 푸시 구독)			Amazon RDS에서 단일 테이블을 복제하려는 경우 AWS DMS 를 사용하거나 읽기 전용 복제본을 설정할 수도 있습니다.
분산 가용성 그룹	 니요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
로그 전달	 니요			재해 복구를 위해 읽기 전용 복제본 또는 AWS DMS 를 사용할 수 있습니다.
관리형 자동 백업			 니요(유지 관리 계획을 구성 및 관리하거나 타사 솔루션을 사용해야 함)	Amazon RDS 설명서의 백업 작업 을 참조하십시오.

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
자동 장애 조치가 포함된 다중 AZ		 On 가용성 그룹을 수동으로 구성한 경우)	 se Edition 전용, Always On 가용성 그룹을 수동으로 구성한 경우)	Amazon RDS 설명서의 Amazon RDS for SQL Server용 다중 AZ 배포 를 참조하십시오.
읽기 전용 복제본	 Server 2016 이상)	 On 가용성 그룹을 수동으로 구성한 경우)	 On 가용성 그룹을 수동으로 구성한 경우)	
RESTORE 명령				AWS 지식 센터 를 참조하십시오.







Scalability













개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
내장 인스턴스 및 데이터베이스 모니터링 및 지표		 니요	 니요 (자체 지표를 타사 솔루션으로 CloudWatch 내보내거나 타사 솔루션 사용)	Amazon CloudWatch 및 AWS Systems Manager 에서 사용자 지정 지표를 사용하여 SQL Server 데이터베이스를 모










개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
				니터링하는 블로그 게시물을 참조하십시오.
구성 가능한 스토리지 크기				
인스턴스당 최대 데이터베이스 수	인스턴스 크기 및 다중 AZ 구성에 따라 다름	SQL Server 최대(5000)	 한 없음	Microsoft SQL Server 설명서의 SQL Server의 최대 용량 사양 을 참조하십시오.
DB 인스턴스의 최대 스토리지 크기	16TiB	16TiB	 한 없음	Amazon RDS는 비휘발성 메모리 익스프레스(NVMe) 인스턴스 스토리지를 사용하여 로컬 디스크의 tempdb 데이터베이스도 지원합니다. Amazon RDS 설명서의 Amazon RDS for SQL Server의 tempdb 데이터베이스에 대한 인스턴스 스토어 지원 을 참조하십시오.

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
DB 인스턴스의 최소 스토리지 크기	20GiB (Enterprise, Standard, Web 및 Express Edition)	20GiB (Enterprise, Standard, Web 및 Express Edition)	 한 없음	자
새로운 쿼리 옵티마이저	 Server 2014 이상)	 Server 2014 이상)	 Server 2014 이상)	
읽기 전용 복제본	 Server 2016 이상)	 On 가용성 그룹을 수동으로 구성한 경우)	 On 가용성 그룹을 수동으로 구성한 경우)	













Security







개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
자동 소프트웨어 패치		 니요	 니요	
암호화된 스토리지는 다음을 사용합니다. AWS KMS	 를 제외한 모든			AWS KMS 암호화를 사용한 Amazon RDS의 데이터 보안 블

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
	SQL Server 에디션)			로그 게시물을 참조하십시오.
유연한 서버 역할	 Server 2014 이상)	 Server 2019)	 Server 2014 이상)	
SQL 인증				
SQL Server Audit				
SSL(전송 중 암호화)				Amazon RDS 설명서의 Microsoft SQL Server DB 인스턴스에 SSL 사용을 참조하십시오.

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
sysadmin 역할	 필요			지원되지 않는 서버 수준 역할에 대해서는 Amazon RDS 설명서의 Microsoft SQL Server 보안 을 참조하십시오. 새로운 DB 인스턴스를 생성할 때, 기본 마스터 사용자는 해당 DB 인스턴스에 대한 특정 권한을 갖습니다 (Amazon RDS 설명서의 계정 권한 참조).
TDE(저장된 암호화)	 se Edition: 2014-2019, Standard Edition: 2019)	 (SQL Server 2019 엔터프라이즈, 스탠다드, 웹 및 개발자 에디션)	 se Edition: 2014-2019, Standard Edition: 2019)	Amazon RDS 및 Amazon RDS 사용자 지정 설명서에서 TDE 지원에 대한 정보를 참조하십시오.
Windows 인증				

Other features

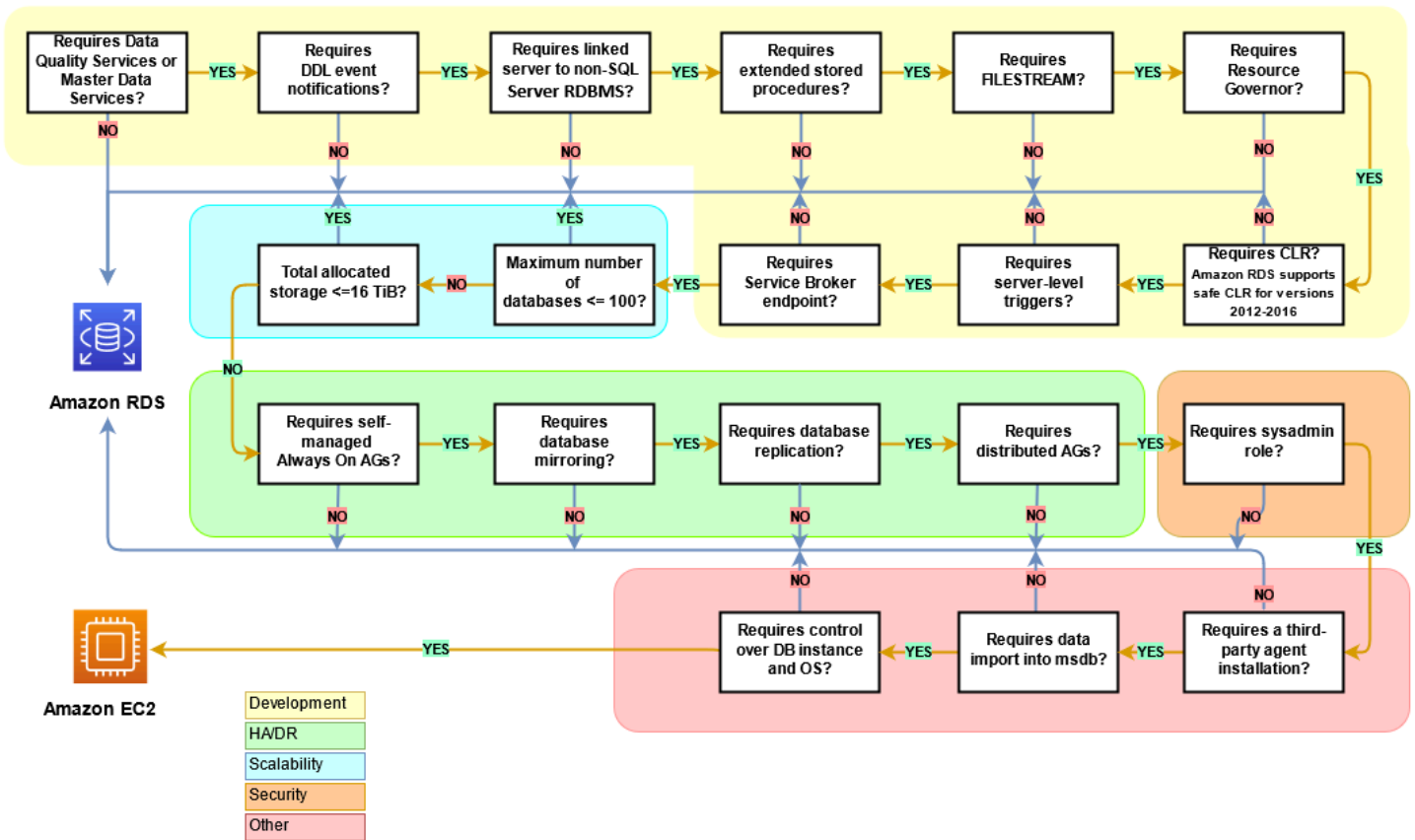
개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
타사 에이전트 설치 기능	 필요			
기존 데이터베이스 이름 변경 기능	 (일 AZ만 해당)	 (용성 그룹의 데이터베이스나 미러링이 활성화된 데이터베이스에는 사용할 수 없음)	 (용성 그룹의 데이터베이스나 미러링이 활성화된 데이터베이스에는 사용할 수 없음)	Amazon RDS에서의 다중 AZ 배포의 경우 Amazon RDS 설명서의 다중 AZ 배포에서 Microsoft SQL Server 데이터베이스 이름 변경 을 참조하십시오.
DB 인스턴스 및 운영 체제에 대한 제어	 필요			워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
사용자 지정 시간대 설정				

