

차세대 데이터 레이크하우스 아키텍처 기반 오픈 소스 활용 방안

: Apache Iceberg, Ahana Presto

—

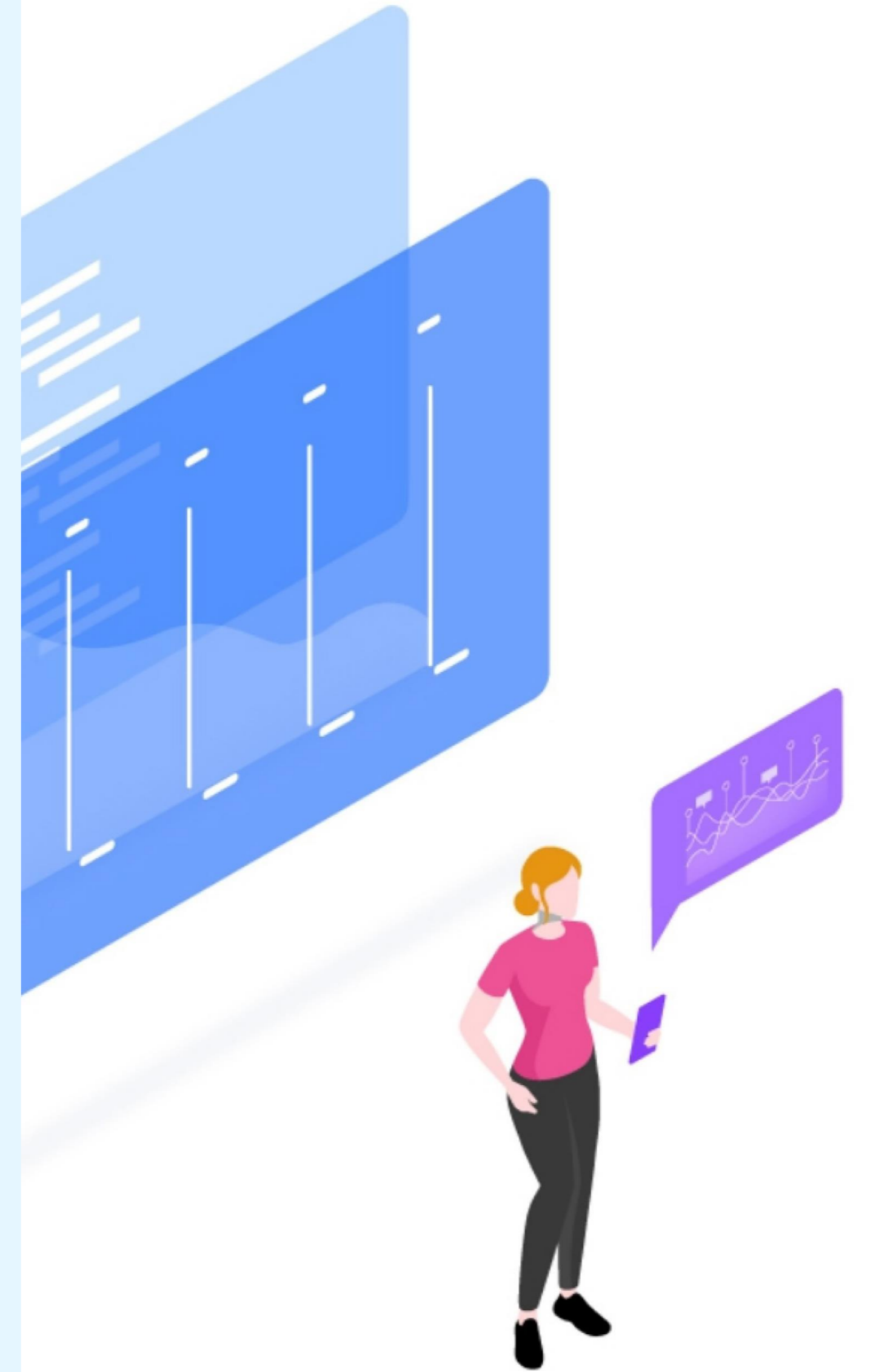
Gyu-pyo Hong
Tech Sales | Data and AI
Gyupyo.hong@ibm.com

IBM Korea

Agenda

- Market trends
- Market growth & dynamics
- Lakehouse overview
- Lakehouse Architecture and Open Source
- Feature highlights
- Key components

Market trends



경쟁력을 갖추려면 **데이터를 차별화 요소**로 사용하십시오!

데이터를 활용하여 고객과 직원을 위한 차별화된 경험 창출



Sales

데이터 중심의 B2B 판매 성장 엔진을 사용하는 회사는 시장보다 높은 성장을 하고 있고, EBITDA(이자, 세금, 감가상각비 차감 전 이익)가 **15~25% 증가합니다.**¹



Marketing

빠르게 변화하는 고객 행동으로 **구매 고려도가 17% 증가했습니다.**²



Operations

2025년까지 거의 모든 직원이 긴 로드맵을 개발하여 문제를 해결하는 데 전념하지 않고 자연스럽게 정기적으로 데이터를 활용하여 업무를 지원합니다.³



Finance

불안정한 조건에서 신뢰할 수 있는 예측을 설정하여 **의사 결정 시간을 90%+ 개선합니다.**⁴



HR

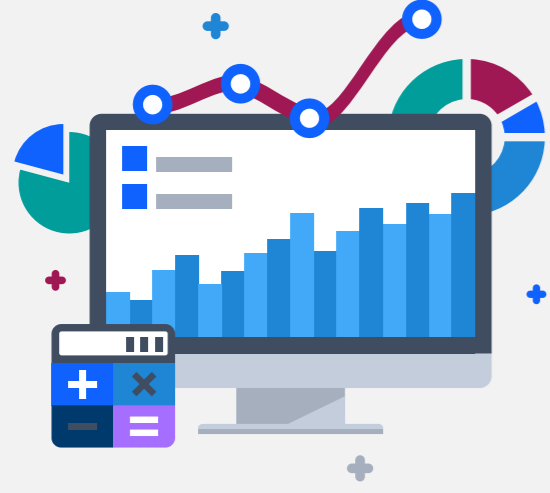
한 달에 600 + 지원자를 참여시켜 **최고의 인재를 유치하고 유지합니다.**⁵



IT

애플리케이션과 인프라를 혁신하고 현대화하여 **전략적 프로젝트에 25% 더 많은 시간을 할애할 수 있습니다.**⁶

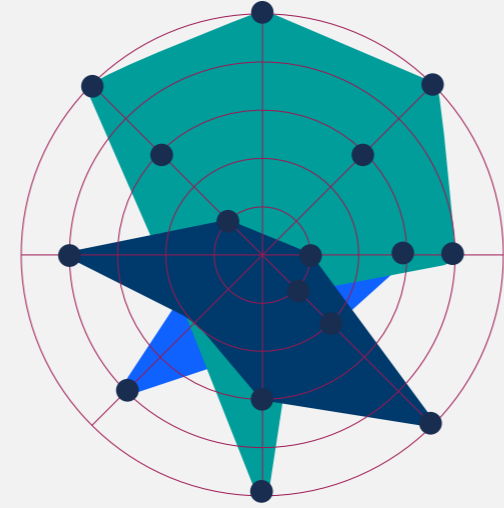
오늘날 기업은 여러 데이터 문제에 직면해 있습니다



더 많은 데이터

폭발적인 데이터 증가

저장된 데이터의 총
볼륨은 향후 5년 동안
250% 이상 증가할
것으로 예상됩니다



다양한 위치

여러 위치, 클라우드,
애플리케이션에 위치한
데이터 사일로 문제

기업의 82%가 데이터
사일로로 인해 접근에
방해를 받고 있습니다.



복잡한 형태

문서, 이미지, 비디오

데이터 정리, 통합 및
준비에 **소요되는 시간의
80%**를 차지합니다.

