aws

최신 데이터 중심 아키텍처 사용 사례 설계 및 구현 모범 사례

AWS 권장 가이드

AWS 권장 가이드: 최신 데이터 중심 아키텍처 사용 사례 설계 및 구현 모범 사례

Table of Contents

최신 데이터 중심 아키텍처 사용 사례 설계 및 구현 모범 사례

Apoorva Patrikar, Amazon Web Services(AWS)

2023년 5월(문서 기록)

조직은 데이터 요구 사항을 중심으로 IT 인프라, 애플리케이션 개발 및 비즈니스 프로세스를 설계하는 데이터 중심 아키텍처를 수용하기 위해 애플리케이션 중심 아키텍처에서 벗어나고 있습니다. 데이터 중심 아키텍처에서 데이터는 핵심 IT 자산이며, 데이터를 최적화하도록 IT 시스템과 프로세스를 설계합니다.

이 가이드에서는 사용 사례에 맞는 최신 데이터 중심 아키텍처를 설계하는 모범 사례를 제공합니다. 이러한 모범 사례를 사용하여 데이터 파이프라인과 해당 파이프라인을 지원하는 데이터 엔지니어링 작업을 현대화할 수 있습니다. 또한 이 가이드에서는 데이터 파이프라인의 데이터 수명 주기에 대한 개요도 제공합니다. 이 수명 주기를 이해하면 데이터를 최적화하는 데이터 파이프라인을 빌드할 수 있습니다.

이 가이드를 사용하여 데이터 파이프라인의 데이터 중심 아키텍처를 설계할 때 많은 조직이 직면하는 다음과 같은 문제를 해결할 수 있습니다.

- 동일한 데이터세트의 여러 버전 저장 자제 - 데이터를 여러 번 자주 처리하는 것은 드문 일이 아니지만 이 접근 방식에는 제한 사항이 있습니다. 실제로 데이터를 여러 번 처리하지 않으면 리소스 집약도를 낮추고 비용 효율적인 경우가 많습니다. 이 가이드에서는 처리된 데이터를 여러 단계로 저장하는 데 중점을 두는 다른 접근 방식을 취할 경우 이점을 보여줍니다.
- 데이터 레이크 수용을 주저함 - 데이터 레이크에 관한 마케팅 클레임을 분류하기 어려울 수 있으며, 조직에 데이터 레이크를 IT 시스템 및 프로세스에 통합하는 데 필요한 기술과 리소스가 있는지 파악하는 것도 쉽지 않을 수 있습니다. 이 가이드는 데이터 레이크가 데이터 중심 아키텍처에서 어떻게 유용한 구성 요소가 될 수 있는지 이해하는 데 도움이 될 수 있습니다.
- 충분한 데이터 엔지니어 고용 - 시장 추세에 따르면 데이터 과학자는 올바른 데이터 엔지니어링 기술이 없더라도 많은 조직에서 데이터 엔지니어링 작업을 수행해야 합니다. 이러한 기술 격차는 시장 출시 계획에 영향을 미칠 수 있습니다. 이 가이드는 데이터 중심 아키텍처를 설계하는 데 필요한 데이터 엔지니어링 기술을 더 잘 이해하는 데 도움이 될 수 있습니다.
- 수평 처리를 위한 AWS 서비스 사용 관련 지식 부족 - 수평 또는 분산 처리를 통해 클러스터는 태스크를 여러 노드에 매핑하고 사용자에게 투명하게 전송하기 전에 결과를 수집하여 데이터 청크를 병렬로 처리할 수 있습니다. 수평 처리를 향한 움직임은 데이터를 보고 처리하는 방식을 중심으로 한 변화입니다. 이러한 전환은 애플리케이션 로직 또는 애플리케이션 자체뿐만 아니라 조직이 데이터를 사용하는 방식에도 영향을 미칩니다. 예를 들어 수평 처리는 중앙 스토리지, 작업 배포 및 모듈화

에 영향을 미칩니다. 수평 처리는 읽기 및 쓰기 작업에서 더 큰 데이터 청크를 선호합니다. 이 가이드에서는 데이터 파이프라인에서 수평적 처리가 작동하는 방법을 설명합니다.

목표 비즈니스 성과

이 가이드는 사용자와 조직이 다음과 같은 비즈니스 성과를 달성하는 데 도움이 될 수 있습니다.

- 데이터 아키텍처 프로젝트 계획 개선
- 데이터 수명 주기에 대한 보다 안전한 거버넌스
- 데이터 파이프라인 프로젝트의 더 빠른 배포
- 데이터 파이프라인 솔루션의 고품질 아키텍처 및 엔지니어링

데이터 엔지니어링 원칙

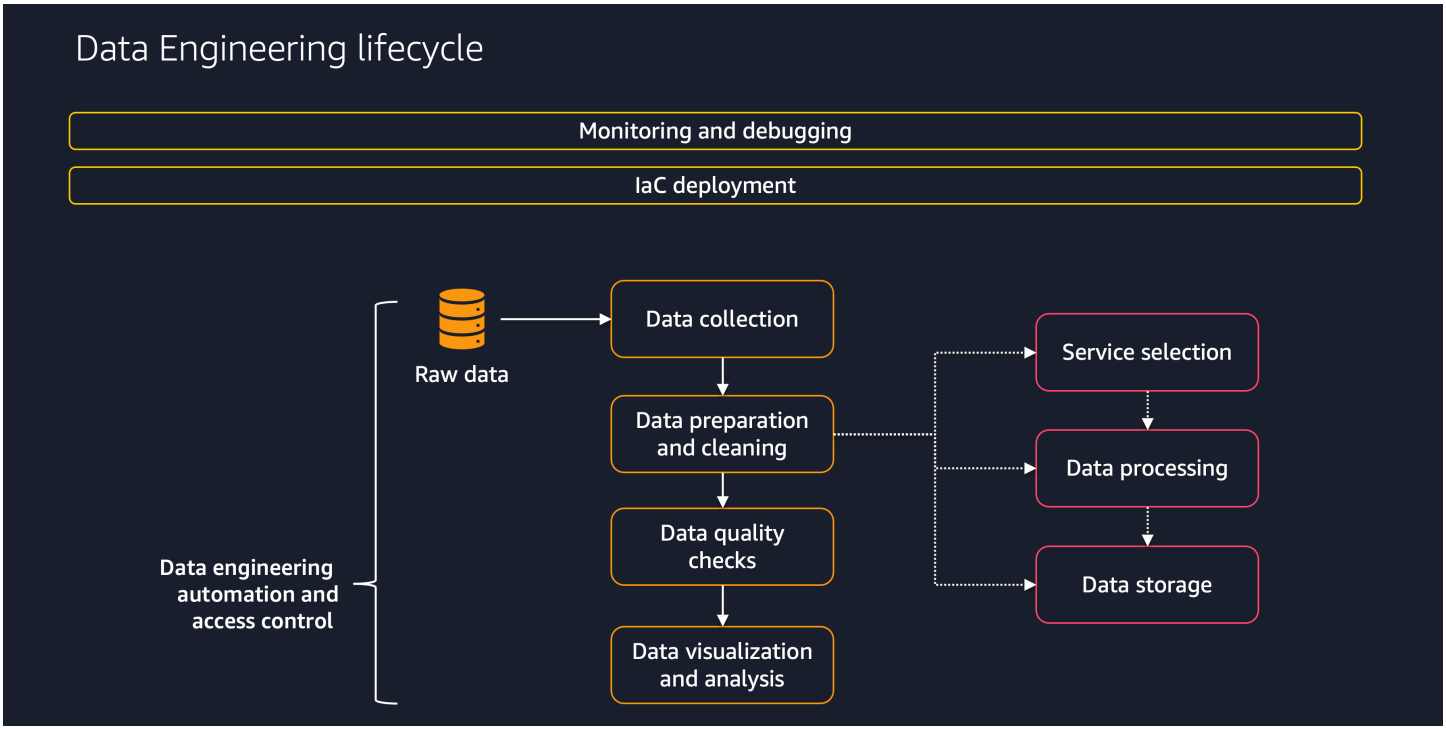
최신 데이터 파이프라인을 위한 아키텍처를 빌드할 때 다음 표의 원칙을 채택하는 것이 좋습니다.