개발 기능	Amazon RDS	Amazon RDS Custom	Amazon EC2	참고
분산 재생	 0: 필요	 0:	 0:	SQL Server 분산 재생 클라이언트 서비스에는 sysadmin 권한이 필요하므로 Amazon RDS에서 지원되지 않습니다.
msdb 데이터베이스로 데이터 가져오기	 0: 필요	 0:	 0:	워크로드에 이 기능이 중요한 경우 Amazon RDS Custom 또는 Amazon EC2를 선택하는 것을 고려하십시오.
설치 방법	N/A	N/A	Amazon Machine Image(AMI) 또는 수동 설치	
SQL Server 에디션	Enterprise, Standard, Web, Express	엔터프라이즈, 스탠다드, 개발자	Enterprise, Standard, Web, Developer, Express	
SQL Server 버전	2014년, 2016년, 2017년, 2019년, 2022년	2019, 2022년	2014, 2016, 2017, 2019, 2022년	

이 기능에 대한 자세한 내용은 다음 주제를 참조하십시오.

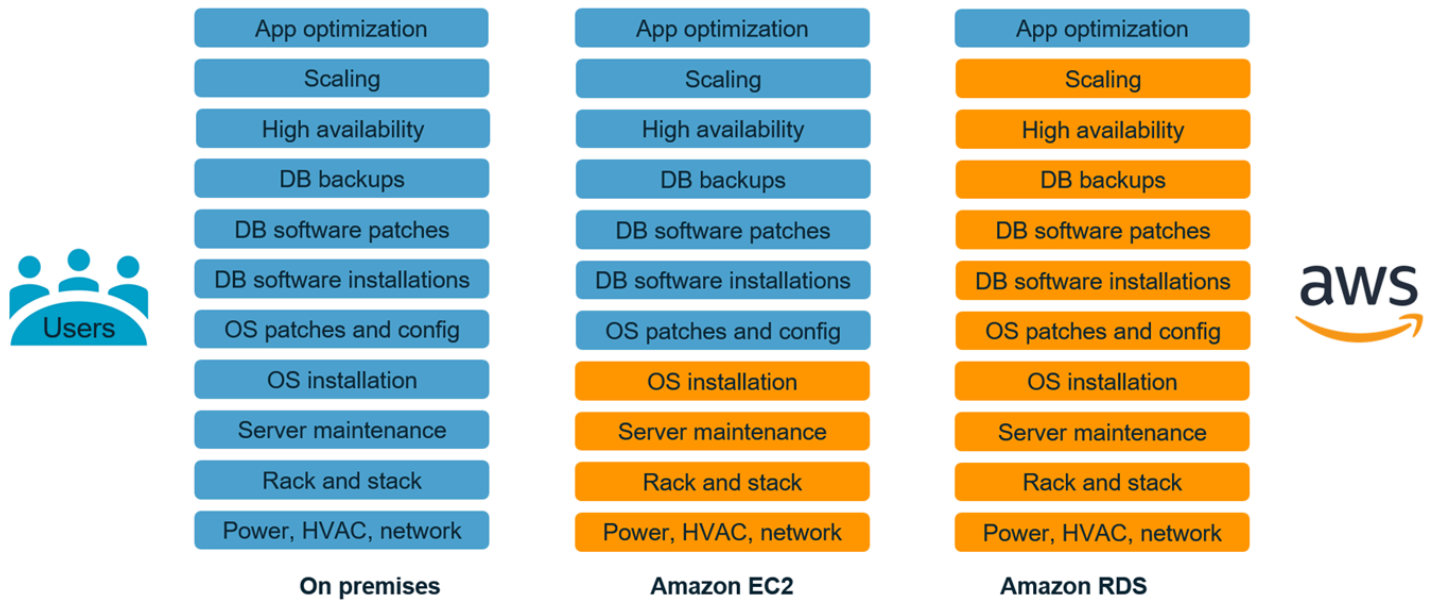
- [AWS에서의 Microsoft 제품](#)
- [Active Directory 참조 아키텍처: AWS에서의 Active Directory 도메인 서비스 구현](#)
- [원격 데스크톱 게이트웨이 커짐 AWS \(AWS 킷 스타트\)](#)
- [마이크로소프트 플랫폼 보안 AWS](#)
- [상시 복제가 켜져 있는 SQL Server AWS \(AWS 킷 스타트\)](#)
- [AWS Directory Service](#)
- [AWSEC2-SQLServerDBRestore](#)(Amazon S3에 저장된 SQL 서버 데이터베이스 백업을 EC2 Linux 인스턴스에서 실행되는 SQL Server 2017로 복원하는AWS Systems Manager Automation 런북)

다음 다이어그램은 위 표의 정보를 시각화하여 의사 결정 프로세스를 지원합니다.



공동 책임

다음 다이어그램은 SQL Server 기능 AWS 및 운영 관리에 대한 책임과 사용자 간의 책임 분담을 보여줍니다.



AWS 서비스를 사용하면 서버 프로비저닝, 패치, 설정, 구성, 백업 또는 복구와 같은 관리 작업에 대해 걱정할 필요가 없습니다. AWS 클러스터를 지속적으로 모니터링하여 자가 복구 스토리지와 자동화된 크기 조정을 통해 워크로드가 계속 가동되고 실행되도록 합니다. 스키마 설계, 쿼리 구성, 최적화와 같은 고부가가치 애플리케이션 개발 작업에 집중하고 운영 작업은 대신 AWS 처리합니다.

애플리케이션 증가, 간헐적인 급증, 성능 요구 사항을 수용하기 위해 인프라를 과도하게 프로비저닝하거나 과소 프로비저닝할 필요가 없으며, 소프트웨어 라이선싱 및 지원, 하드웨어 교체, 하드웨어 유지 관리를 위한 리소스 등 고정 자본 비용이 발생할 필요가 없습니다. AWS 이를 관리하므로 인프라 관리가 아닌 새로운 애플리케이션을 혁신하고 구축하는 데 시간을 할애할 수 있습니다.

자세한 내용은 AWS 웹사이트의 [공동 책임 모델](#)을 참조하십시오.