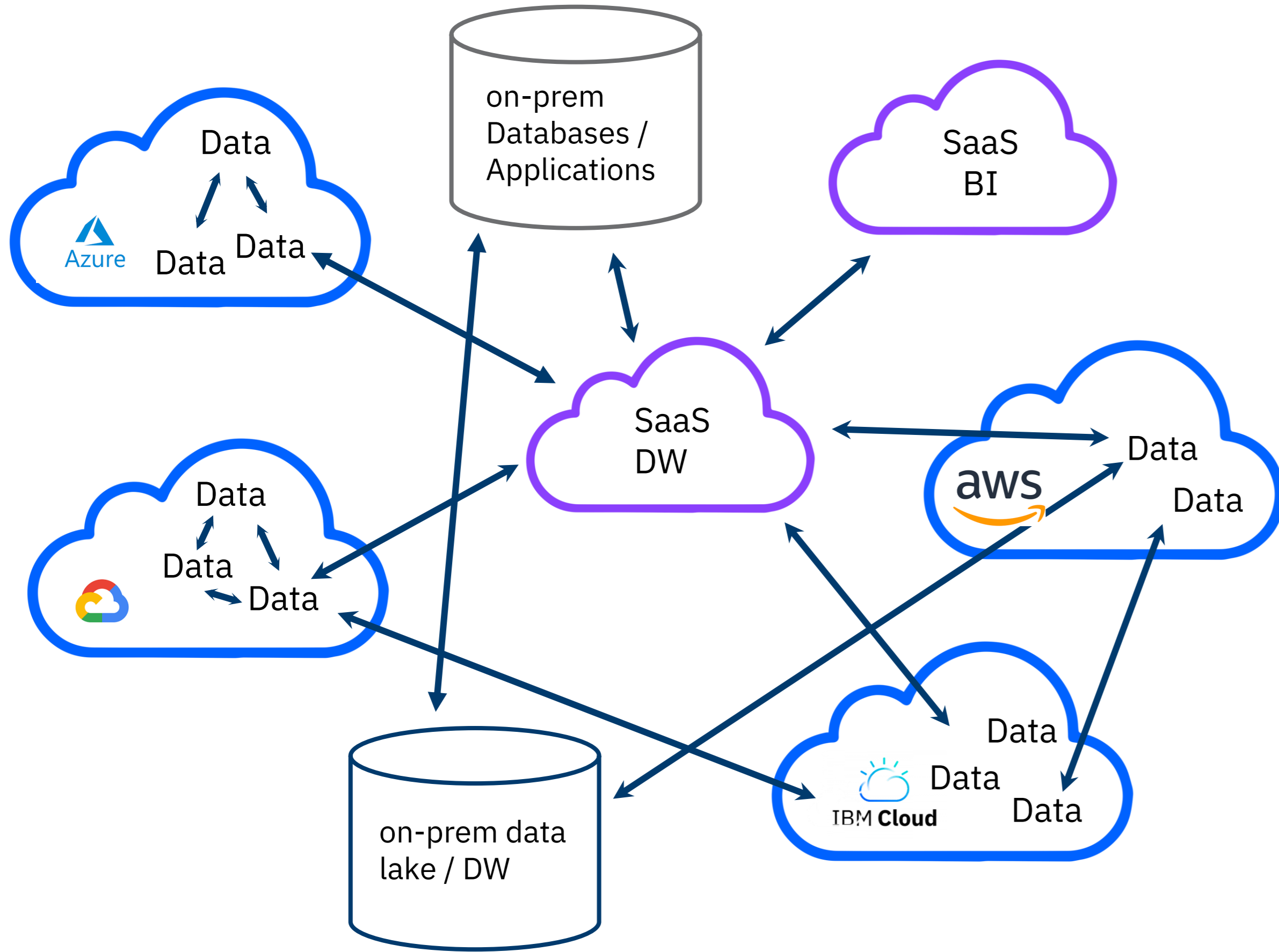


데이터 품질

오래되고 일관성이 없음

기업의 82%는 데이터
품질이 데이터 통합
프로젝트의 장벽이라고
말합니다.

데이터 관리 시장 동인



Hybrid-cloud data ecosystem

복원력(resiliency)을 개선하고 직간접 데이터 관리 비용을 줄이기 위한 인프라 현대화

- 하이브리드 클라우드 배포에서 데이터베이스, 앱 및 ETL에 대한 호환성
- 다운타임 없는 데이터베이스 마이그레이션 툴링 및 자동화
- 최저 TCO(No DBA)를 위한 완전 관리형 서비스 및 소비 기반 가격 책정

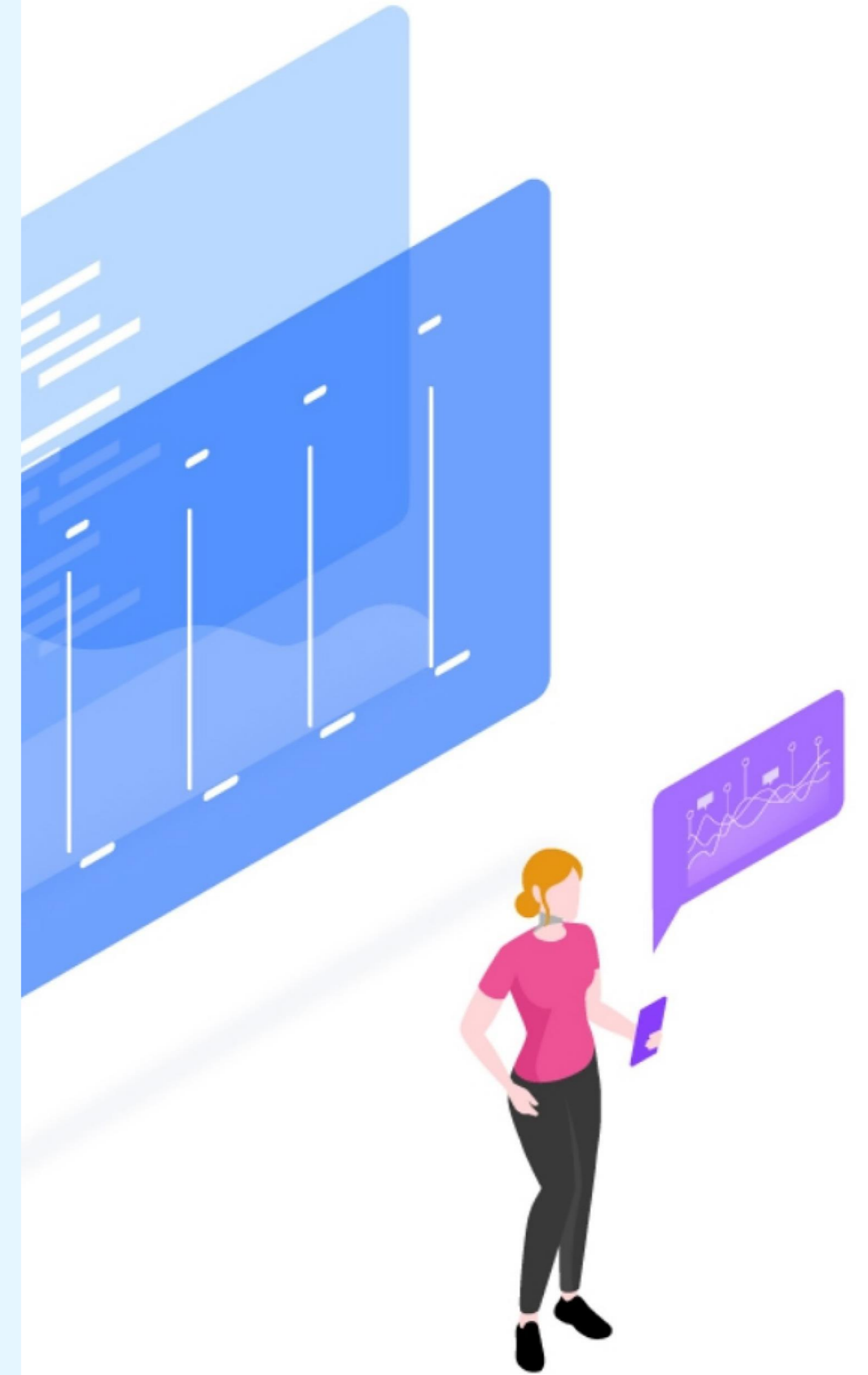
새로운 인사이트를 도출하여 비즈니스 성과를 개선하는 Digital Transformation

- 데이터의 저장 위치에 상관없이 손쉬운 정형 및 비정형 데이터 결합
- 다양한 페르소나 및 특정 워크로드에 대해 선택된 분석 도구 활용
- Zero-copy(ETL 없음)로 신뢰할 수 있는 단일 소스 및 가치 창출 시간 단축

민첩성을 개선하고 손해 배상 및 이미지에 대한 손상을 방지하기 위한 규정 준수 및 보안

- 데이터 검색/카탈로그, 품질, 계보, 시맨틱 및 규정 준수/프라이버시를 위한 중앙 집중식 또는 통합 메타데이터 관리
- 모든 리포지토리 및 쿼리 엔진에서 관리되고 안전한 데이터 액세스 제어 지점을 통해 엔터프라이즈 전체에서 데이터 액세스를 손쉽게 함