원칙	예시	사용 사례:
유연성	마이크로서비스 사용	FastGo enjoys flexibility and scalability with a microservices architecture on AWS(AWS 사례 연구)
재현성	코드형 인프라(IaC)를 사용하여 서비스 배포	Part 3: How NatWest Group built auditable, reproducible, and explainable ML models with Amazon SageMaker(AWS 기계 학습 블로그)
재사용성	공유 방식으로 라이브러리 및 참조 사용	Create and reuse governed datasets in Amazon QuickSight with new Dataset-as-a-Source feature(AWS 빅 데이터 블로그)
확장성	모든 데이터 로드를 수용할 수 있는 서비스 구성 선택	AWS 클라우드에서 성장 및 규모 조정을 위해 데이터 레이크 설계(AWS 권장 가이드)
감사 가능성	로그, 버전 및 종속성을 사용하여 감사 추적 유지	How Parametric Built Audit Surveillance using AWS Data Lake Architecture(AWS 아키텍처 블로그)

데이터 수명 주기

데이터 파이프라인을 빌드하려면 먼저 파일 서버, 데이터베이스, 스토리지 버킷 또는 API 직접 호출과 같은 외부 또는 내부 데이터 소스에서 AWS로 데이터를 수집해야 합니다. 수집된 데이터는 익명화, 열 삭제 또는 데이터 정리와 같은 변환을 거치거나 거치지 않을 수 있습니다.

이 섹션에서는 다음 다이어그램과 같이 데이터 수명 주기 프로세스의 단계에 대한 개요를 제공합니다.

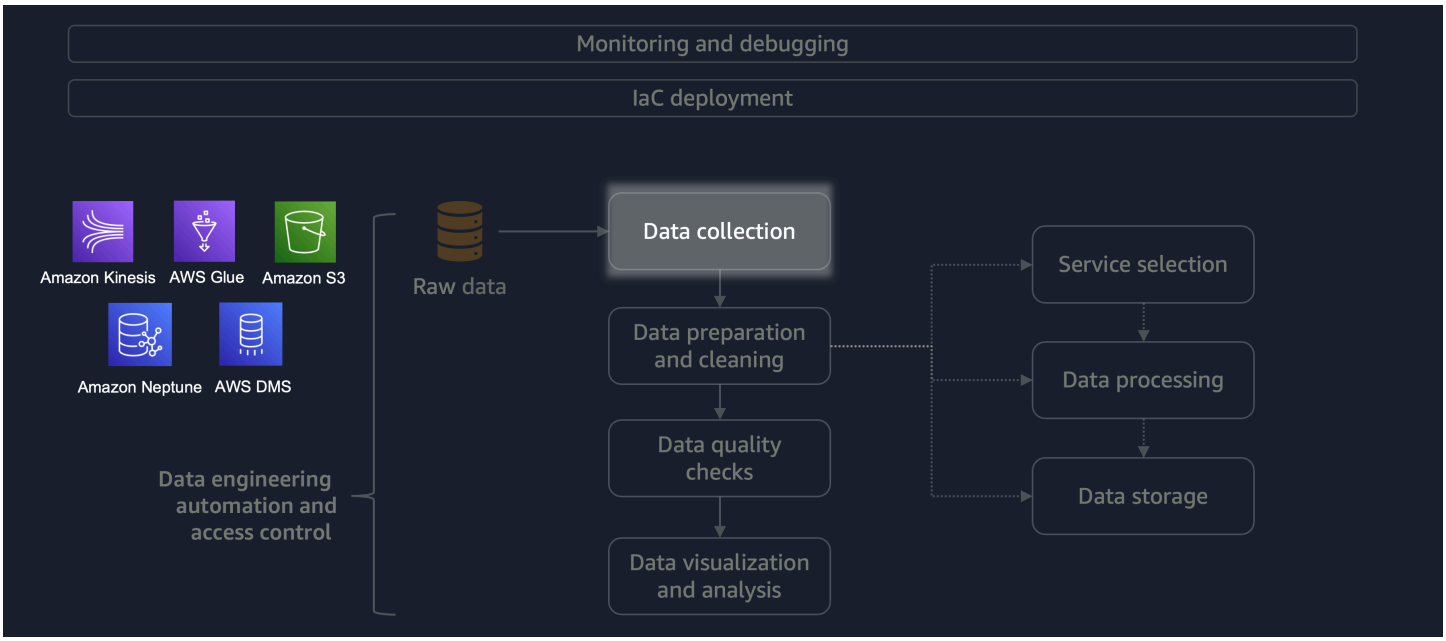


이 단계에는 다음이 포함됩니다.

- 데이터 수집
- 데이터 준비 및 정리
- 데이터 품질 검사
- 데이터 시각화 및 분석
- 모니터링 및 디버깅
- IaC 배포
- 자동화 및 액세스 제어

데이터 수집

AWS 내 다양한 소스에서 데이터를 수집할 수 있지만 사용 사례에 적합한 데이터 수집 도구를 선택하는 것이 중요합니다. 다음 다이어그램에서는 데이터 수집 단계가 데이터 엔지니어링 자동화 및 액세스 제어 수명 주기에 어떻게 부합하는지 보여줍니다.



AWS는 다음과 같은 데이터 수집 도구를 제공합니다.