SQL Server 데이터베이스 마이그레이션 패턴

다음 링크를 사용하여 SQL Server 데이터베이스를 AWS(으)로 마이그레이션하기 위한 AWS 규범적 지침 패턴을 확인하세요:

- [리호스팅 패턴 \(SQL 서버에서 Amazon EC2로\)](#)
- [리플랫폼 패턴 \(SQL 서버에서 Amazon RDS for SQL Server로\)](#)
- [패턴 재설계 \(SQL Server에서 오픈 소스 및 AWS 클라우드 네이티브 데이터베이스로\)](#)

특정 도구 사용에 대한 패턴을 찾으려면 검색 상자에 도구 이름을 입력하거나 필터에서 도구 이름을 선택하세요. 예를 들어 [이 쿼리](#)를 사용하면 AWS DMS을(를) 사용하는 모든 SQL Server 마이그레이션 패턴을 확인할 수 있습니다.

파트너

데이터베이스 마이그레이션은 전문 지식과 도구가 필요한 까다로운 프로젝트일 수 있습니다. 파트너십을 통해 마이그레이션을 가속화하고 결과 도출 시간을 단축할 수 있습니다. [AWS Database Migration Service 파트너](#)는 고객이 쉽고 안전하게 클라우드로 마이그레이션할 수 있도록 지원하는 데 필요한 전문 지식을 갖추고 있습니다. 이 파트너는 SQL Server에서 SQL Server로의 동종 마이그레이션과 SQL Server에서 Amazon Aurora로 또는 Amazon RDS for MySQL과 같이 서로 다른 데이터베이스 플랫폼 간의 이기종 마이그레이션에 대한 전문 지식을 두루 갖추고 있습니다.

요구 사항 및 선호에 따라 파트너를 통해 전체 마이그레이션을 처리하거나 마이그레이션의 일부 측면만 도움을 받을 수 있습니다. 또한 AWS 파트너 네트워크(APN)가 제공하는 도구와 솔루션을 사용하여 마이그레이션에 도움을 받을 수 있습니다. 마이그레이션 도구 및 솔루션의 전체 카탈로그는 [AWS 파트너 도구 및 솔루션](#)을 참조하세요.

추가 리소스

블로그 게시물

- [Amazon RDS for SQL Server의 지역 간 재해 복구](#)
- [Database Migration—What Do You Need to Know Before You Start?](#)
- [Amazon EC2 윈도우와 Amazon리눅스 2 인스턴스 간 Always On 가용성 그룹 배포](#)
- [분산된 가용성 그룹을 사용하여 하이브리드 Microsoft SQL Server 솔루션을 설계하는 방법](#)
- [트랜잭션 복제를 사용하여 Amazon RDS for SQL Server로 마이그레이션하는 방법](#)
- [AWS Database Migration Service를 사용하여 Amazon RDS for SQL Server에서 진행 중인 복제 소개](#)
- [AWS가 Microsoft Windows Server 및 SQL Server 워크로드를 실행하는 데 가장 적합한 클라우드인 이유를 알아보십시오.](#)
- [온프레미스 SQL Server Windows 워크로드를 Amazon EC2 Linux로 마이그레이션하십시오](#)
- [SQL Server 데이터베이스를 MySQL 호환 데이터베이스 엔진으로 마이그레이션](#)
- [온프레미스 SQL Server Windows 워크로드를 Amazon EC2 Linux로 마이그레이션하기](#)
- [FSx for Windows File Server를 사용하여 Microsoft SQL Server고가용성 배포를 단순화하세요](#)
- [Store SQL Server backups in Amazon S3 using Storage Gateway](#)

AWS 설명서

- [Amazon Aurora](#)
- [Amazon EC2](#)
- [Amazon RDS](#)
- [Amazon RDS Custom](#)
- [AWS DMS](#)
- [AWS SCT](#)
- [SQL Server Licensing](#)

감사의 말

작성자는 이 가이드에 기여한 다음 전문가들에게 감사를 포함합니다.

- Marcelo Fernandes, AWS 마이그레이션 시니어 컨설턴트 — [Amazon EC2와 Amazon RDS 중에서 선택하기](#) 섹션
- Tarun Chawla, 데이터베이스 마이그레이션 컨설턴트 — [Amazon EC2와 Amazon RDS 중에서 선택하기](#) 섹션
- Alex Zuo, 시니어 기술 제품 관리자, Amazon EC2의 SQL 서버 — [AWS Migration Hub Orchestrator](#) 섹션

부록: SQL Server 데이터베이스 마이그레이션 설문

이 섹션의 설문을 출발점으로 삼아 마이그레이션 프로젝트의 평가 및 계획 단계들을 위한 정보를 수집 하십시오. 이 설문을 Microsoft Excel 형식으로 다운로드하여 정보를 기록하는 데 사용할 수 있습니다.



[설문 다운로드](#)

일반 정보

1. 귀하 SQL Server 인스턴스의 명칭은 무엇입니까?
2. SQL Server 버전은 무엇입니까?
3. SQL Server 데이터베이스의 에디션은 무엇입니까: 표준, 개발자 또는 기업?
4. 데이터베이스 타입(OLTP, DW, 보고, 배치 처리)은 무엇입니까?
5. SQL Server 인스턴스에는 몇 개의 데이터베이스가 있습니까?
6. 데이터베이스 크기는 어느 정도입니까?
7. 데이터베이스 정렬이란 무엇입니까?
8. 데이터베이스의 시간대는 무엇입니까?
9. 초당 평균 및 최대 I/O 트랜잭션(TPS)은 얼마입니까?
10. 데이터베이스의 읽기/쓰기 작업 IOPS(평균 및 최대)는 얼마입니까?
11. 시간당 생성되는 트랜잭션 로그는 몇 개입니까(평균 및 최대 크기 기준)?
12. 데이터베이스에 다른 데이터베이스를 가리키는 연결된 서버가 있습니까?
13. 데이터베이스의 SLA 요건은 무엇입니까?
14. 데이터베이스에 대해 RTO 및 RPO의 요건은 무엇입니까?
15. 마이그레이션을 위해 허용할 수 있는 데이터베이스 가동 중지 시간은 어느 정도입니까?
16. 규정 준수, 규제 또는 감사 요건이 있습니까?
17. SQL Server 데이터베이스를 모니터링하기 위해 어떤 도구를 사용하십니까?

인프라

1. 데이터베이스의 호스트 명칭은 무엇입니까?

2. 이 데이터베이스에 사용되는 작동 시스템은 무엇입니까?
3. 서버의 CPU 코어는 몇 개입니까?
4. 서버의 메모리 크기는 얼마입니까?
5. 데이터베이스가 가상화된 시스템에 있습니까, 아니면 물리적 서버에 있습니까?
6. 로컬 스토리지를 사용하고 있습니까?
7. 네트워크 연결 스토리지(NAS) 또는 스토리지 영역 네트워크(SAN) 스토리지 타입을 사용하십니까?
8. 클러스터나 단일 인스턴스가 있습니까?

데이터베이스 백업

1. 데이터베이스를 어떻게 백업합니까? 얼마나 자주?
2. 트랜잭션 로그 및 백업의 보존 기간은 어떻게 됩니까?
3. 백업은 어디에 저장하나요?