Market growth & dynamics

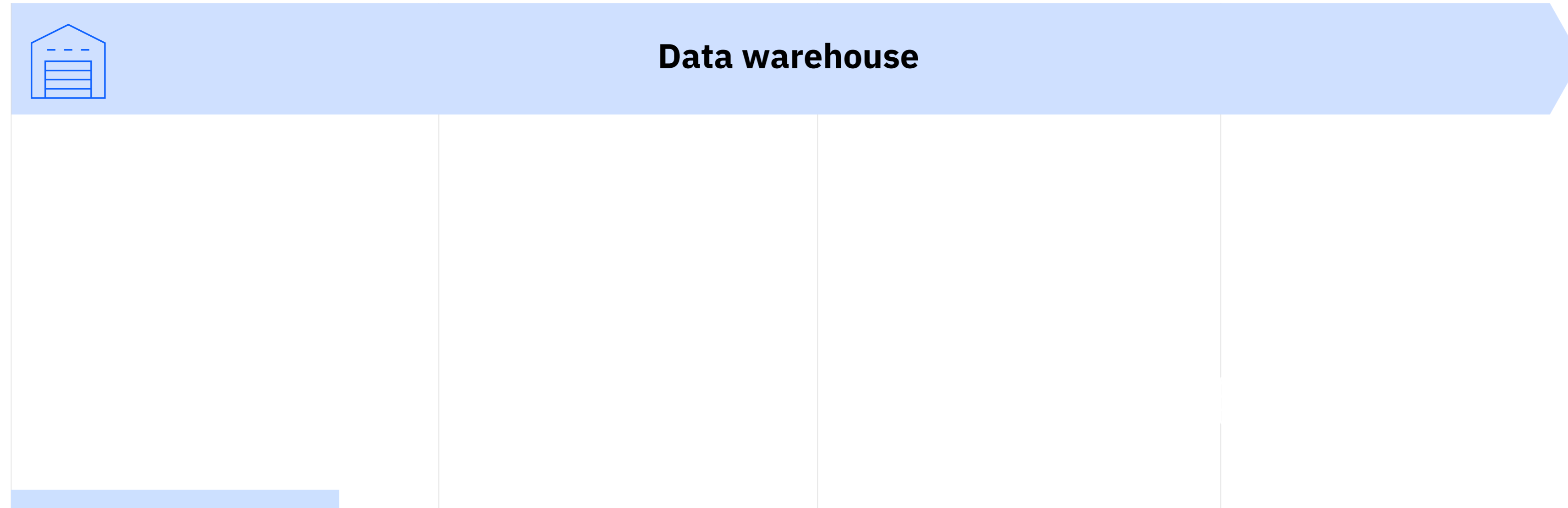


데이터 웨어하우스는 대부분의 조직에서 분석작업 중심으로 유지되고 있음

Late 90s

Early 2000s

Present



높은 초기 비용
구조화된 데이터만 가능
ETL 필요
공급업체 종속
제한된 확장성

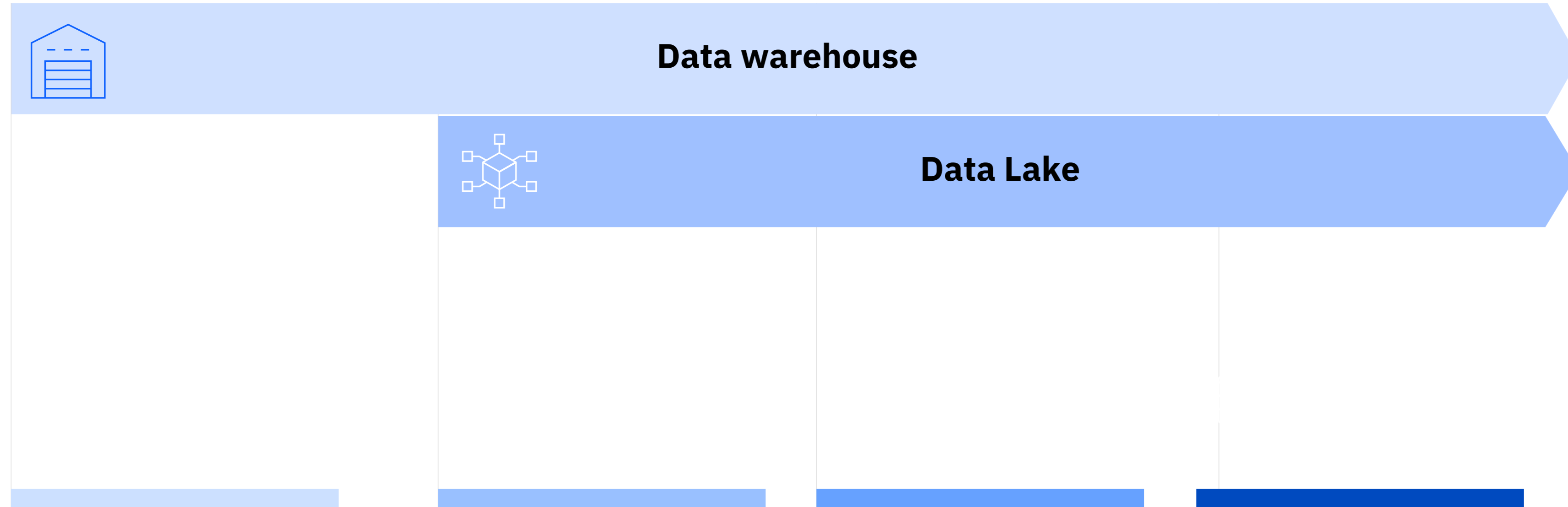
- 데이터 웨어하우스는 데이터를 분석하는 지배적인 방법으로 부상
- 정규화되고 신뢰할 수 있는 데이터를 통해 쉽게 분석할 수 있었지만 비용이 많이 드는 선택
- 데이터 웨어하우스 기술은 Appliance 제품 폼 팩터에서 인메모리 기술에 이르기까지 지속적으로 개선

데이터 레이크의 출현

Late 90s

Early 2000s

Present



높은 초기 비용
구조화된 데이터만 가능
ETL 필요
공급업체 종속
제한된 확장성

높은 복잡성
열악한 데이터 품질
제한된 성능
유지 관리 비용 과다

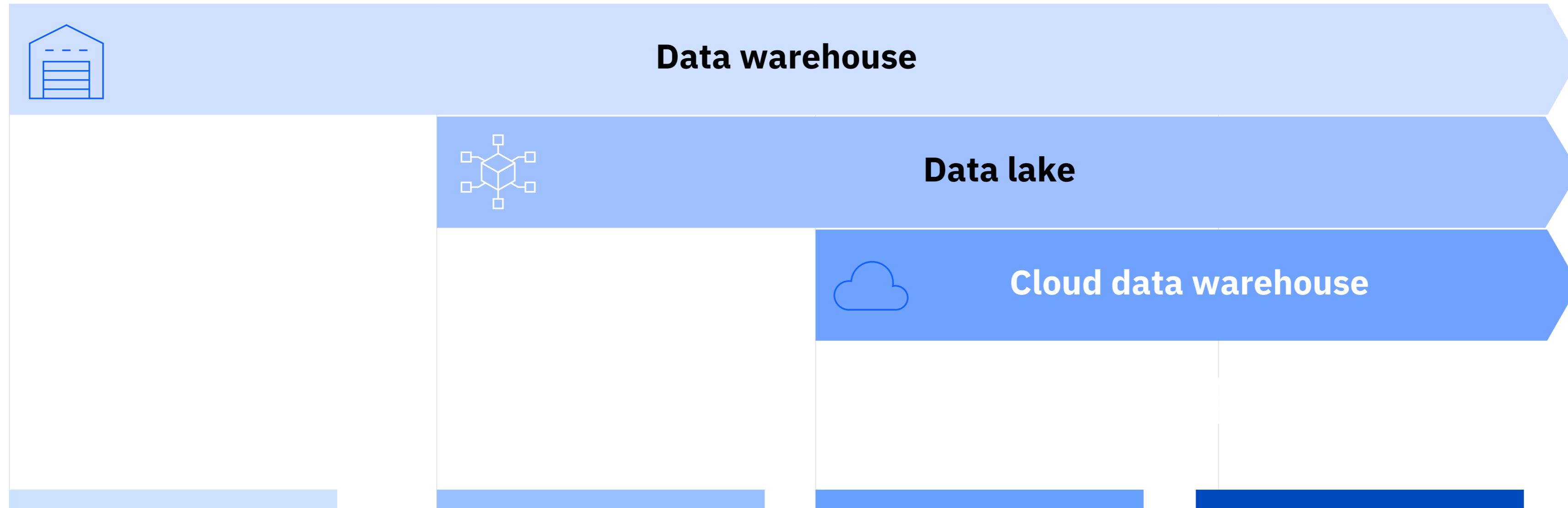
- 데이터의 양, 속도 및 다양성이 증가함에 따라 데이터 레이크는 데이터 웨어하우스를 대체할 새로운 기술로 부상
- RAW 및 비정형 형식으로 저장된 데이터 = 대량의 데이터에 대한 비용 절감
- 뛰어난 유연성과 확장성
- 사용하기 어렵고 유지 관리가 복잡하며 데이터 과학자가 필요
- 결국 대부분의 데이터 레이크는 실패했고 2-tier 아키텍처가 필요

클라우드 데이터 웨어하우스는 데이터 웨어하우스의 문제를 해결하기 위해 등장

Late 90s

Early 2000s

Present



높은 초기 비용
구조화된 데이터만 가능
ETL 필요
공급업체 종속
제한된 확장성

높은 복잡성
열악한 데이터 품질
제한된 성능
유지 관리 비용 과다

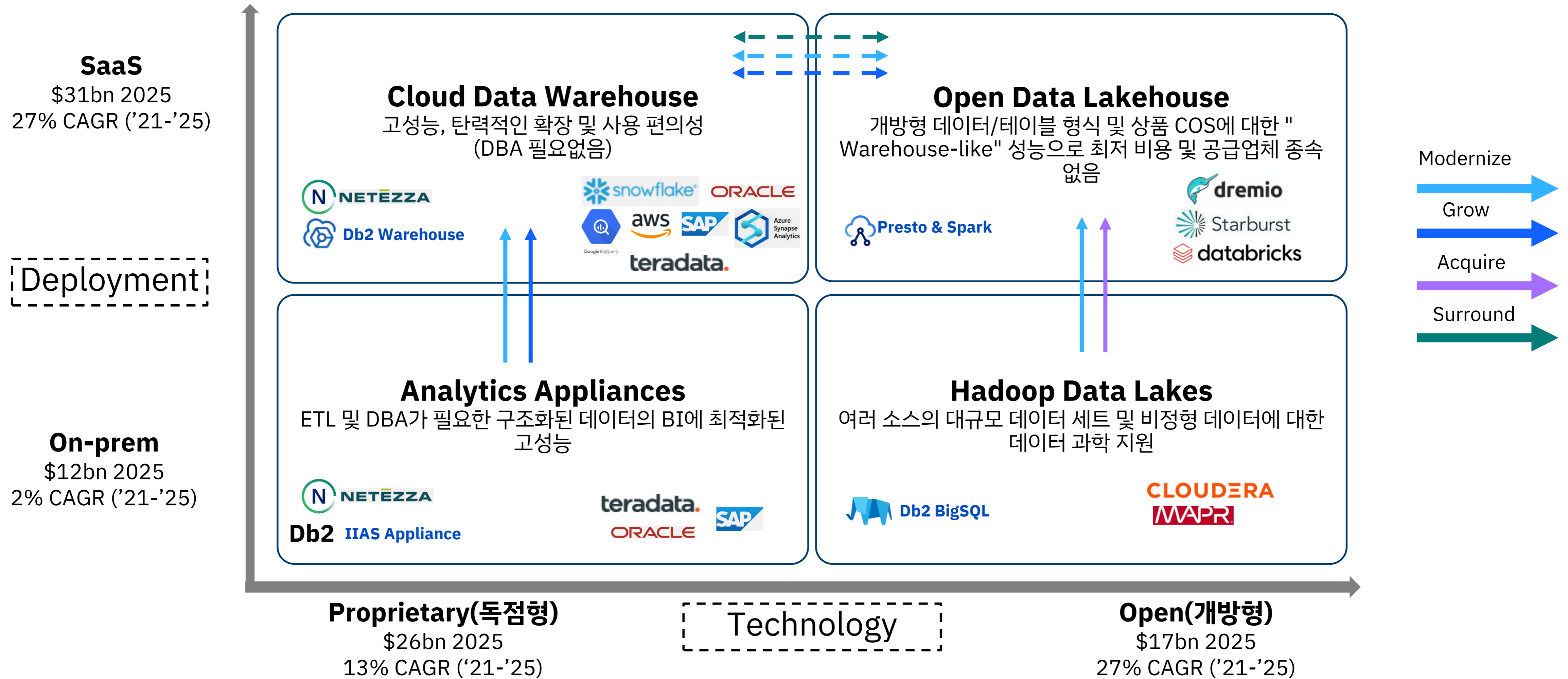
데이터 마이그레이션
공급업체 종속
높은 비용
제한된 AI/ML 사용 사례