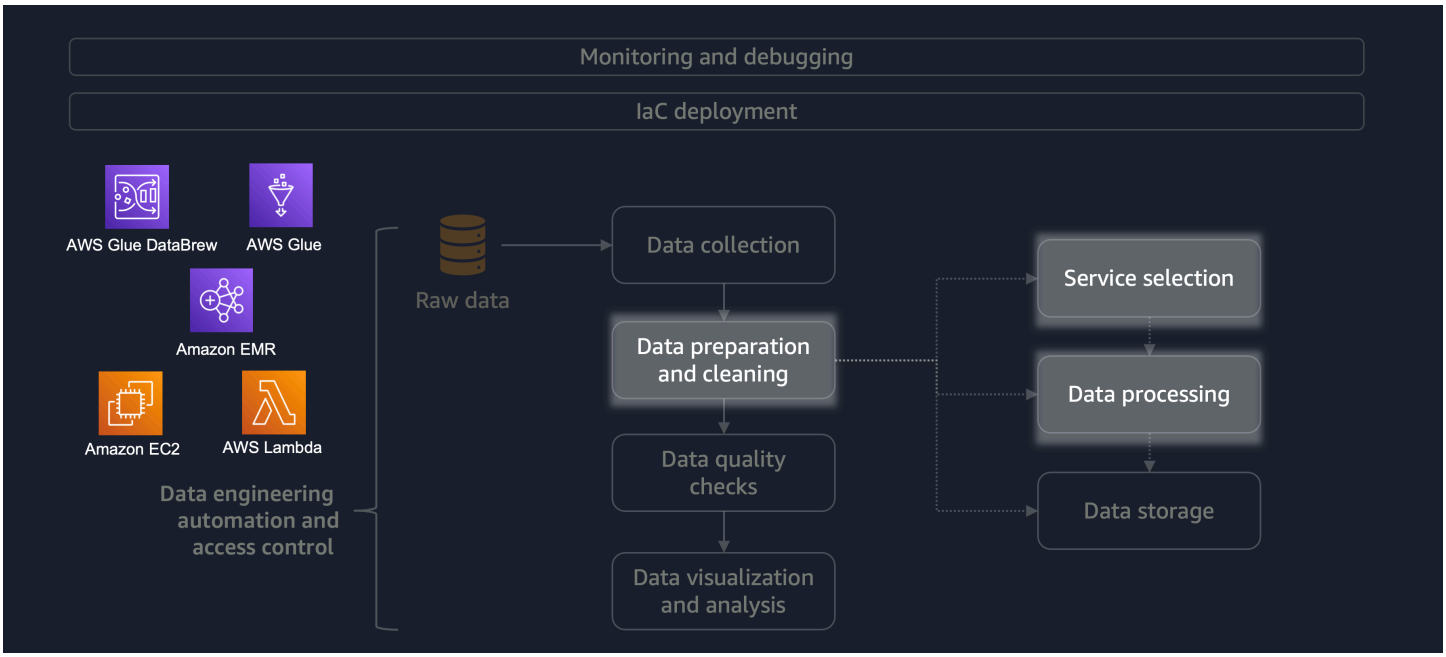
- Amazon Kinesis는 스트리밍 데이터를 수집하는 데 도움이 됩니다. 또한 Kinesis는 원활한 통합 및 처리 기능도 제공합니다.
- AWS Database Migration Service(AWS DMS)를 사용하면 관계형 데이터베이스에서 데이터를 수집할 수 있습니다. AWS DMS는 AWS에서 호스팅되는 Amazon Simple Storage Service(Amazon S3)와 같은 데이터베이스 서비스 및 온프레미스 간의 구성 옵션과 직접 연결을 보유합니다.
- AWS Glue는 비정형 데이터를 수집하는 데 도움이 되는 추출, 전환, 적재(ETL) 도구입니다.

Amazon S3를 스토리지로 사용하여 비정형 또는 반정형 데이터를 수집하는 몇 가지 사용 사례가 있습니다. 예를 들어 제조 현장의 데이터 수집 사용 사례에서는 기계 기록 데이터를 XML 파일로, 이벤트 데이터를 JSON 파일로, 관계형 데이터베이스에서 구매 데이터를 수집할 수 있어야 합니다. 이 사용 사례에서는 세 데이터 소스를 모두 조인해야 할 수도 있습니다.

데이터 수집 프로세스를 시작하기 전에 수집해야 하는 데이터를 이해한 다음이 데이터를 수집하는 데 적합한 도구를 선택하는 것이 좋습니다.

데이터 준비 및 정리

데이터 준비 및 정리는 데이터 수명 주기에서 가장 중요하지만 가장 시간이 많이 걸리는 단계 중 하나입니다. 다음 다이어그램에서는 데이터 준비 및 정리 단계가 데이터 엔지니어링 자동화 및 액세스 제어 수명 주기에 어떻게 부합하는지 보여줍니다.



다음은 데이터 준비 또는 정리에 대한 몇 가지 예제입니다.

- 텍스트 열을 코드에 매핑
- 빈 열 무시
- 빈 데이터 필드를 `0`, `None` 또는 `''`로 채우기
- 개인 식별 정보(PII) 마스킹 또는 익명화

다양한 데이터가 있는 대규모 워크로드가 있는 경우 데이터 준비 및 정리 작업에 Amazon EMR 또는 AWS Glue를 사용하는 것이 좋습니다. Amazon EMR 및 AWS Glue는 모두 비정형, 반정형 및 관계형 데이터에서 사용할 수 있으며, 둘 다 Apache Spark를 사용하여 `DataFrame` 또는 `DynamicFrame`을 생성하여 수평 처리로 작업할 수 있습니다. 또한 AWS Glue DataBrew를 사용하여 노코드 접근 방식으로 데이터를 정리하고 처리할 수 있습니다. 또한 DataBrew는 데이터세트를 열 통계로 프로파일링하고, 데이터 리니지를 제공하며, 모든 열 또는 지정된 열에 대한 데이터 품질 규칙을 포함할 수 있습니다.

분산 처리가 필요하지 않고 15분 이내에 완료할 수 있는 소규모 워크로드의 경우 데이터 준비 및 정리에 AWS Lambda를 사용하는 것이 좋습니다. Lambda는 소규모 워크로드를 위한 비용 효율적인 경량 옵션입니다. 클라우드에 들어갈 수 없는 고도의 보안 데이터의 경우 AWS Outposts 서버를 사용하여