데이터베이스 기능

1. SQL Server 인스턴스에 자동 튜닝을 사용합니까?
2. 병렬 인덱스 연산을 사용합니까?
3. 분할된 표 병렬 처리 기능을 사용합니까?
4. 표 및 인덱스 파티셔닝을 사용합니까?

데이터베이스 보안

1. 동적 데이터 마스킹을 사용합니까?
2. Transparent Database Encryption(TDE)과 같은 보안 기능을 사용하십니까?
3. 서버 감사를 사용하십니까 아니면 데이터베이스 감사를 사용하십니까?
4. 고급 압축을 사용합니까?

데이터베이스 고가용성 및 재해 복구

1. 고가용성 요건은 무엇입니까?

-
2. 트랜잭션 복제를 사용하십니까?
 3. P2P 트랜잭션 복제를 사용하십니까?
 4. SQL Server 환경에서는 어떤 타입의 고가용성 솔루션(예: 장애 조치 클러스터링, Always On 가용성 그룹, 데이터베이스 미러링)을 사용하고 있습니까?
 5. 기본 및 대기 데이터베이스 영역은 어디에 있습니까?
 6. 재해 복구 솔루션으로 어떤 것을 사용하십니까(예: 로그 전달, Always On 가용성 그룹, SAN 기반 가상화 환경)?
 7. 데이터베이스 연결성을 위해 도메인 이름 시스템(DNS) 별칭을 사용하십니까?

문서 기록

아래 표에 이 가이드의 주요 변경 사항이 설명되어 있습니다. 향후 업데이트에 대한 알림을 받으려면 [RSS 피드](#)를 구독하십시오.

변경 사항	설명	날짜
업데이트된 섹션	영구 예약 및 상시 운영 장애 조치 클러스터 인스턴스 섹션 에 FCI를 AWS 배포하기 위한 기타 옵션을 포함한 Amazon EBS Multi-Attach에 대한 정보 가 추가되었습니다.	2024년 4월 1일
업데이트된 정보	SQL Server 2022의 의사 결정 매트릭스 를 업데이트했습니다.	2024년 3월 18일
업데이트된 정보	Amazon RDS Custom이 이제 TDE를 지원한다는 점을 반영하도록 의사 결정 매트릭스 를 업데이트했습니다.	2023년 11월 16일
추가된 섹션	AWS Migration Hub Orchestrator 에 대한 정보가 추가되었습니다.	2023년 6월 29일
삭제된 섹션	종단되는 CloudEndure 마이그레이션 관련 정보가 제거되었습니다. AWS Application Migration Service 로 마이그레이션하는 데 권장되는 기본 마이그레이션 lift-and-shift 서비스입니다. AWS 클라우드	2022년 9월 23일
업데이트된 섹션	의사 결정 매트릭스 에 더 많은 정보를 추가했습니다.	2022년 8월 3일

추가 및 업데이트된 섹션	SQL Server 데이터베이스를 Amazon RDS Custom으로 마이그레이션 하고 Babelfish 를 사용하여 SQL Server 데이터베이스를 Aurora PostgreSQL로 마이그레이션하는 방법에 대한 정보가 추가되었습니다. 의사결정 매트릭스 를 최신 정보로 업데이트했습니다.	2022년 7월 29일
수정된 텍스트	의사결정 매트릭스 의 CLR 정보를 수정했습니다.	2022년 6월 21일
업데이트된 섹션	제품 가용성에 대한 최신 정보로 CloudEndure 마이그레이션 섹션을 업데이트했습니다.	2022년 5월 10일
삭제된 섹션	더 이상 지원되지 않는 VMware용 Amazon RDS 관련 정보가 제거되었습니다.	2022년 4월 19일
추가된 섹션	SQL Server용 AWS Launch Wizard 에 대한 정보가 추가되었습니다.	2021년 7월 15일
추가된 의사결정 매트릭스	Amazon EC2와 Amazon RDS 중에서 선택하기 섹션에서는 SQL Server 지원을 side-by-side 비교했습니다.	2021년 6월 28일
업데이트된 로그 전달 정보	로그 전달 섹션에서 Amazon RDS for SQL Server에서의 로그 전송에는 사용자 지정 스크립트가 필요하다는 점을 명확히 했습니다.	2021년 3월 25일
최초 게시	—	2020년 10월 9일

AWS 권장 가이드 용어집

다음은 AWS 권장 가이드에서 제공하는 전략, 가이드, 패턴 등에서 일반적으로 사용되는 용어입니다. 용어집 항목을 제안하려면 용어집 끝에 있는 피드백 제공 링크를 사용하세요.

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. 예를 들어, 온프레미스 Oracle 데이터베이스를 Amazon Aurora PostgreSQL 호환 버전으로 마이그레이션합니다.
- 리플랫폼(리프트 앤드 리세이프) - 애플리케이션을 클라우드로 이동하고 일정 수준의 최적화를 도입하여 클라우드 기능을 활용합니다. 예를 들어, 온프레미스 Oracle 데이터베이스를 AWS 클라우드의 Amazon RDS for Oracle로 마이그레이션합니다.
- 재구매(드롭 앤드 슝) - 일반적으로 기존 라이선스에서 SaaS 모델로 전환하여 다른 제품으로 전환합니다. 예를 들어, 고객 관계 관리(CRM) 시스템을 Salesforce.com으로 마이그레이션합니다.
- 리호스팅(리프트 앤드 시프트) - 애플리케이션을 변경하지 않고 클라우드로 이동하여 클라우드 기능을 활용합니다. 예를 들어, 온프레미스 Oracle 데이터베이스를 AWS 클라우드의 EC2 인스턴스에 있는 Oracle로 마이그레이션합니다.
- 재배포(하이퍼바이저 수준의 리프트 앤 시프트) - 새 하드웨어를 구매하거나, 애플리케이션을 다시 작성하거나, 기존 운영을 수정하지 않고도 인프라를 클라우드로 이동합니다. 이 마이그레이션 시나리오는 온프레미스 환경과 AWS 간의 가상 머신 호환성 및 워크로드 이동성을 지원하는 AWS의 VMware Cloud에 한정됩니다. 인프라를 AWS의 VMware Cloud로 마이그레이션할 때 온프레미스 데이터 센터에서 VMware Cloud Foundation 기술을 사용할 수 있습니다. 예를 들어, Oracle 데이터베이스를 호스팅하는 하이퍼바이저를 AWS의 VMware Cloud로 재배포합니다.
- 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](#) 제어를 참조하십시오.

추상화된 서비스

[매니지드 서비스를](#) 참조하십시오.

산

[원자성, 일관성, 격리성, 내구성을](#) 참조하십시오.

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 마이그레이션보다 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 연산을 수행하고 그룹에 대한 단일 반환값을 계산하는 SQL 함수입니다. 집계 함수의 예로는 및 등이 SUM 있습니다. MAX

AI

[인공 지능을](#) 참조하십시오.

AIOps

[인공 지능 운영을](#) 참조하십시오.

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

시스템을 멀웨어로부터 보호하기 위해 승인된 애플리케이션만 사용할 수 있는 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 검색 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 참조하세요.

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 참조하세요.

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 [AWS용 ABAC란 무엇입니까?](#)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 영역에 장애가 발생할 경우 분리되도록 설계된 AWS 리전 내의 개별적인 지점으로, 같은 리전 내의 다른 가용 영역에 비해 저렴하고 지연 시간이 짧은 네트워크 연결을 제공합니다.