- 특히 클라우드 데이터 웨어하우스는 컴퓨팅과 스토리지를 분리 도입
- 기존 웨어하우스의 확장성 문제 해결 - 데이터 재배포 없음
- 문제를 해결하기 위해 동일한 데이터에 더 많은 컴퓨팅 리소스를 추가하는 기능
- 그러나 관리하기가 더 쉽지만 on-prem 웨어하우스보다 고비용

시장의 움직임

주요 혼란으로 인해 분석 저장소 시장이 on-prem에서 SaaS로, 독점에서 개방형 기술로 성장하고 있습니다

분석 저장소 시장 구조



데이터 관리 기술의 근본적인 변화가 시장을 혼란에 빠뜨림

On-prem / cloud

Multi / Hybrid cloud

종속형, 독점 기술

개방형, 상호 운용 가능

거버넌스 (after-the-fact)

Active 거버넌스

Silo 형태의 데이터 분석 및 BI

AI의 대규모 배포

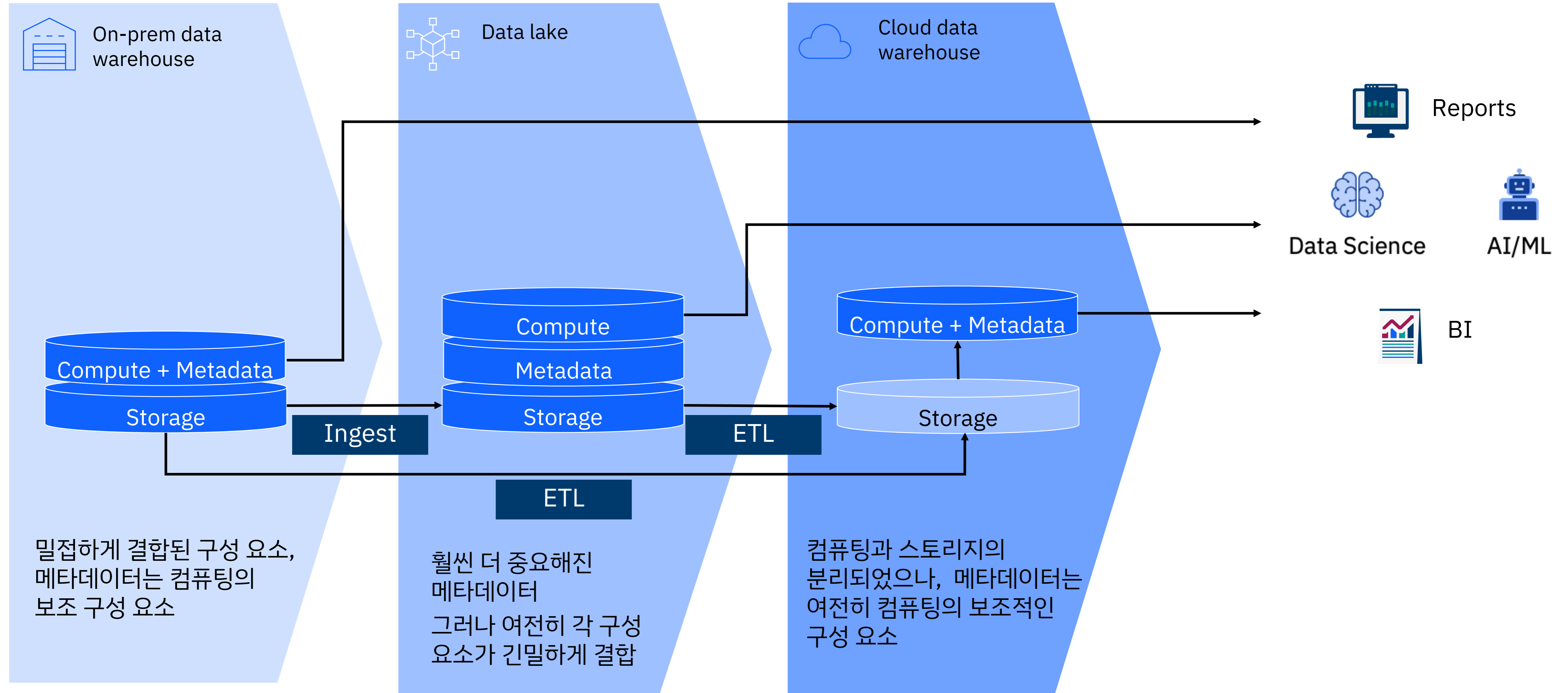
일반적인 데이터 관리 아키텍처 – multi tier

Late 90s

Early 2000s

Present

End User Applications



레이크하우스 아키텍처

Next



Data lakehouse



SQL Engine

Metadata Layer

Ingest

Object Storage 1

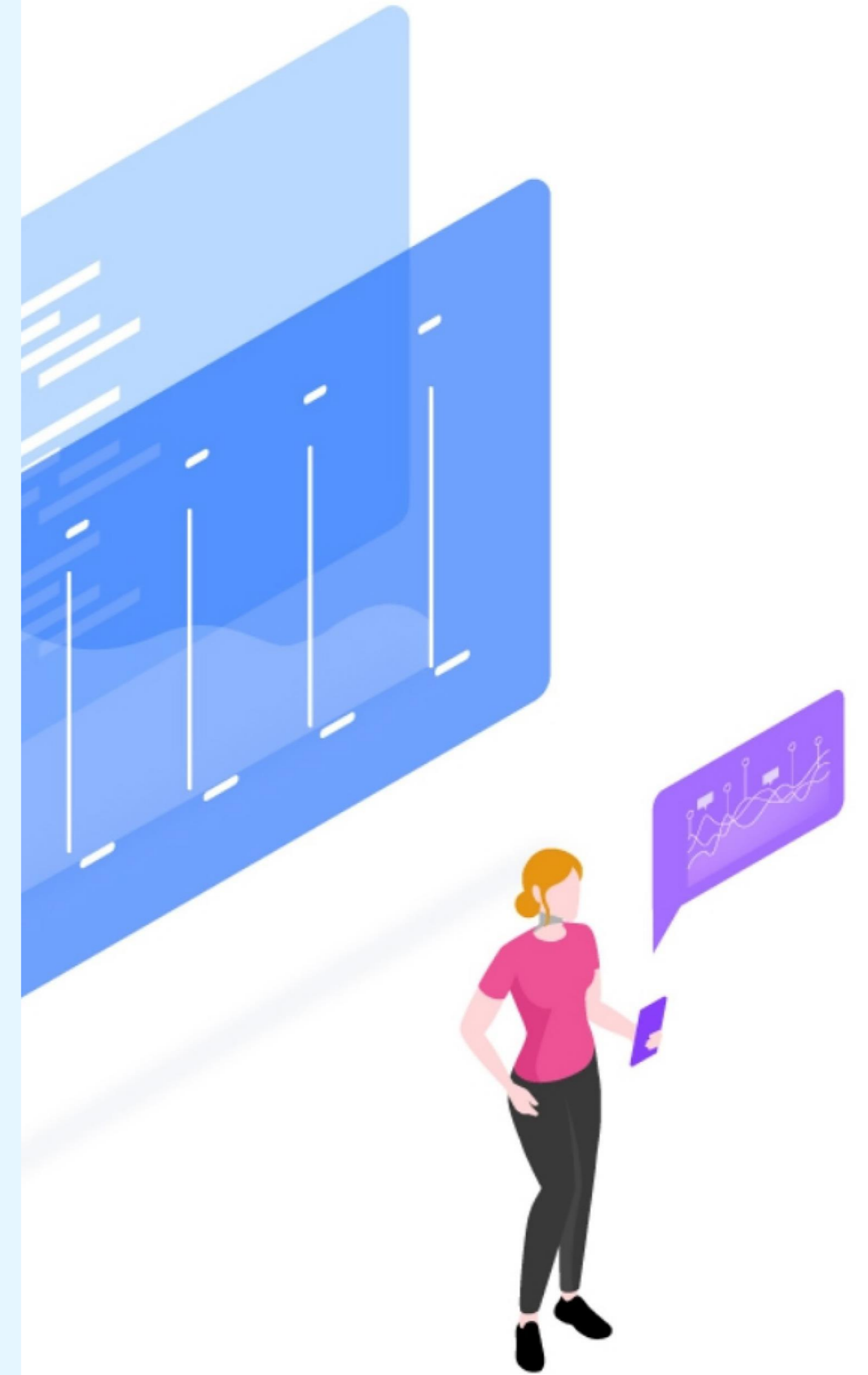
Object Storage 2

Transform

- ☑ 컴퓨팅, 메타데이터 및 스토리지의 완전한 분리
- ☑ 차세대 SQL 엔진
- ☑ 데이터 레이크의 데이터에 대한 Warehouse-like 액세스
- ☑ 모든 워크로드를 한 곳에서 처리
- ☑ 단순화된 아키텍처
- ☑ 복잡한 데이터 파이프라인 감소
- ☑ 모든 데이터를 한 곳에