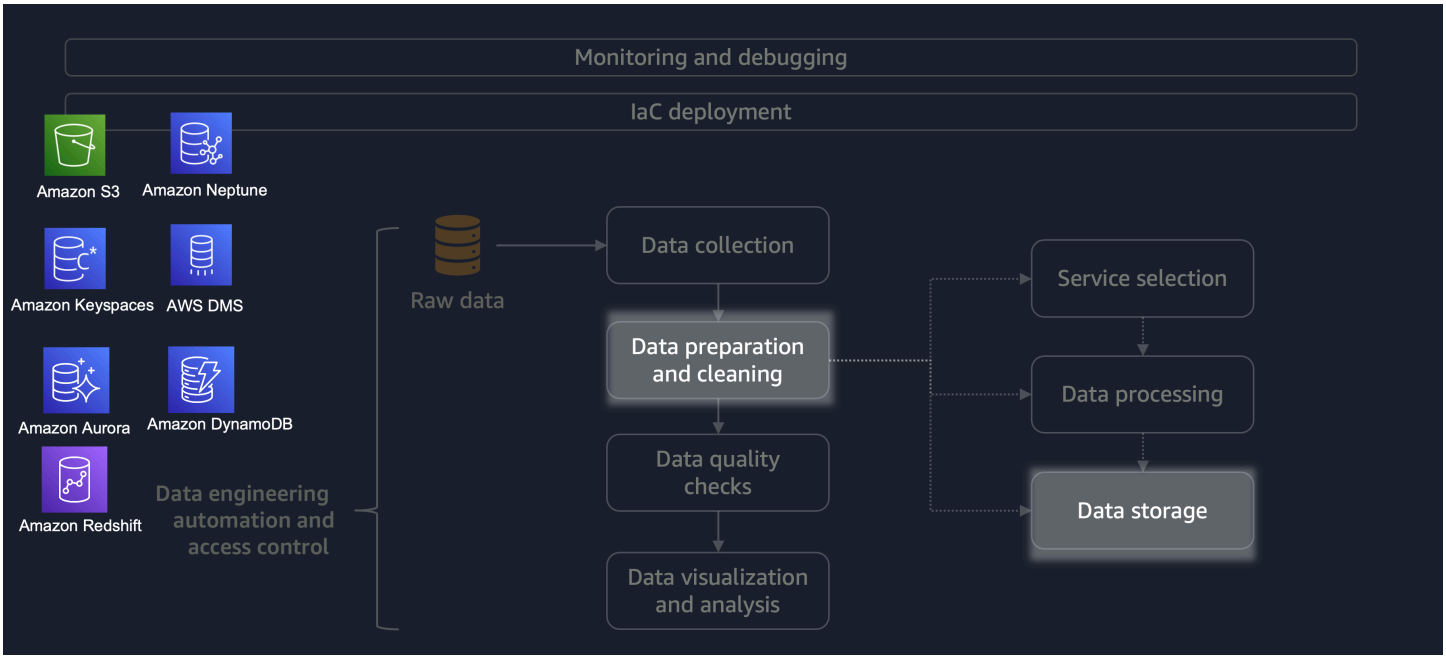
Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스에서 데이터 익명화를 수행하는 것이 좋습니다.

데이터 준비 및 정리에 적합한 AWS 서비스를 선택하고 선택과 관련된 장단점을 이해하는 것이 중요합니다. 예를 들어 AWS Glue, DataBrew 및 Amazon EMR 중에서 선택하는 시나리오를 고려합니다. AWS Glue는 ETL 작업이 빈번하지 않은 경우에 적합합니다. 자주 발생하지 않는 작업은 하루에 한 번, 일주일에 한 번 또는 한 달에 한 번 수행됩니다. 또한 데이터 엔지니어가 일반적으로 Spark 코드 작성(빅 데이터 사용 사례의 경우) 또는 스크립팅에 능숙하다고 가정할 수 있습니다. 작업이 더 빈번한 경우 AWS Glue를 지속적으로 실행하면 비용이 많이 들 수 있습니다. 이 경우 Amazon EMR은 분산 처리 기능을 제공하고 서버리스 버전과 서버 기반 버전을 모두 제공합니다. 데이터 엔지니어에게 적절한 기술이 없거나 결과를 빠르게 전달해야 하는 경우 DataBrew가 좋은 옵션입니다. DataBrew는 코드 개발 노력을 줄이고 데이터 준비 및 정리 프로세스의 속도를 높일 수 있습니다.

처리가 완료되면 ETL 프로세스의 데이터가 AWS에 저장됩니다. 스토리지 선택은 처리하는 데이터 유형에 따라 달라집니다. 예를 들어 그래프 데이터, 키 값 페어 데이터, 이미지, 텍스트 파일 또는 관계형 정형 데이터와 같은 비관계형 데이터로 작업할 수 있습니다.

다음 다이어그램과 같이 데이터 스토리지에 다음 AWS 서비스를 사용할 수 있습니다.

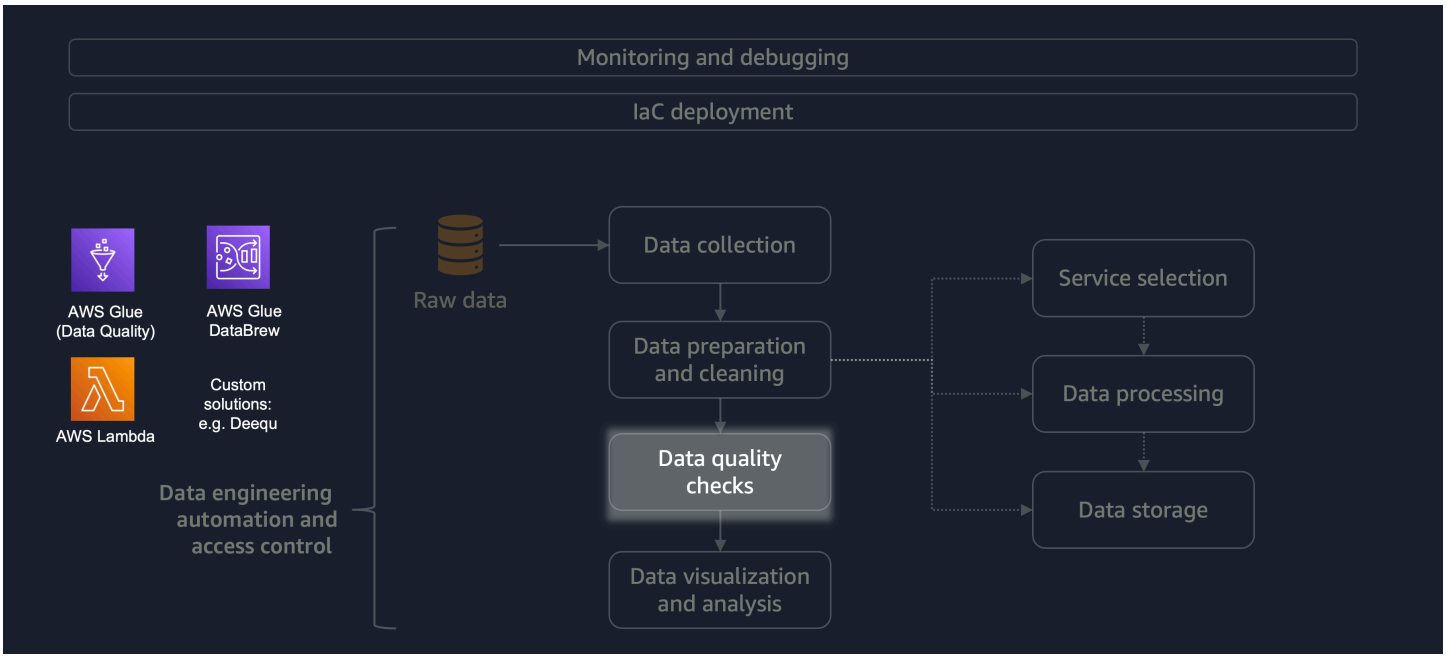
- Amazon S3는 비정형 데이터 또는 반정형 데이터(예: Apache Parquet 파일, 이미지 및 비디오)를 저장합니다.
- Amazon Neptune은 SPARQL 또는 GREMLIN을 사용하여 쿼리할 수 있는 그래프 데이터세트를 저장합니다.
- Amazon Keyspaces(Apache Cassandra용)는 Apache Cassandra와 호환되는 데이터세트를 저장합니다.
- Amazon Aurora는 관계형 데이터세트를 저장합니다.
- Amazon DynamoDB는 키 값 또는 문서 데이터를 NoSQL 데이터베이스에 저장합니다.
- Amazon Redshift는 정형 데이터에 대한 워크로드를 데이터 웨어하우스에 저장합니다.