AWS Cloud Adoption Framework(AWS CAF)

조직이 클라우드로 성공적으로 전환하기 위한 효율적이고 효과적인 계획을 개발하는 데 도움이 되는 AWS의 지침 및 모범 사례 프레임워크입니다. AWS CAF는 비즈니스, 사람, 거버넌스, 플랫폼, 보안 및 운영 관점이라는 6가지 중점 영역으로 지침을 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 조직이 클라우드를 성공적으로 채택할 수 있도록 인력 개발, 교육 및 커뮤니케이션에 대한 지침을 제공합니다. 자세한 내용은 [AWS CAF 웹 사이트](#)와 [AWS CAF 백서](#)를 참조하세요.

AWS Workload Qualification Framework(AWS WQF)

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

B

BCP

[비즈니스 연속성 계획을](#) 참조하십시오.

동작 그래프

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

빅 엔디안 시스템

가장 중요한 바이트를 먼저 저장하는 시스템입니다. [엔디안도](#) 참조하십시오.

바이너리 분류

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

블룸 필터

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

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 [브랜치 정보](#) (GitHub 문서) 를 참조하십시오.

브레이크 글래스 액세스

예외적인 상황에서 승인된 프로세스를 통해 사용자가 일반적으로 액세스 권한이 없는 데이터에 빠르게 액세스할 수 있는 AWS 계정 있는 수단입니다. 자세한 내용은 Well-Architected AWS 지침의 [브레이크 글래스 절차 구현](#) 표시기를 참조하십시오.

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 [컨테이너화된 마이크로서비스 실행](#) AWS의 [비즈니스 역량 중심의 구성화](#) 섹션을 참조하세요.

비즈니스 연속성 계획(BCP)

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

C

CAF

[클라우드 채택 프레임워크를 참조하십시오AWS.](#)

CCoE

[클라우드 센터 오브 엑셀런스를 참조하십시오.](#)

CDC

[변경 데이터 캡처를 참조하십시오.](#)

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 장애를 일으키는 이벤트를 발생시키는 행위 [AWS Fault Injection Service\(AWS FIS\)](#) 를 사용하여 AWS 워크로드에 스트레스를 주는 실험을 수행하고 응답을 평가할 수 있습니다.

CI/CD

[지속적 통합 및 지속적 전달을 참조하십시오.](#)

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

데이터를 대상 AWS 서비스에서 수신하기 전에 로컬에서 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](#) 기술과 연결됩니다.

클라우드 운영 모델

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

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 AWS Cloud Enterprise Strategy Blog의 [The Journey Toward Cloud-First & the Stages of Adoption](#) 블로그 게시물에서 정의했습니다. AWS 마이그레이션 전략과 어떤 관련이 있는지에 대한 자세한 내용은 [마이그레이션 준비 가이드](#)를 참조하세요.

CMDB

[구성 관리 데이터베이스](#)를 참조하십시오.

코드 리포지토리

소스 코드와 설명서, 샘플, 스크립트 등의 기타 자산이 버전 관리 프로세스를 통해 저장되고 업데이트되는 위치입니다. 일반 클라우드 리포지토리에는 또는 이 포함됩니다 GitHub . AWS CodeCommit 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

기계가 이미지 속 사람, 장소, 사물을 인간 수준 이상의 정확도로 식별하는 데 사용하는 AI 분야입니다. 딥 러닝 모델로 구축되는 경우가 많으며 단일 이미지 또는 일련의 이미지에서 유용한 정보를 자동으로 추출, 분석, 분류 및 이해합니다.

구성 관리 데이터베이스(CMDB)

하드웨어 및 소프트웨어 구성 요소와 해당 구성을 포함하여 데이터베이스와 해당 IT 환경에 대한 정보를 저장하고 관리하는 리포지토리입니다. 일반적으로 마이그레이션의 포트폴리오 검색 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 AWS 계정 및 리전 또는 조직 전체에 단일 엔터티로 규정 준수 팩을 배포할 수 있습니다. 자세한 내용은 AWS Config 설명서의 [Conformance packs](#)를 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 참조하세요. CD는 지속적 배포를 의미하기도 합니다. 자세한 내용은 [지속적 전달\(Continuous Delivery\)](#)과 [지속적인 개발](#)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 Framework 보안 원칙의 구성 요소입니다. 자세한 내용은 [데이터 분류](#)를 참조하세요.

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 최소화

꼭 필요한 데이터만 수집하고 처리하는 원칙입니다. AWS 클라우드에서 데이터 최소화를 실천하면 개인 정보 보호 위험, 비용 및 분석에 따른 탄소 발자국을 줄일 수 있습니다.

데이터 경계

신뢰할 수 있는 ID만 예상 네트워크에서 신뢰할 수 있는 리소스에 액세스하도록 하는 데 도움이 되는 AWS 환경 내 일련의 예방 가드레일입니다. 자세한 내용은 [데이터 경계 구축](#)을 참조하십시오.

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 시스템. 데이터 웨어하우스에는 일반적으로 대량의 과거 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

[데이터베이스 정의 언어를 참조하십시오.](#)

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.

defense-in-depth

네트워크와 그 안의 데이터 기밀성, 무결성 및 가용성을 보호하기 위해 컴퓨터 네트워크 전체에 일련의 보안 메커니즘과 제어를 신중하게 계층화하는 정보 보안 접근 방식입니다. AWS에서 이 전략을 채택하면 AWS Organizations 구조의 다양한 계층에 여러 제어 기능을 추가하여 리소스를 보호할 수 있습니다. 예를 들어, defense-in-depth 접근 방식에는 다단계 인증, 네트워크 분할 및 암호화를 결합할 수 있습니다.

위임된 관리자

AWS Organizations에서 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 참조하세요.

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

[환경을 참조하십시오.](#)

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 [Implementing security controls on AWS의 Detective controls](#)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.

치수 표

[스타 스키마에서](#) 팩트 테이블의 양적 데이터에 대한 데이터 속성을 포함하는 작은 테이블입니다. 차원 테이블 속성은 일반적으로 텍스트처럼 동작하는 텍스트 필드 또는 불연속형 숫자입니다. 이러한 속성은 일반적으로 쿼리 제한, 필터링 및 결과 집합 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)

[재해로 인한 다운타임과 데이터 손실을 최소화하기 위해 사용하는 전략과 프로세스입니다.](#) 자세한 내용은 AWS Well-Architected Framework의 [AWS 기반 워크로드의 재해 복구: 클라우드에서의 재해 복구](#)를 참조하세요.

DML

[데이터베이스 조작 언어](#)를 참조하십시오.

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 도

메인 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 참조하세요.

DR

[재해 복구를](#) 참조하십시오.

드리프트 감지

기존 구성으로부터의 편차 추적 예를 들어 [시스템 리소스의 편차를 감지하는 AWS CloudFormation](#) 데 사용하거나 거버넌스 요구 사항 준수에 영향을 미칠 수 있는 [착륙 지대의 변경을 탐지하는 AWS Control Tower](#) 데 사용할 수 있습니다.

DVSM

[개발 가치 흐름 매핑](#)을 참조하십시오.

E

EDA

[탐색적 데이터 분석](#)을 참조하십시오.

엣지 컴퓨팅

IoT 네트워크의 엣지에서 스마트 디바이스의 컴퓨팅 성능을 개선하는 기술 [클라우드 컴퓨팅과](#) 비교할 때 엣지 컴퓨팅은 통신 대기 시간을 줄이고 응답 시간을 개선할 수 있습니다.