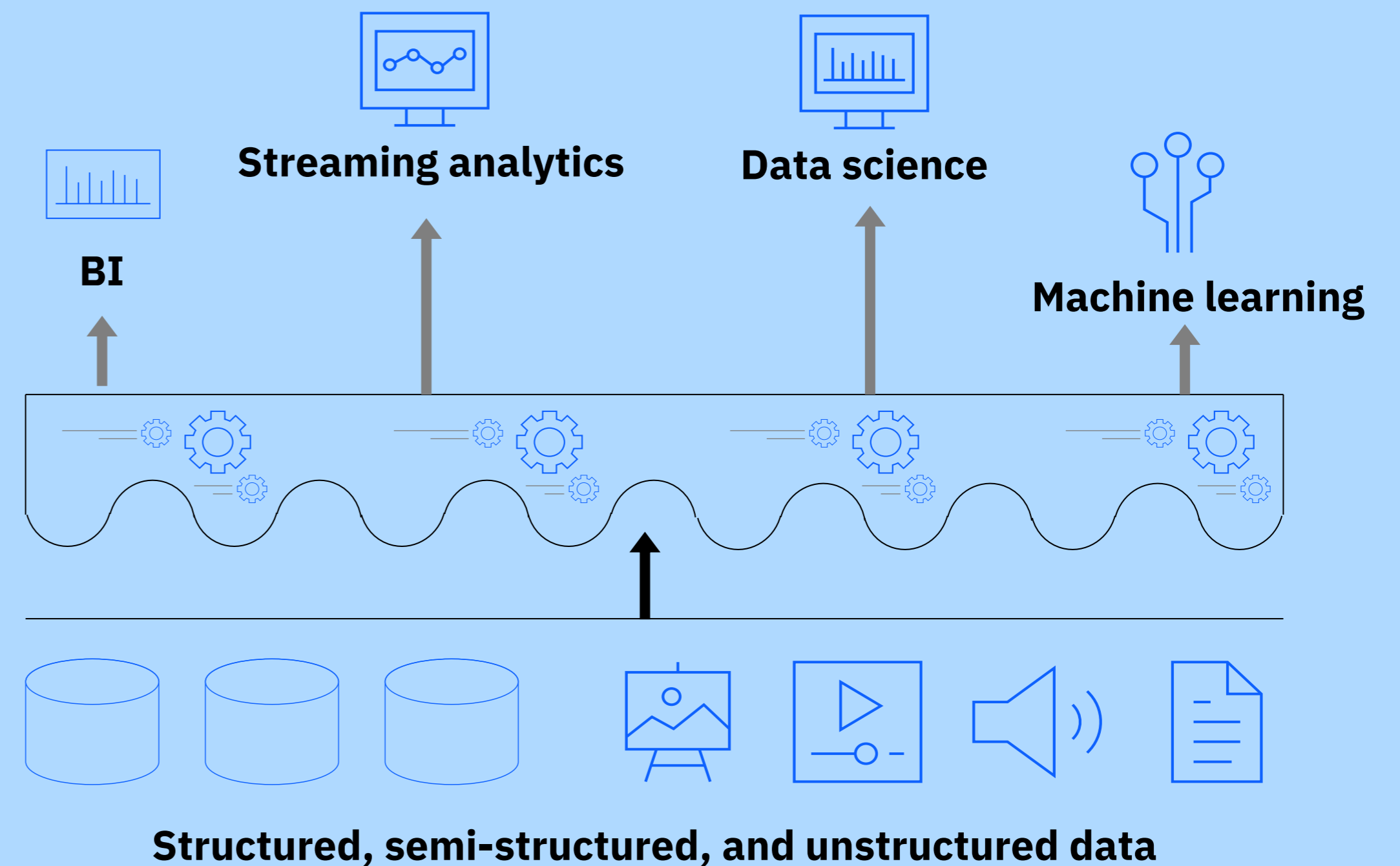
But also has its challenges ...

Lakehouse overview



레이크하우스는 데이터 레이크와 웨어하우스를 최적의 **단일 통합 플랫폼**으로 결합하여, 매우 복잡한 데이터 변환과 **다양한 사용 사례**를 지원합니다

Lakehouse

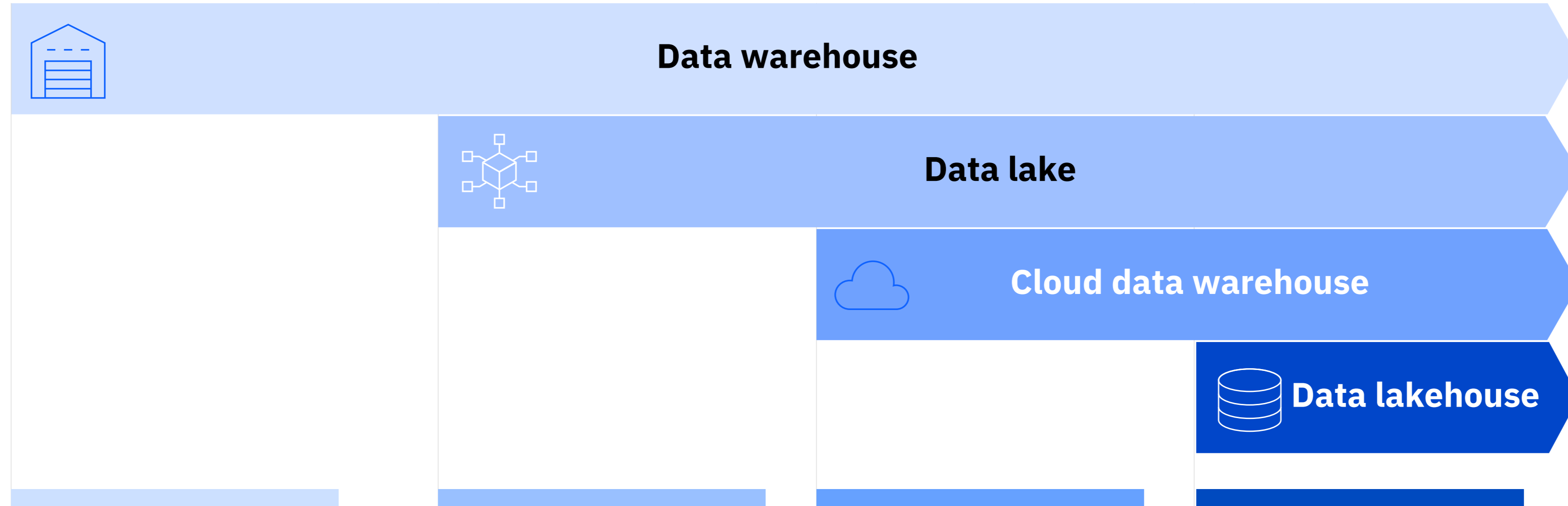


데이터 레이크하우스는 분석의 새로운 시대를 열기 위한 신기술입니다

Late 90s

Early 2000s

Present



높은 초기 비용
구조화된 데이터만 가능
ETL 필요
공급업체 종속
제한된 확장성

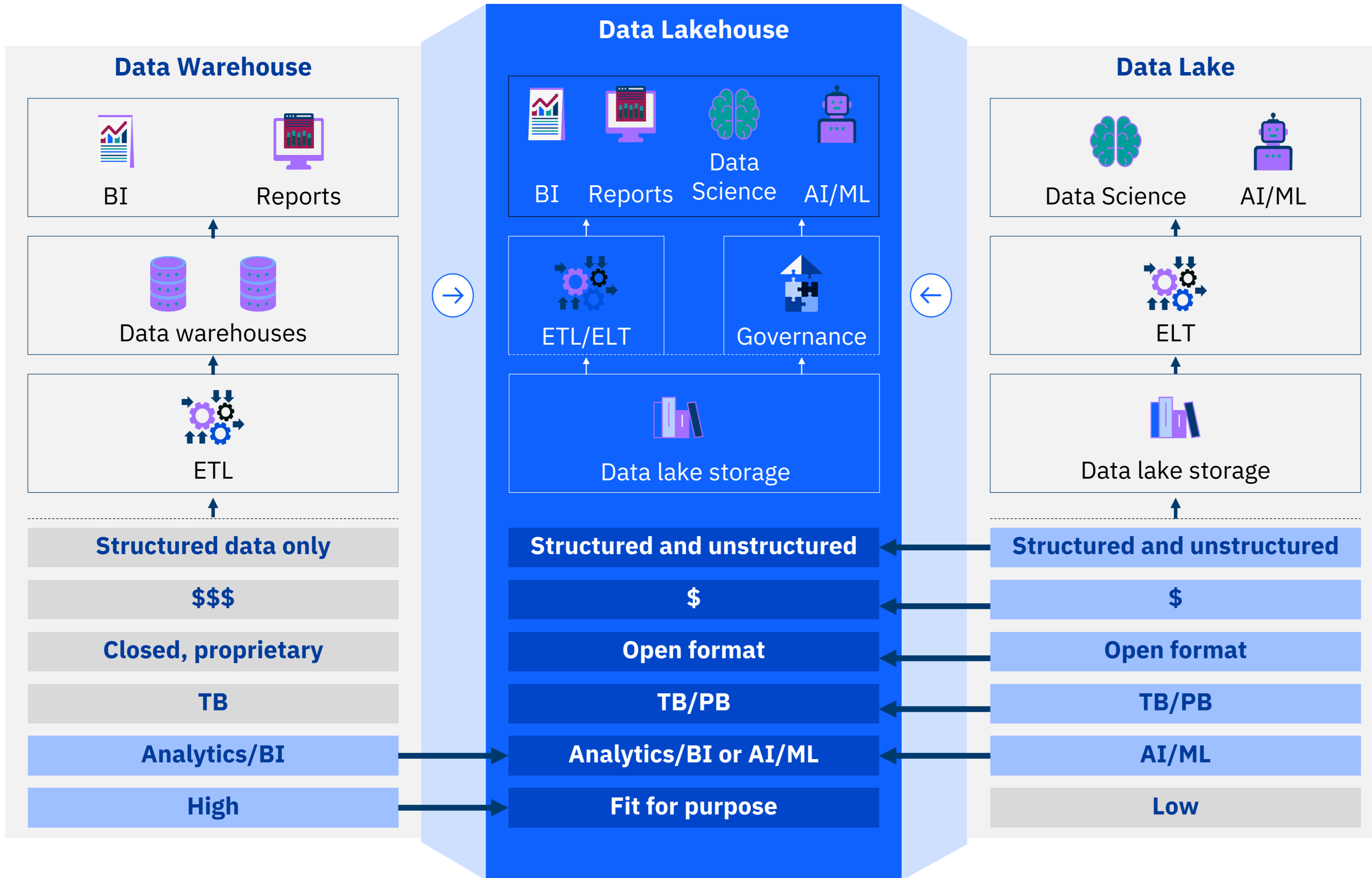
높은 복잡성
열악한 데이터 품질
제한된 성능
유지 관리 비용 과다

데이터 마이그레이션
공급업체 종속
높은 비용
제한된 AI/ML 사용 사례

기술 curve의 초기 단계
BI와 AI를 한 곳에서
DW보다 저렴한 비용
개방성과 유연성

- 오늘날 대부분의 기업은 데이터 레이크와 여러 웨어하우스를 가진 2-tier 아키텍처를 필요로 합니다
- 데이터는 레이크에서 웨어하우스로 이동 및 복제되며, 웨어하우스는 여전히 주요 데이터에 대한 액세스 계층입니다
- 데이터 레이크하우스는 웨어하우스와 데이터 레이크의 장점을 결합합니다.
- 데이터 웨어하우스 엔진(성능) + 데이터 레이크 스토리지(비용)