올바른 서비스와 올바른 구성을 사용하면 가장 효율적이고 효과적인 방식으로 데이터를 저장할 수 있습니다. 이렇게 하면 데이터 검색과 관련된 노력이 최소화됩니다.

데이터 품질 검사

데이터 품질은 필수이지만 데이터 정리 프로세스에서 종종 간과되는 부분입니다. 다음 다이어그램에서는 데이터 품질 검사가 데이터 엔지니어링 자동화 및 액세스 제어 수명 주기에 어떻게 부합하는지 보여줍니다.

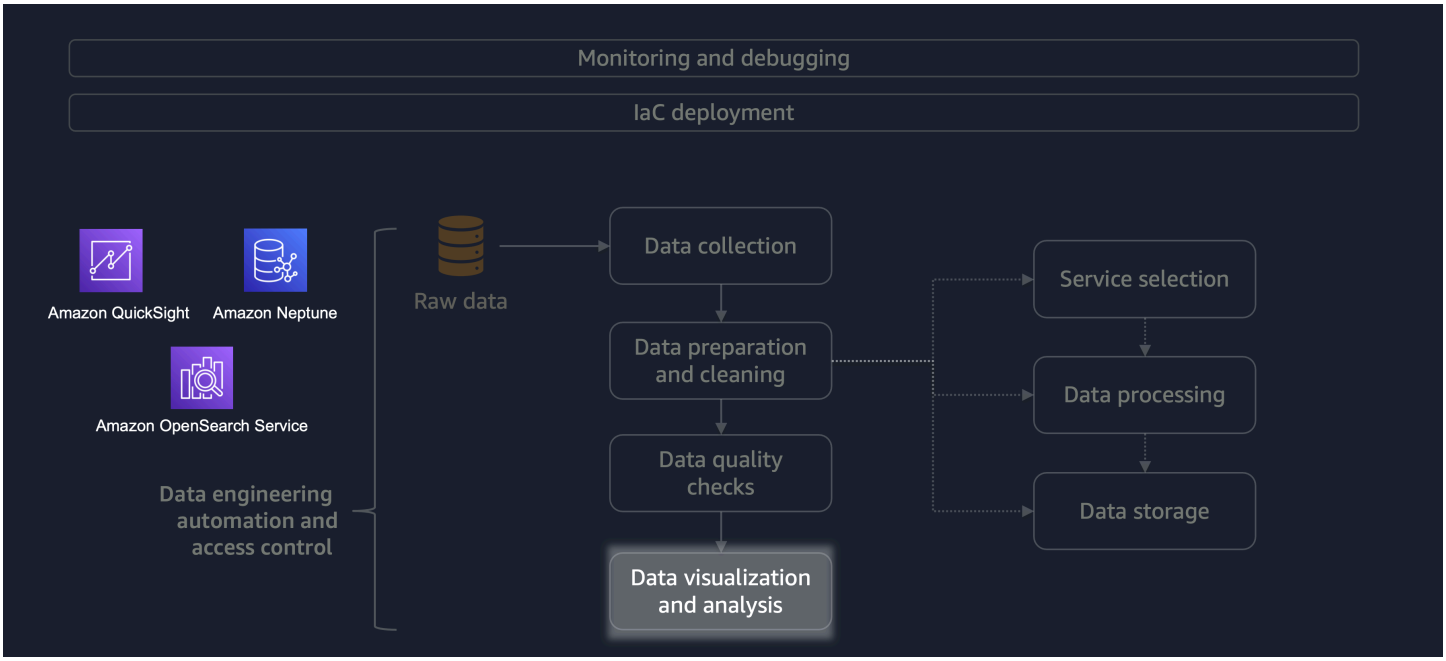


다음 표에서는 사용 사례에 따라 다양한 데이터 품질 솔루션에 대한 개요를 제공합니다.

사용 사례:	솔루션	예제
열 수준 또는 테이블 수준 품질 조건을 추가하는 노코드 솔루션	AWS Glue DataBrew	모든 열 값이 1에서 12 사이인지 또는 테이블이나 열이 비어 있는지 확인합니다.
열 수준 또는 테이블 수준 품질 조건을 추가하기 위해 AWS Glue 작업 또는 노코드 솔루션(미리 보기)에 추가된 사용자 지정 코드	AWS Glue Data Quality	`first_name` 열이 null이 아닌지 또는 `phone_number` 열에 숫자 또는 '+' 연산자 및/또는 평균 또는 합계와 같은 통계 함수만 포함되어 있는지 확인합니다.
사용자 지정 검사	AWS Lambda, AWS Glue 또는 Amazon EMR과 같은 원하는 ETL	A 열의 값이 항상 B 열 및 C 열의 해당 값보다 큰지 또는 `continent` 열 값이 항상 지리적으로 정확하고 `city` 열에서 파생되었는지 확인합니다.
지표 보고서, 제약 조건 검증 및 제약 조건 제안이 포함된 정교한 솔루션	Deequ	`review_id` 열 지표의 완전성에 대한 `CompletenessConstraint` 가 1인지 확인합니다.