암호화

사람이 읽을 수 있는 일반 텍스트 데이터를 암호문으로 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](#) 엔드포인트를 참조하십시오.

엔드포인트 서비스

Virtual Private Cloud(VPC)에서 호스팅하여 다른 사용자와 공유할 수 있는 서비스입니다. AWS PrivateLink를 사용하여 엔드포인트 서비스를 생성하고 다른 AWS 계정 또는 AWS Identity and Access Management(IAM) 보안 주체에게 권한을 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 참조하세요.

봉투 암호화

암호화 키를 다른 암호화 키로 암호화하는 프로세스입니다. 자세한 내용은 AWS Key Management Service(AWS KMS) 설명서의 [Envelope encryption](#)을 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 대응 등이 포함됩니다. AWS 마이그레이션 전략의 에픽에 대한 자세한 내용은 [프로그램 구현 가이드](#)를 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 스키마의](#) 중앙 테이블. 비즈니스 운영에 대한 정량적 데이터를 저장합니다. 일반적으로 팩트 테이블에는 측정값이 포함된 열과 차원 테이블의 외부 키가 포함된 열 등 두 가지 유형의 열이 포함됩니다.

빨리 실패하세요

빈번하고 점진적인 테스트를 통해 개발 라이프사이클을 단축하는 철학. 이는 애자일 접근 방식의 중요한 부분입니다.

장애 격리 경계

장애 영향을 제한하고 워크로드의 복원력을 개선하는 데 도움이 되는 가용 영역 AWS 리전, 컨트롤 플레인 또는 데이터 플레인과 같은 경계 AWS 클라우드 자세한 내용은 [AWS 장애 격리](#) 경계를 참조하십시오.

기능 브랜치

[브랜치를](#) 참조하십시오.

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 [다음은 AWS 사용한 기계 학습 모델 해석 가능성을](#) 참조하십시오.

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 활용할 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.

FGAC

[세분화된 액세스 제어](#)를 참조하십시오.

세분화된 액세스 제어(FGAC)

여러 조건을 사용하여 액세스 요청을 허용하거나 거부합니다.

플래시컷 마이그레이션

단계별 접근 방식 대신 [변경 데이터 캡처를 통한 지속적인 데이터](#) 복제를 통해 최단 시간에 데이터를 마이그레이션하는 데이터베이스 마이그레이션 방법입니다. 목표는 가동 중지 시간을 최소화하는 것입니다.

G

지리적 차단

[지리적 제한](#)을 참조하십시오.

지리적 제한(지리적 차단)

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

Gitflow 워크플로

하위 환경과 상위 환경이 소스 코드 리포지토리의 서로 다른 브랜치를 사용하는 방식입니다.

Gitflow 워크플로는 레거시로 간주되며 [트렁크 기반 워크플로는](#) 현대적이고 선호되는 접근 방식입니다.

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, 이들은, Amazon AWS Config AWS Security Hub GuardDutyAWS Trusted Advisor, Amazon Inspector 및 사용자 지정 AWS Lambda 검사를 사용하여 구현됩니다.

H

하

[고가용성을](#) 확인하세요.

이기종 데이터베이스 마이그레이션

다른 데이터베이스 엔진을 사용하는 대상 데이터베이스로 소스 데이터베이스 마이그레이션(예: Oracle에서 Amazon Aurora로) 이기종 마이그레이션은 일반적으로 리아키텍트 작업의 일부이며 스키마를 변환하는 것은 복잡한 작업일 수 있습니다. AWS는 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.

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. 긴급성 때문에 핫픽스는 일반적으로 일반적인 DevOps 릴리스 워크플로 외부에서 만들어집니다.

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

[인프라를 코드로 보세요.](#)

ID 기반 정책

하나 이상의 IAM 보안 주체에 연결된 정책으로, AWS 클라우드 환경 내에서 IAM 보안 주체의 권한을 정의합니다.

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 인터넷을 참조하십시오.](#)

불변의 인프라

기존 인프라를 업데이트, 패치 또는 수정하는 대신 프로덕션 워크로드용 새 인프라를 배포하는 모델입니다. [변경 불가능한 인프라는 기본적으로 변경 가능한 인프라보다 더 일관되고 안정적이며 예측 가능합니다.](#) 자세한 내용은 Well-Architected AWS 프레임워크의 [변경 불가능한 인프라를 사용한 배포](#) 모범 사례를 참조하십시오.

인바운드(수신) VPC

AWS 다중 계정 아키텍처에서 애플리케이션 외부의 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.

중분 마이그레이션

한 번에 전체 전환을 수행하는 대신 애플리케이션을 조금씩 마이그레이션하는 전환 전략입니다. 예를 들어, 처음에는 소수의 마이크로서비스나 사용자만 새 시스템으로 이동할 수 있습니다. 모든 것

이 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.

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 참조하세요.

검사 VPC

AWS 다중 계정 아키텍처에서 동일한 또는 다른 AWS 리전에 있는 VPC, 인터넷 및 온프레미스 네트워크 간의 네트워크 트래픽 검사를 관리하는 중앙 집중식 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.

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 참조하세요.

해석력

모델의 예측이 입력에 따라 어떻게 달라지는지를 사람이 이해할 수 있는 정도를 설명하는 기계 학습 모델의 특성입니다. 자세한 내용은 [Machine learning model interpretability with AWS](#)를 참조하세요.

IoT

[사물 인터넷을 참조하십시오.](#)

IT 정보 라이브러리(TIL)

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

IT 서비스 관리(TSM)

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

ITIL

[IT 정보 라이브러리를](#) 참조하십시오.

ITSM

[IT 서비스 관리를](#) 참조하십시오.

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 확장성과 안전성을 갖춘 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 [Setting up a secure and scalable multi-account AWS environment](#)를 참조하세요.

대규모 마이그레이션

300대 이상의 서버 마이그레이션입니다.

LBAC

[레이블 기반 액세스 제어를](#) 참조하십시오.

최소 권한

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

리프트 앤드 시프트

[7 R](#)을 참조하십시오.

리틀 엔디안 시스템

가장 덜 중요한 바이트를 먼저 저장하는 시스템입니다. [엔디안도](#) 참조하십시오.

하위 환경

[환경 참조.](#)

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 참조하세요.

기본 브랜치

[브랜치](#) 참조.

매니지드 서비스

AWS 서비스인프라 계층, 운영 체제 및 플랫폼을 AWS 운영하며 사용자는 엔드포인트에 액세스하여 데이터를 저장하고 검색합니다. 관리형 서비스의 예로는 아마존 심플 스토리지 서비스 (Amazon S3) 와 아마존 DynamoDB가 있습니다. 이러한 서비스를 추상화된 서비스라고도 합니다.

MAP

[Migration Acceleration 프로그램](#)을 참조하십시오.

기구

도구를 만들고 도구 채택을 유도한 다음 결과를 검토하여 조정하는 전체 프로세스입니다. 메커니즘은 작동하면서 자체적으로 강화되고 개선되는 사이클입니다. 자세한 내용은 AWS Well-Architected [프레임워크에서의 메커니즘 구축](#)을 참조하십시오.

멤버 계정

AWS Organizations의 조직에 속하는 관리 계정 이외의 모든 AWS 계정입니다. 하나의 계정은 한 번에 하나의 조직 멤버만 될 수 있습니다.

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 영역

에 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 통합](#)을 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 [AWS에서 마이크로서비스 구현](#)을 참조하세요.