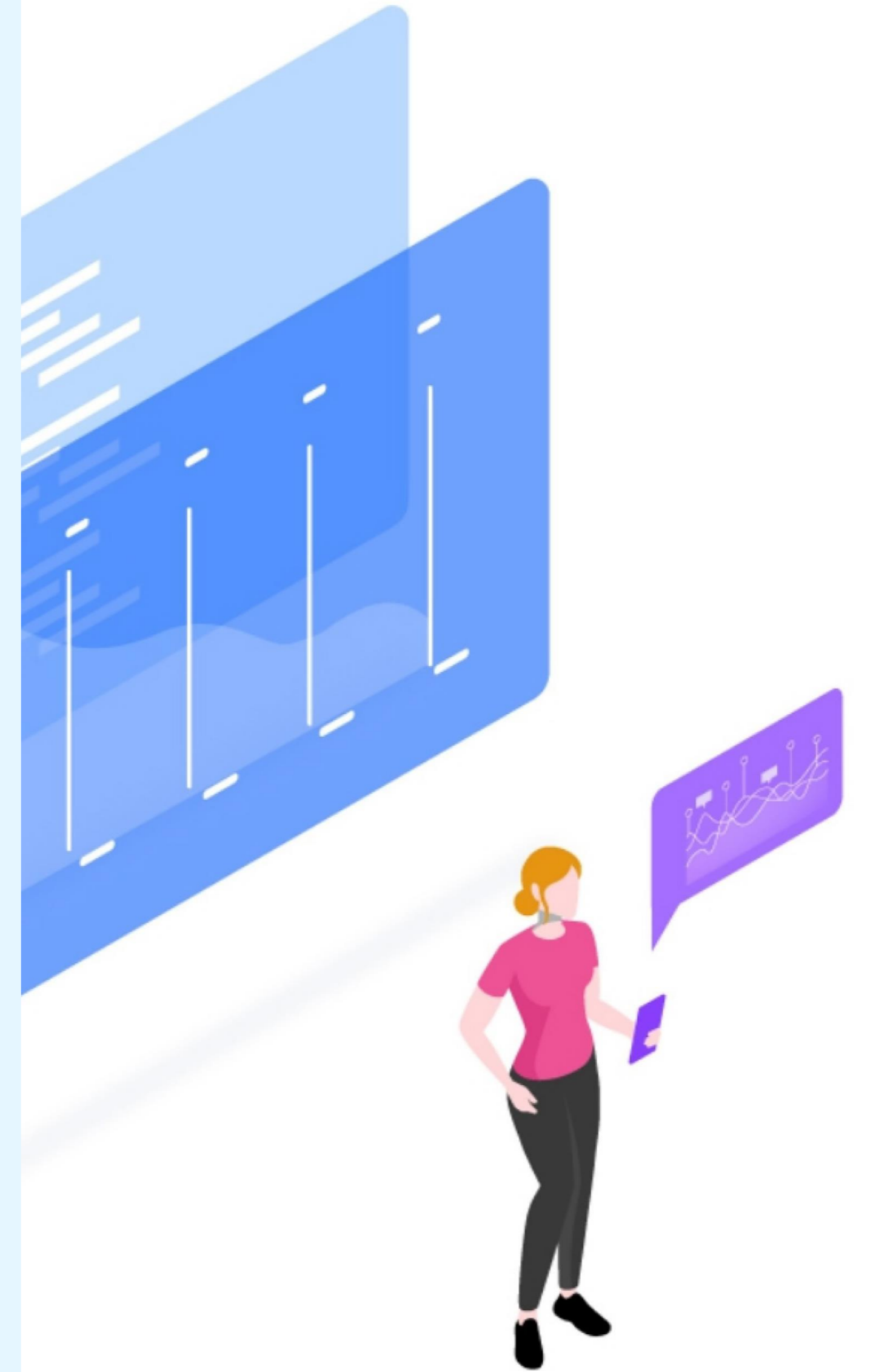
레이크하우스는 데이터 웨어하우스와 데이터 레이크의 장점을 결합한 새로운 차원의 데이터 저장소입니다



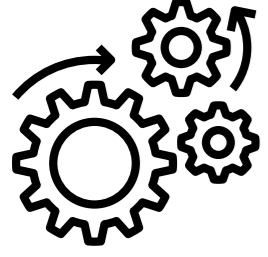
1세대 레이크하우스에는 여전히 비용 및 복잡성 문제의 해결을 제한하는 주요 제약 조건이 존재

- ① 제한된 워크로드를 지원하도록 설정된 단일 쿼리 엔진(일반적으로 BI 또는 ML만 해당)
- ② 일반적으로 멀티/하이브리드 클라우드 배포를 지원하지 않고 단일 클라우드를 통해서만 배포
- ③ 전체 에코시스템에 배포할 수 있는 최소한의 거버넌스 및 메타데이터 기능만 제공

Lakehouse Architecture and Open Source



현대적인 레이크하우스 솔루션을 위한 구성 요소



엔진은 레이크하우스에 대한 쿼리, 분석 및 변환을 가능하게 합니다. **올바른 워크로드 및 사용 사례에 적합한 엔진을 사용할 수 있어야 합니다.**

Multi-Engines



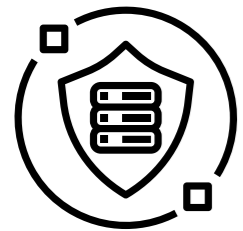
리포지토리는 모든 엔진과 사용자가 구조화된 방식으로 **데이터를 일관되게 찾고 쿼리할 수** 있도록 스키마 및 테이블 메타데이터를 유지 관리합니다.

Metadata repository



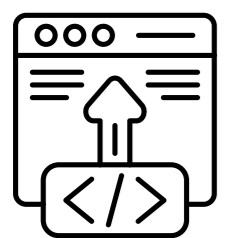
데이터 저장소는 데이터가 물리적으로 저장되는 위치입니다. 고객은 S3 또는 HDFS와 같은 자체 스토리지를 사용하고 필요한 곳에 배치할 수 있기를 기대합니다. **컴퓨팅 엔진이 스토리지와 분리되어 있는 경우 레이크하우스에서 여러 엔진의 동시 액세스를 사용하도록 설정해야 합니다**

Data Storage



레이크하우스는 액세스, 개인 정보 보호, 세이프 하버 또는 주권 규정에 대한 **데이터 정책을 시행할 수 있어야 합니다**

Data Governance

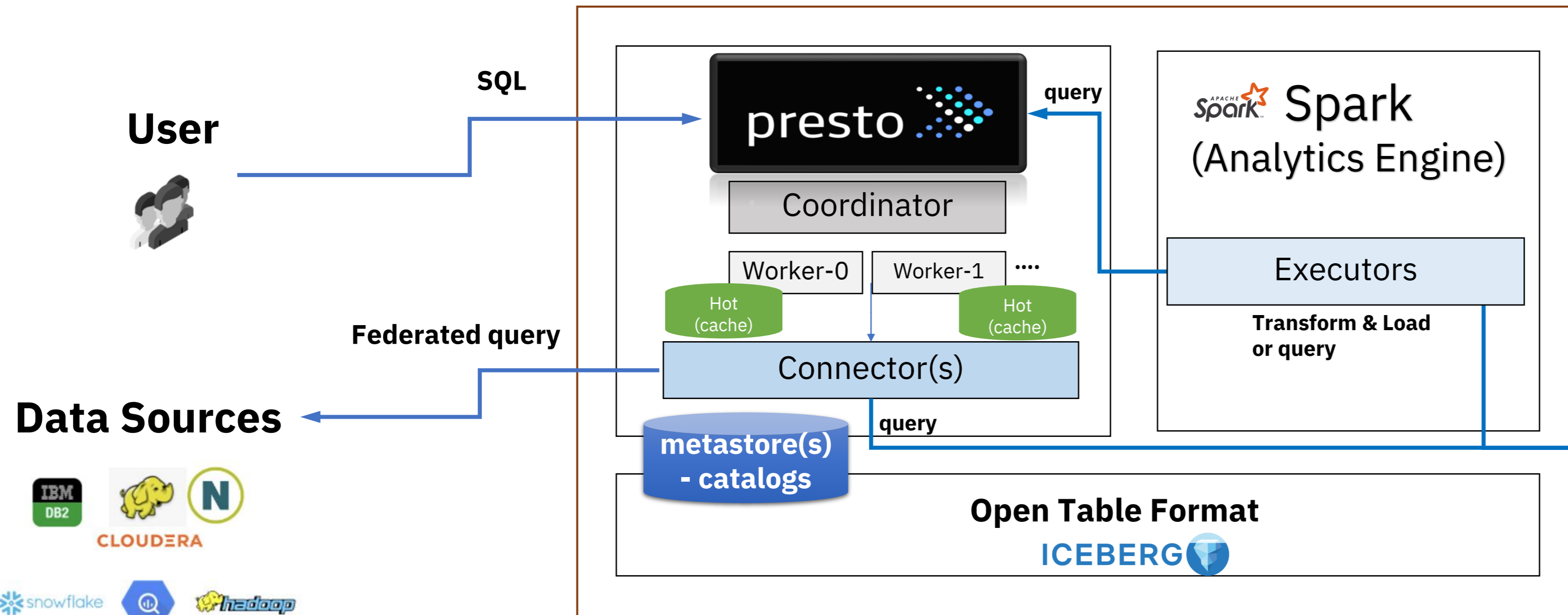


레이크하우스는 어디에나 배치하거나 버스트할 수 있어야 하며 하이브리드 방식으로 클라우드에 걸쳐 있어야 합니다. 애플리케이션과 최종 사용자는 **어디서든** 레이크 하우스 엔진에 액세스해야 합니다