데이터 시각화 및 분석

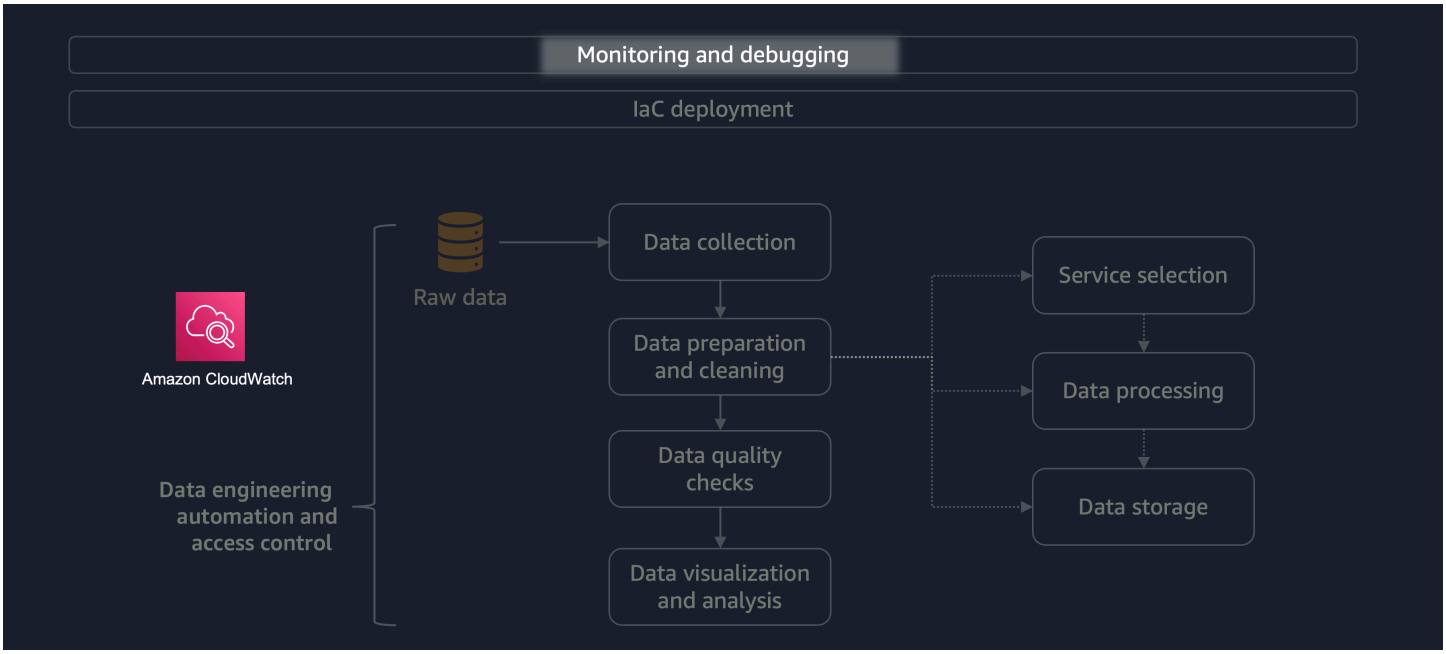
데이터 품질 검사를 완료한 후 다음 다이어그램과 같이 데이터 분석 또는 시각화 단계로 진행할 수 있습니다.



이 단계에서는 Quick Sight를 사용하여 그래프 또는 차트를 생성하거나, Neptune을 사용하여 그래프 데이터베이스 작업 및 시각화를 수행하거나, OpenSearch를 사용하여 오픈 소스 검색 및 분석을 수행할 수 있습니다. 일반적으로 Amazon SageMaker 파이프라인 또는 Amazon S3의 간단한 읽기를 사용하여 데이터 과학 또는 기계 학습(ML) 워크플로에 클린 데이터를 제공할 수도 있습니다. 데이터 시각화 및 분석 단계에서는 데이터 엔지니어링 파이프라인의 순차적 부분을 완료합니다.

모니터링 및 디버깅

데이터 수명 주기의 특정 단계는 순차적이지 않지만 일관되게 존재합니다. 다음 다이어그램과 같이 모니터링 및 디버깅 단계에서도 마찬가지입니다.

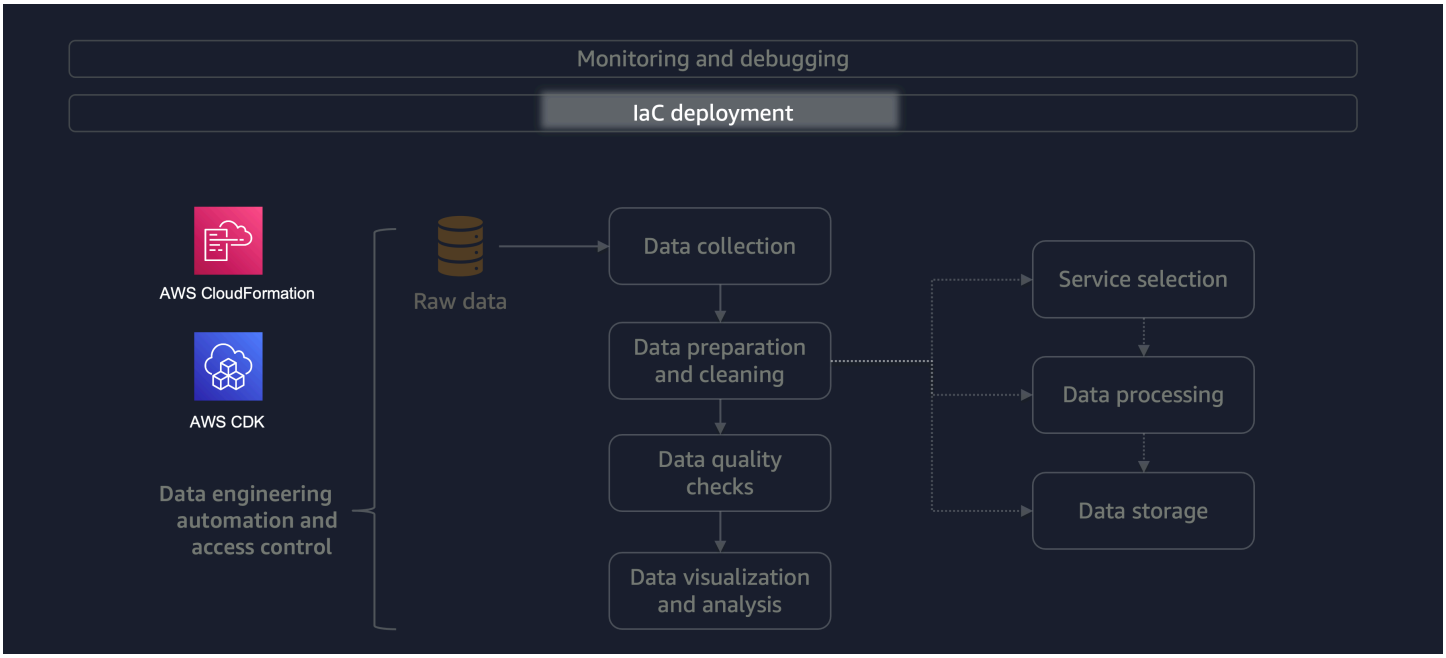


정확성과 성능을 위해 데이터 엔지니어링 프로세스를 지속적으로 모니터링해야 합니다. Amazon CloudWatch는 모든 오류 및 정보 로그를 로그 그룹에 로깅하므로 데이터 엔지니어링을 모니터링하는 데 중요한 역할을 합니다. 모니터링을 사용하여 자동화된 오류 복구를 빌드할 수 있습니다. 예를 들어 데이터 품질 규칙이 충족되지 않는 경우 파이프라인을 중지하거나 성공한 실행과 실패한 실행을 별도로 로깅하여 복구 작업을 활성화할 수 있습니다. 모니터링은 데이터 엔지니어링 프로세스(즉, 전체 ETL 프로세스)와 데이터의 전반적인 신뢰성을 개선합니다.