Migration Acceleration Program(MAP)

조직이 클라우드로 전환하기 위한 강력한 운영 기반을 구축하고 마이그레이션 초기 비용을 상쇄할 수 있도록 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 운영, 비즈니스 분석가 및 소유자, 마이그레이션 엔지니어, 개발자, 스프린트에서 일하는 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 참조하세요.

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. 예를 들어, AWS Application Migration Service를 사용하여 Amazon EC2로 마이그레이션을 리호스팅합니다.

Migration Portfolio Assessment(MPA)

AWS 클라우드로 마이그레이션의 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 컨설턴트에게 무료로 제공됩니다.

마이그레이션 준비 상태 평가(MRA)

AWS CAF를 사용하여 조직의 클라우드 준비 상태에 대한 인사이트를 얻고, 장단점을 파악하고, 파악된 격차를 해소하기 위한 실행 계획을 수립하는 프로세스입니다. 자세한 내용은 [마이그레이션 준비 가이드](#)를 참조하세요. MRA는 [AWS 마이그레이션 전략](#)의 첫 번째 단계입니다.

마이그레이션 전략

워크로드를 AWS 클라우드로 마이그레이션하는 데 사용되는 접근 방식입니다. 자세한 내용은 이 용어집의 [7R](#) 항목을 참조하고 대규모 마이그레이션을 [가속화하기 위한 조직 동원을](#) 참조하십시오.

ML

[기계 학습을 참조하십시오.](#)

MPA

[마이그레이션 포트폴리오 평가를](#) 참조하십시오.

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 애플리케이션 현대화 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 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 워크로드를 위해 기존 인프라를 업데이트하고 수정하는 모델입니다. 일관성, 안정성 및 예측 가능성을 개선하기 위해 AWS Well-Architected Framework는 [변경 불가능한](#) 인프라를 모범 사례로 사용할 것을 권장합니다.

O

OAC

[원본 액세스 제어를 참조하십시오.](#)

좋아요

[원본 액세스 ID를 참조하십시오.](#)

OCM

[조직 변경 관리를 참조하십시오.](#)

오프라인 마이그레이션

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

이

[운영 통합을 참조하십시오.](#)

안녕하세요.

[운영 수준 계약을](#) 참조하십시오.

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.

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 발생 가능한 실패의 범위를 이해, 평가, 예방 또는 줄이는 데 도움이 되는 질문 및 관련 모범 사례로 구성된 체크리스트입니다. 자세한 내용은 Well-Architected AWS 프레임워크의 [운영 준비 상태 검토 \(ORR\)](#) 를 참조하십시오.

운영 통합(OI)

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

조직 트레일

AWS Organizations의 조직 내 모든 AWS 계정에 대한 모든 이벤트를 로깅하기 위해 AWS CloudTrail에서 생성하는 트레일입니다. 이 트레일은 조직에 속한 각 AWS 계정에 생성되고 각 계정의 활동을 추적합니다. 자세한 내용은 설명서의 [조직을 위한 트레일 만들기를](#) 참조하십시오.

CloudTrail

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 변화 속도 때문에 이 프레임워크를 인력 가속화라고 합니다. 자세한 내용은 [사용 가이드](#)를 참조하세요.

오리진 액세스 제어(OAC)

CloudFront에서는 Amazon Simple Storage Service (Amazon S3) 콘텐츠의 보안을 위해 액세스를 제한하는 향상된 옵션을 제공합니다. OAC는 모든 AWS 리전의 모든 S3 버킷, AWS KMS(SSE-KMS)를 사용한 서버측 암호화, S3 버킷에 대한 동적 PUT 및 DELETE 요청을 지원합니다.

오리진 액세스 ID(OAI)

CloudFront에서는 Amazon S3 콘텐츠 보안을 위해 액세스를 제한하는 옵션입니다. OAI를 사용하면 Amazon S3가 인증할 수 있는 보안 주체를 CloudFront 생성합니다. 인증된 보안 주체는 특정 배포를 통해서만 S3 버킷의 콘텐츠에 액세스할 수 있습니다. CloudFront 더 세분화되고 향상된 액세스 제어를 제공하는 [OAC](#)도 참조하세요.

또는

[운영 준비 상태](#) 검토를 참조하십시오.

아웃바운드(송신) VPC

AWS 다중 계정 아키텍처에서 애플리케이션 내에서 시작되는 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 참조하세요.

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 정보를](#) 참조하십시오.

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.

정책

[권한을 정의 \(ID 기반 정책 참조\)](#) 하거나, [액세스 조건을 지정 \(리소스 기반 정책 참조\)](#) 하거나, [조직의 모든 계정에 대한 최대 권한을 정의 AWS Organizations \(서비스 제어 정책 참조\)](#) 할 수 있는 [개체입니다](#).

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 스토어를 사용하면 마이크로서비스를 더 쉽게 구현하고 성능과 확장성을 높일 수 있습니다. 자세한 내용은 [마이크로서비스에서 데이터 지속성 활성화](#)를 참조하세요.

포트폴리오 평가

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

조건자

일반적으로 조항에 있는 true false OR를 반환하는 쿼리 조건입니다. WHERE

조건부 푸시다운

전송하기 전에 쿼리의 데이터를 필터링하는 데이터베이스 쿼리 최적화 기법입니다. 이렇게 하면 관계형 데이터베이스에서 검색하고 처리해야 하는 데이터의 양이 줄어들고 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 참조하세요.

보안 주체

작업을 수행하고 리소스에 액세스할 수 있는 AWS의 객체입니다. 이 엔터티는 일반적으로 AWS 계정의 루트 사용자, IAM 역할 또는 사용자입니다. 자세한 내용은 IAM 설명서의 [역할 용어 및 개념](#)의 보안 주체를 참조하세요.

개인 정보 보호 중심 설계

전체 엔지니어링 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 작업](#)을 참조하세요.

사전 예방적 제어

규정을 준수하지 않는 리소스의 배포를 방지하도록 설계된 [보안 제어입니다](#). 이러한 컨트롤은 리소스를 프로비저닝하기 전에 리소스를 스캔합니다. 리소스가 컨트롤과 호환되지 않으면 프로비저닝되지 않습니다. 자세한 내용은 AWS Control Tower 설명서의 [컨트롤 참조 안내서](#)를 참조하고 보안 제어 구현의 [사전 제어를](#) 참조하십시오. AWS

프로덕션 환경

[환경을](#) 참조하십시오.

가명화

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

Q

쿼리 계획

SQL 관계형 데이터베이스 시스템의 데이터에 액세스하는 데 사용되는 일련의 단계 (예: 지침).

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

RACI ([책임](#), [책임](#), [상담](#), [정보 제공](#)) 를 참조하십시오.

랜섬웨어

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

RASCI 매트릭스

[책임](#), [책임](#), [상담](#), [정보 제공 \(RACI\)](#) 을 참조하십시오.

RCAC

[행 및 열 액세스 제어를](#) 참조하십시오.

읽기 전용 복제본

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

재설계

[7 R을](#) 참조하십시오.

Recovery Point Objective(RPO)

마지막 데이터 복구 시점 이후 허용되는 시간입니다. 이에 따라 마지막 복구 시점과 서비스 중단 시점 사이에 허용 가능한 데이터 손실이 결정됩니다.

Recovery Time Objective(RTO)

서비스 중단과 서비스 복구 사이에 허용되는 최대 지연 시간입니다.

리팩터링

[7 R을](#) 참조하십시오.