Hybrid Cloud & on-premise Service

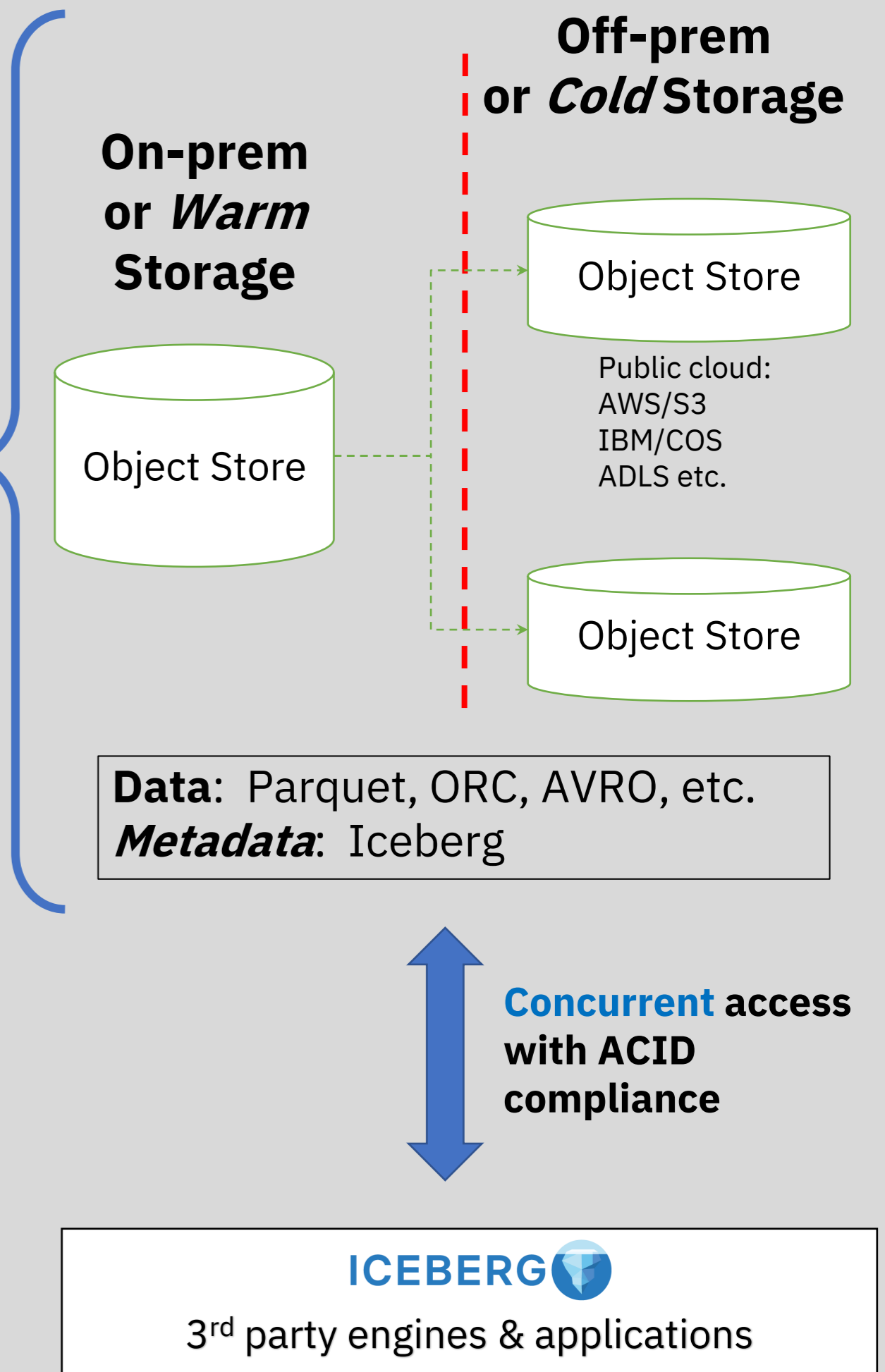
현대적인 레이크하우스 아키텍처

Compute



- 컴퓨팅 & 스토리지 분리, 독립적으로 확장 가능
 - ✓ Multi 엔진 지원 (초기에는 Presto & Spark)
 - ✓ multiple 스토리지 buckets
- 데이터 복사본을 방지하기 위해 데이터 원본에 대한 **현재 위치 액세스**
 - ✓ Federated Queries
- 대상 오브젝트 저장소에 대한 간편한 **데이터 복사**(또는 웜 > 콜드에서 아카이브)
- 개방형 Format 및 표준: 서로 다른 기술들을 연계

Storage



Data: Parquet, ORC, AVRO, etc.
Metadata: Iceberg

Concurrent access with ACID compliance

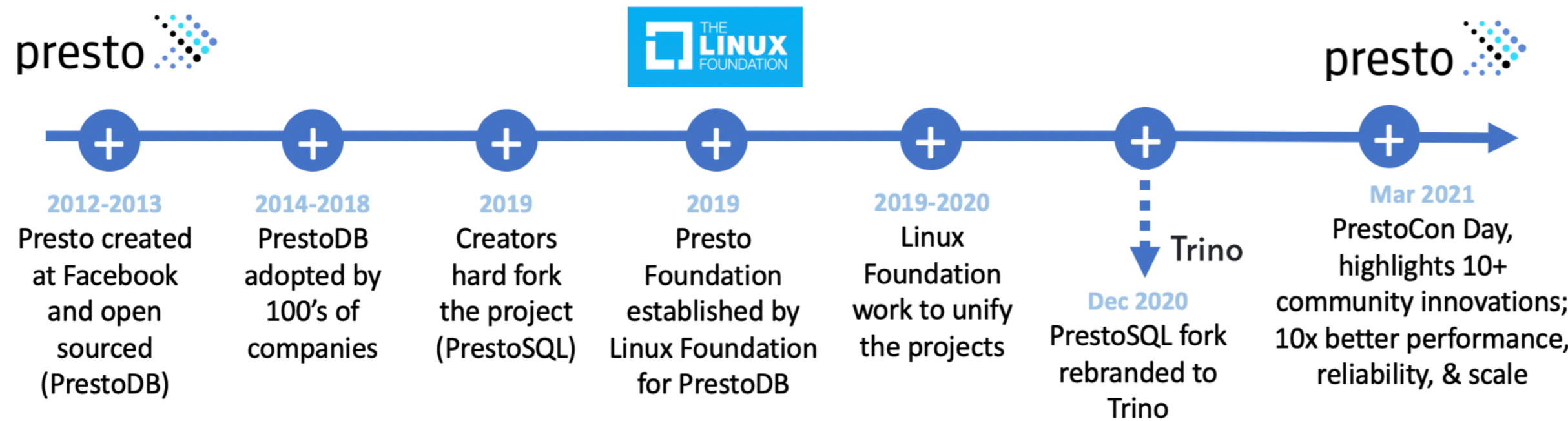
ICEBERG
 3rd party engines & applications

Apache Presto



Presto는 빠르고 안정적이며 효율적인 오픈 소스 SQL 쿼리 엔진입니다. Presto를 사용하면 대용량 데이터에 대해 1초 미만의 성능으로 대화형/Adhoc 쿼리를 실행할 수 있습니다.

A Brief History of Presto



Ahana



ahana™

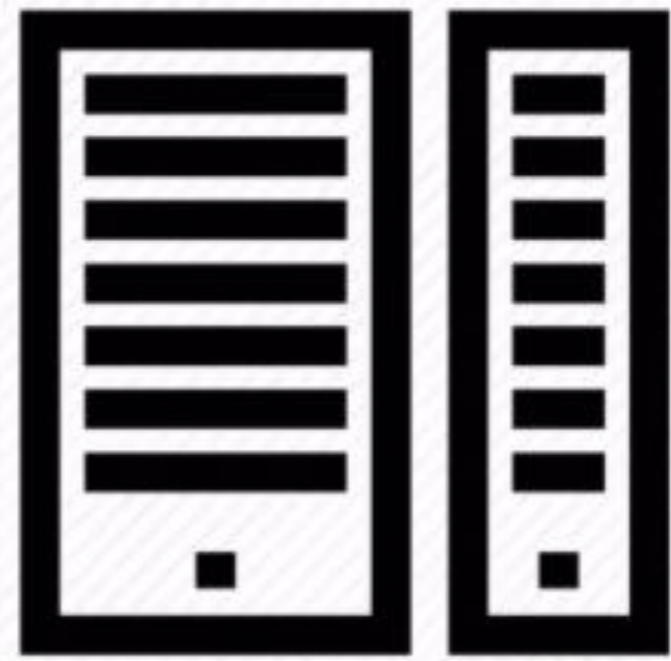
Presto의 유일한 SaaS 제공업체인 Ahana는 개방형 데이터 레이크를 단순화하려는 비전을 가지고 설립되었습니다. Cloud 상에서 유일한 Presto 관리형 서비스를 제공합니다. Ahana는 PrestoDB를 감독하는 Presto Foundation의 최고 회원(Primer member)입니다. 2023년 IBM에 인수되었습니다.

PrestoDB / Trino
PrestoDB와 Trino는 서로 다른 두 개의 github 저장소입니다.

Ahana는 두개의 프로젝트를 중심으로 기술 개발을 하고 있습니다. “**Project RaptorX**” 와 “**Velox**”가 그것입니다.

Project RaptorX – 10배 더 빠른 PrestoDB 구현

Presto @ Facebook Scale



50K+
Servers



~ 1 EB data
scan per day

코드명 "RaptorX 프로젝트"는 Metascale 페타바이트 워크로드에서 Presto를 10배 더 빠르게 만드는 것을 목표로 합니다.