또한 모니터링 및 디버깅 프로세스에 대한 관련 지표가 포함된 CloudWatch 대시보드를 생성하는 것이 좋습니다. 이를 통해 데이터 엔지니어링 프로세스가 원활하고 예상대로 실행되도록 할 수 있습니다. 보고뿐만 아니라 운영에도 중요합니다. 예를 들어 CloudWatch 대시보드는 사용자에게 로드 상태를 표시하여 프로세스의 신뢰성이나 품질 저하로 인해 삭제된 데이터의 비율 또는 최대 장애가 발생한 소스를 이해하는 데 도움이 될 수 있습니다. CloudWatch 대시보드는 결과를 시각화할 뿐만 아니라 ETL 프로세스의 문제점을 식별하여 프로세스를 개선하는 데도 도움이 됩니다.

IaC 배포

코드형 인프라(IaC) 배포에 대한 메커니즘 없이는 최신 아키텍처가 불완전합니다. 다음 다이어그램에서는 IaC 배포와 관련된 AWS 서비스를 보여줍니다.



배포된 인프라는 항상 IaC 도구를 사용하는 코드로 지원하는 것이 좋습니다. 예를 들어 AWS CloudFormation 또는 AWS Cloud Development Kit(AWS CDK)를 사용할 수 있습니다. AWS CDK는 CloudFormation과 관련된 래퍼입니다.

모범 사례로, 선택한 코드 리포지토리로 코드를 푸시하는 것이 좋습니다. 또한 여러 팀원이 동일한 코드베이스에서 동시에 작업할 수 있도록 버전 관리 및 협업 기능을 제공하는 동시에 다른 개발자의 코드를 기본 브랜치로 통합해도 충돌이 발생하지 않도록 코드 리포지토리에서 소스 제어를 사용하는 것이 모범 사례입니다.

자동화 및 액세스 제어

자동화

파이프라인 자동화는 최신 데이터 중심 아키텍처 설계의 중요한 부분입니다. 프로덕션 시스템을 성공적으로 실행하려면 시작 트리거, 연결 단계 및 실패한 단계와 전달된 단계를 분리하는 메커니즘이 있는 데이터 파이프라인을 사용하는 것이 좋습니다. 나머지 ETL 프로세스를 방해하지 않으면서 실패를 기록하는 것도 중요합니다.

AWS Glue 워크플로를 사용하여 파이프라인을 생성할 수 있습니다. 파이프라인은 모든 AWS Glue 작업, Amazon EventBridge 트리거 및 크롤러를 지원합니다. 워크플로를 처음부터 생성하거나 AWS Glue 블루프린트를 사용하여 생성할 수도 있습니다. 블루프린트는 재사용 가능한 사용 사례를 시작하는 데 도움이 되는 프레임워크를 제공합니다. 예를 들어 Amazon S3에서 DynamoDB 테이블로 데이터를 가져오는 워크플로일 수 있습니다. 파라미터를 사용하여 블루프린트를 재사용할 수도 있습니다.

데이터 파이프라인에 AWS Glue 외부의 더 많은 서비스가 포함된 경우 AWS Step Functions를 오케스트레이터로 사용하는 것이 좋습니다. Step Functions는 보안 인시던트 대응을 위한 수동 승인 단계를 포함하여 자동화된 워크플로를 생성할 수 있습니다. 대규모 병렬 또는 순차 처리에 Step Functions를 사용할 수도 있습니다.

마지막으로 EventBridge를 사용하여 일정, 이벤트 또는 온디맨드에 트리거를 삽입하는 것이 좋습니다. EventBridge를 사용하여 필터가 있는 파이프라인을 생성할 수도 있습니다.

액세스 관리

액세스 제어를 위해 AWS Identity and Access Management(IAM)를 사용하는 것이 좋습니다. IAM을 사용하면 AWS의 서비스 및 리소스에 액세스할 수 있는 사용자 또는 대상을 지정하고 세분화된 권한을 중앙에서 관리할 수 있습니다. 스토리지부터 자동화, 처리 도구 사용에 이르기까지 수명 주기의 모든 단계에서 올바른 액세스 권한이 필요합니다. 데이터 중심 사용 사례로 작업하는 동안 AWS Lake Formation을 사용하여 광범위한 분석과 여러 계정에서 데이터를 사용할 수 있도록 하는 프로세스를 단순화할 수 있습니다.

모범 사례

스토리지 및 기술 모범 사례를 따르는 것이 좋습니다. 이러한 모범 사례는 데이터 중심 아키텍처를 최대한 활용하는 데 도움이 될 수 있습니다.

빅 데이터에 대한 스토리지 모범 사례

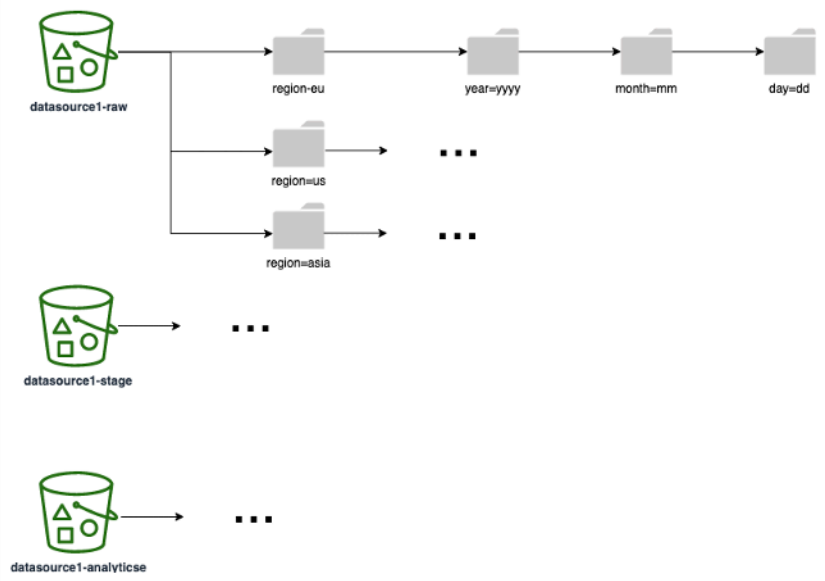
다음 표에서는 Amazon S3에서 빅 데이터 처리 로드를 위해 파일을 저장하는 일반적인 모범 사례를 설명합니다. 마지막 열은 사용자가 설정할 수 있는 수명 주기 정책에 관한 예제입니다. Amazon S3 Intelligent-Tiering이 활성화된 경우(데이터 액세스 패턴이 자동으로 변경될 때 자동 스토리지 비용 절감 제공) 정책을 수동으로 설정하지 않아도 됩니다.