리전

AWS 리소스를 지리적 영역에 모아 놓은 것입니다. 각 AWS 리전은 내결함성, 안정성 및 복원력을 제공하기 위해 격리되어 있으며 다른 리전과는 독립적입니다. 자세한 내용은 AWS 일반 참조의 [AWS 리전 관리](#)를 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

[7 R을](#) 참조하십시오.

release

배포 프로세스에서 변경 사항을 프로덕션 환경으로 승격시키는 행위입니다.

고쳐 놓다

[7 R을](#) 참조하십시오.

리플랫폼

[7 R을 참조하십시오.](#)

환매

[7 R을 참조하십시오.](#)

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 조연자 (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 Implementing security controls on AWS의 [Responsive controls](#)를 참조하세요.

retain

[7 R을 참조하십시오.](#)

은퇴

[7 R을 참조하십시오.](#)

회전

공격자가 자격 증명에 액세스하는 것을 더 어렵게 만들기 위해 [암호를](#) 주기적으로 업데이트하는 프로세스입니다.

행 및 열 액세스 제어(RCAC)

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

RPO

[복구 지점 목표를 참조하십시오.](#)

RPO

[복구 시간 목표를 참조하십시오.](#)

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 (IdPs) 가 사용하는 개방형 표준입니다. 이 기능은 페더레이션형 AWS Single Sign-On(SSO)을 활성화하므로 조직의 모든 멤버에 대해 IAM 사용자를 생성하지 않아도 사용자가 AWS Management Console에 로그인하거나 AWS API 작업을 직접적으로 호출할 수 있습니다. SAML 2.0 기반 페더레이션에 대한 자세한 내용은 IAM 설명서의 [SAML 2.0 기반 페더레이션 정보](#)를 참조하세요.

SCP

[서비스 제어 정책을](#) 참조하십시오.

secret

에는 AWS Secrets Manager 암호화된 형태로 저장하는 비밀번호나 사용자 자격 증명과 같은 기밀 또는 제한된 정보. 비밀 값과 해당 메타데이터로 구성됩니다. 비밀 값은 바이너리, 단일 문자열 또는 여러 문자열일 수 있습니다. 자세한 내용은 Secrets Manager 문서의 [시크릿](#)을 참조하십시오.

보안 제어

위험 행위자가 보안 취약성을 악용하는 능력을 방지, 탐지 또는 감소시키는 기술적 또는 관리적 가드레일입니다. [보안 제어에는 예방적, 탐정적, 대응적, 사전 예방적 등 네 가지 기본 유형이 있습니다.](#)

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 대응 자동화

보안 이벤트에 자동으로 대응하거나 보안 이벤트를 해결하도록 설계된 사전 정의되고 프로그래밍 된 조치입니다. 이러한 자동화는 보안 모범 사례를 구현하는 데 도움이 되는 [탐지](#) 또는 [대응형](#) 보안 제어 역할을 합니다. AWS 자동 응답 조치의 예로는 VPC 보안 그룹 수정, Amazon EC2 인스턴스 패치, 자격 증명 교체 등이 있습니다.

서버측 암호화

데이터를 수신하는 AWS 서비스가 대상에서 데이터를 암호화하는 것입니다.

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 정책](#)을 참조하세요.

서비스 엔드포인트

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

서비스 수준에 관한 계약(SLA)

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

서비스 수준 지표 (SLI)

오류율, 가용성 또는 처리량과 같은 서비스의 성능 측면을 측정하는 것입니다.

서비스 수준 목표 (SLO)

[서비스 수준 지표로 측정되는 서비스 상태를 나타내는 대상 지표입니다.](#)

공동 책임 모델

클라우드 보안 및 규정 준수를 위해 AWS와 공유하는 책임을 설명하는 모델입니다. AWS는 클라우드의 보안을 담당하고, 사용자는 클라우드에서 보안을 담당합니다. 자세한 내용은 [공동 책임 모델](#)을 참조하세요.

시앰

[보안 정보 및 이벤트 관리 시스템](#)을 참조하십시오.

단일 장애 지점 (SPOF)

응용 프로그램의 중요한 단일 구성 요소에서 발생한 오류로 인해 시스템이 중단될 수 있습니다.

SLA

SLA ([서비스 수준 계약](#)) 를 참조하십시오.

SLI

[서비스 수준](#) 표시기를 참조하십시오.

SLO

[서비스 수준 목표를](#) 참조하십시오.

split-and-seed 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 의 [애플리케이션 현대화를 위한 단계별 접근 방식을 참조하십시오. AWS 클라우드](#)

SPOF

[단일 장애 지점](#) 보기

스타 스키마

하나의 큰 팩트 테이블을 사용하여 트랜잭션 또는 측정 데이터를 저장하고 하나 이상의 작은 차원 테이블을 사용하여 데이터 속성을 저장하는 데이터베이스 구성 구조입니다. 이 구조는 [데이터 웨어하우스에서](#) 사용하거나 비즈니스 인텔리전스 용도로 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 참조하세요.

서브넷

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

대칭 암호화

동일한 키를 사용하여 데이터를 암호화하고 복호화하는 암호화 알고리즘입니다.

합성 테스트

잠재적 문제를 감지하거나 성능을 모니터링하기 위해 사용자 상호 작용을 시뮬레이션하는 방식으로 시스템을 테스트합니다. [Amazon CloudWatch Synthetics](#)를 사용하여 이러한 테스트를 생성할 수 있습니다.

T

tags

AWS 리소스를 구성하는 메타데이터 역할을 하는 키-값 페어입니다. 태그를 사용하면 리소스를 손쉽게 관리, 식별, 정리, 검색 및 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

[환경을 참조하십시오.](#)

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.

전송 게이트웨이

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

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 Organizations의 조직과 해당 계정에서 작업을 수행하도록 지정하는 서비스에 권한을 부여하는 것입니다. 신뢰할 수 있는 서비스는 필요할 때 각 계정에 서비스 연결 역할을 생성하여 관리 작업을 수행합니다. 자세한 내용은 AWS Organizations 설명서의 [다른 AWS 서비스와 함께 AWS Organizations 사용](#)을 참조하세요.

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 먹을 수 있는 소규모 DevOps 팀이죠. 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 참조하세요.

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

[환경을](#) 보세요.

V

정리

스토리지를 회수하고 성능을 향상시키기 위해 증분 업데이트 후 정리 작업을 수반하는 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 참조하세요.

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.

윈도우 함수

현재 레코드와 어떤 식으로든 관련된 행 그룹에 대해 계산을 수행하는 SQL 함수입니다. 윈도우 함수는 이동 평균을 계산하거나 현재 행의 상대적 위치를 기반으로 행 값에 액세스하는 등의 작업을 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.

원

[한 번 쓰고, 많이 읽으세요.](#)

WQF

[AWS 워크로드 검증 프레임워크](#)를 참조하십시오.

한 번 작성하고 여러 번 읽기 (WORM)

데이터를 한 번 쓰고 데이터가 삭제되거나 수정되지 않도록 하는 스토리지 모델입니다. 인증된 사용자는 필요한 만큼 데이터를 여러 번 읽을 수 있지만 변경할 수는 없습니다. 이 데이터 스토리지 인프라는 [변경할 수 없는](#) 것으로 간주됩니다.

Z

제로데이 익스플로잇

[제로데이](#) 취약점을 악용하는 공격 (일반적으로 멀웨어)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.

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.