데이터 계층 이름	설명	수명 주기 정책 전략 예제
원시	처리되지 않은 원시 데이터 포함 참고: 외부 데이터 소스의 경우 원시 데이터 계층은 일반적으로 데이터의 1:1 복사본이지만 수집 프로세스 중에 AWS 리전 또는 날짜를 기준으로 키로 AWS 데이터를 분할할 수 있습니다.	1년이 지나면 파일을 S3 Standard-IA 스토리지 클래스로 이동합니다. S3 Standard-IA에서 2년이 지나면 Amazon Simple Storage Service Glacier(Amazon S3 Glacier)에 파일을 아카이브합니다. Amazon Glacier(기존 독립 실행형 볼트 기반 서비스)는 2025년 12월 15일부터 기존 고객에게 영향을 주지 않고 더 이상 신규 고객을 받지 않습니다. Amazon Glacier는 데이터를 볼트에 저장하고 Amazon S3 및 Amazon S3 Glacier 스토리지 클래스와 구별되는 자체 API를 갖춘 독립 실행형 서비스입니다. 기존 데이터는 Amazon Glacier에서 무기한으로 안전하게 보관되며 액세스 가능합니다. 마이그레이션은 필요하

		지 않습니다. 저비용 장기 아카이브 스토리지의 경우는 S3 버킷 기반 API, 전체 가용성, 저렴한 비용 및 서비스 통합을 통해 우수한 고객 경험을 제공하는 Amazon S3 Glacier 스토리지 클래스를 AWS 권장합니다. S3 APIs AWS 리전 AWS 향상된 기능을 원하는 경우 Amazon S3 볼트에서 Amazon AWS S3 Glacier 스토리지 클래스로 데이터를 전송하기 위한 솔루션 지침을 사용하여 Amazon S3 Glacier 스토리지 클래스로 마이그레이션하는 것이 좋습니다.
단계	소비에 최적화된 중간 처리 데이터 포함 예제: CSV에서 Apache Parquet으로 변환된 원시 파일 또는 데이터 변환	정의된 기간 이후에 또는 조직의 요구 사항에 따라 데이터를 삭제할 수 있습니다. 짧은 시간(예: 90일 후) 후에 데이터 레이크에서 일부 데이터 파생물(예: 원래 JSON 형식의 Apache Avro 변환)을 제거할 수 있습니다.
분석	특정 사용 사례에 대해 집계된 데이터를 사용 가능한 형식으로 포함 예: Apache Parquet	데이터를 S3 Standard-IA로 이동한 다음 정의된 기간 이후에 또는 조직의 요구 사항에 따라 데이터를 삭제할 수 있습니다.

다음 다이어그램에서는 모든 데이터 계층에서 사용할 수 있는 분할 전략(하나의 S3 폴더/접두사에 해당)에 대한 예제를 보여줍니다. 다운스트림에서 데이터가 사용되는 방식에 따라 분할 전략을 선택하는 것이 좋습니다. 예를 들어 데이터를 기반으로 보고서를 작성하는 경우(보고서의 가장 일반적인 쿼리가

리전 및 날짜를 기준으로 결과를 필터링하는 경우) 쿼리 성능과 런타임을 개선하기 위해 리전과 날짜를 파티션으로 포함해야 합니다.



기술 모범 사례

기술 모범 사례는 데이터 중심 아키텍처를 설계하는 데 사용하는 특정 AWS 서비스 및 처리 기술에 따라 달라집니다. 그러나 다음 모범 사례를 염두에 두는 것이 좋습니다. 이러한 모범 사례는 일반적인 데이터 처리 사용 사례에 적용됩니다.

영역	모범 사례
SQL	데이터에 속성을 프로젝션하여 쿼리해야 하는 데이터의 양을 줄입니다. 전체 테이블을 구문 분석하는 대신 데이터 프로젝션을 사용하여 테이블의 특정 필수 열만 스캔하고 반환할 수 있습니다. 여러 테이블 간의 조인은 리소스 집약적인 요구로 인해 성능에 상당한 영향을 미칠 수 있으므로 가능하면 대규모 조인은 피합니다.
Apache Spark	AWS Glue (AWS Big Data 블로그)에서 워크로드 파티셔닝을 사용하여 Spark 애플리케이션을 최적화합니다.

	AWS Glue (AWS Big Data 블로그)에서 메모리 관리를 최적화합니다.
데이터베이스 설계	아키텍처 데이터베이스 모범 사례(AWS 아키텍처 센터)를 따릅니다.
데이터 정리	`catalogPartitionPredicate` 에서 서버 측 파티션 정리를 사용합니다.
규모 조정	수평적 스케일링을 이해하고 구현합니다.

FAQ

데이터 파이프라인과 ML 파이프라인의 차이는 무엇인가요?

데이터 파이프라인은 일반적으로 데이터를 수집, 정리 및 처리하여 기계 학습(ML) 또는 기타 분석 및 시각화 프로세스에 대해 호환되도록 구성되거나 최적화된 데이터 엔지니어링 파이프라인입니다. ML 파이프라인은 일반적으로 ML 모델 생성을 자동화합니다.

수평적 스케일링과 수직적 스케일링의 차이는 무엇인가요?

수평적 스케일링은 처리 능력을 늘리고 클러스터 사용을 지원하기 위해 하드웨어를 추가하는 방식입니다(예: Amazon EMR 또는 AWS Glue 사용). 수직적 스케일링은 기존 하드웨어의 처리 능력을 강화하는 방식입니다(예: EC2 인스턴스의 RAM 용량 증가).

다음 단계

아래와 같은 다음 단계를 수행하는 것이 좋습니다.

1. 현재 데이터 엔지니어링 프로세스를 평가하고 추가 지침 및 지원을 데이터 전문가에게 문의합니다.
2. 데이터 수명 주기의 모든 단계가 효율성을 위해 최적화되도록 데이터 엔지니어링 프로세스를 조정합니다.
3. 이 가이드의 데이터 엔지니어링 원칙과 모범 사례를 사용하여 데이터 중심 아키텍처를 설계하고 사용 사례에 가장 적합한 AWS 서비스를 선택합니다.

리소스

- AWS Data and Analytics Competency Partner
- AWS Glue
- AWS Step Functions
- Introduction to Apache Spark

문서 기록

아래 표에 이 가이드의 주요 변경 사항이 설명되어 있습니다. 향후 업데이트에 대한 알림을 받으려면 RSS 피드를 구독하십시오.

변경 사항	설명	날짜
최초 게시	—	2023년 5월 5일

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