• Project RaptorX의 배경

Presto는 스토리지-컴퓨팅 분리 Query 엔진.

이는 스토리지 계층과 컴퓨팅 계층 사이에서 막대한 양의 데이터를 검색하는 작업이 필요하며, 이는 네트워크 IO의 급격한 증가 발생 의미

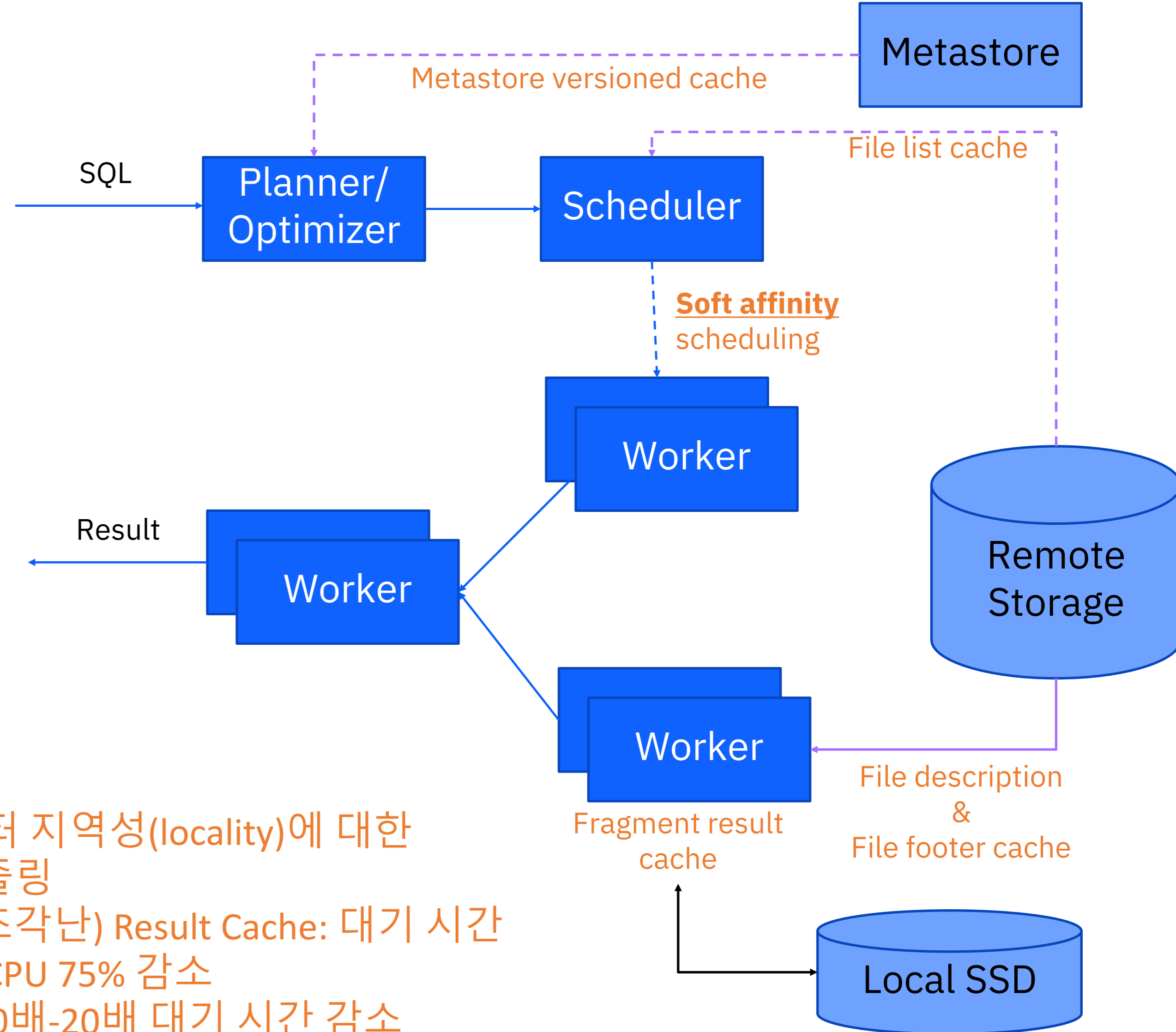
- Query 대기 시간에 대한 새로운 Challenge 발생
- 모든 데이터베이스와 마찬가지로 최적화된 I/O는 Presto의 중요한 관심사
- 가능하다면 I/O를 전혀 수행하지 않는 것이 우선순위
- 메모리 활용도와 캐싱 구조가 가장 중요하다는 것을 의미

Project RaptorX – Hierarchical Caching(계층적 Cash 관리)

- 기존 PrestoDB 사용자/워크로드에 대한 마이그레이션 없는 쿼리 가속화 솔루션 개발
- 데이터 복제 없이 페타바이트 규모의 대화형 워크로드를 가속화하는 것
- **쿼리 수명 주기에 대한 포괄적인 분석을 수행하여 I/O가 감소된 캐싱 증분의 개선 영역을 식별**
- 캐싱이 확실한 답이지만 캐시를 관리하는 데 많은 작업이 수행됨.
- **쿼리 수명 주기의 다양한 측면을 처리하는 다단계(Multi-level) 캐싱으로 해소**

- ✓ Metastore Cache: 대기 시간 20% 감소
- ✓ File List Cache: 쿼리당 100ms 감소
- ✓ File Desc & Footer Cache: CPU 및 대기 시간 40% 감소

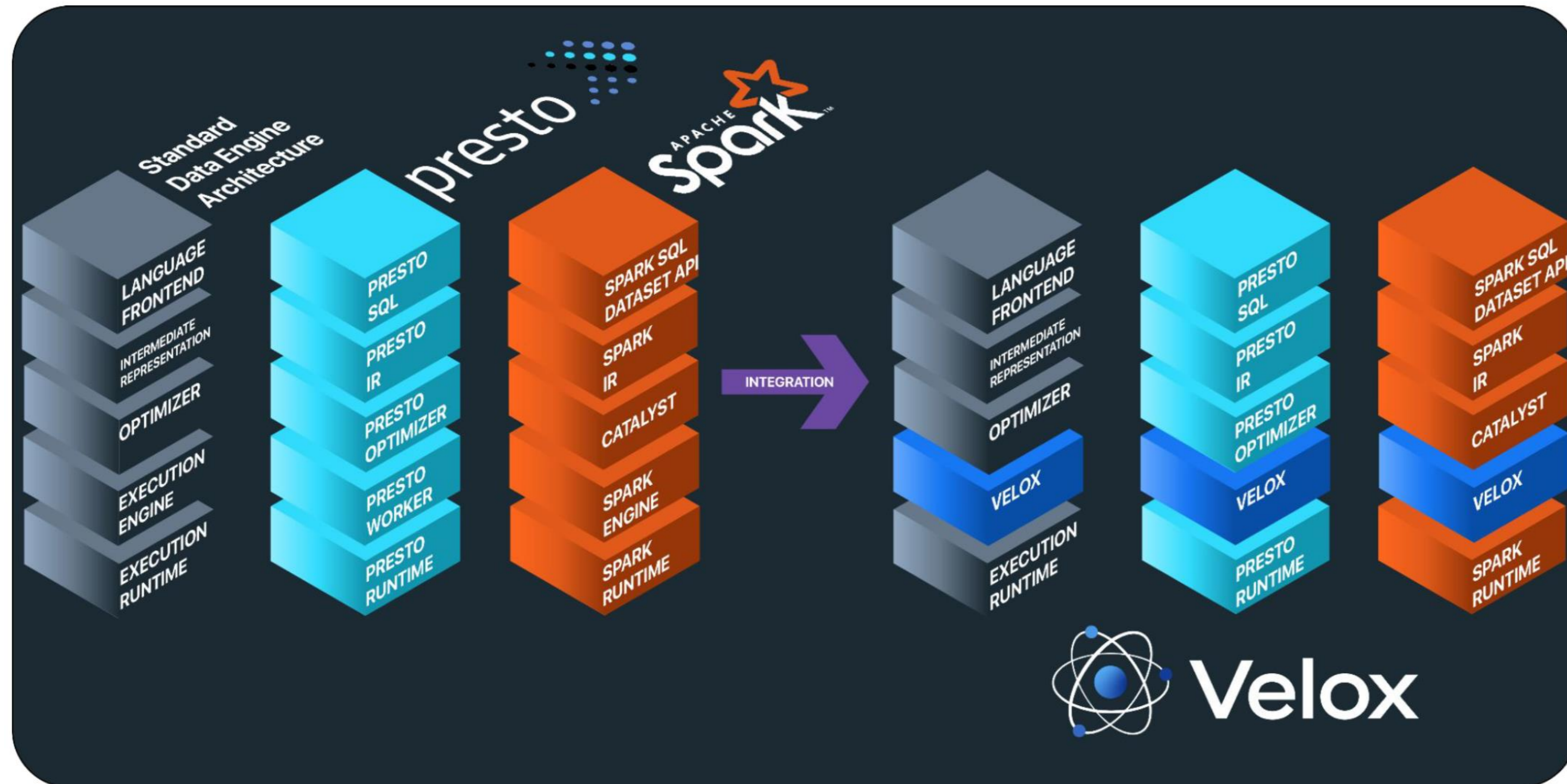
- ✓ 컴퓨팅/데이터 지역성(locality)에 대한 선호도 스케줄링
- ✓ Fragmented(조각난) Result Cache: 대기 시간 45% 감소 및 CPU 75% 감소
- ✓ Data Cache: 10배-20배 대기 시간 감소



Velox – 범용 vectorized database 가속 라이브러리

Velox는 Prestissimo* 프로젝트의 일환으로 Presto에 통합되고 있으며, 여기서 Presto의 Java worker는 Velox를 기반으로 하는 C++ 프로세스로 대체됩니다.

Velox는 SQL 파서가 아닌 라이브러이므로 데이터 프레임 계층, 글로벌 옵티마이저가 없으며 컨트롤 플레인입니다.

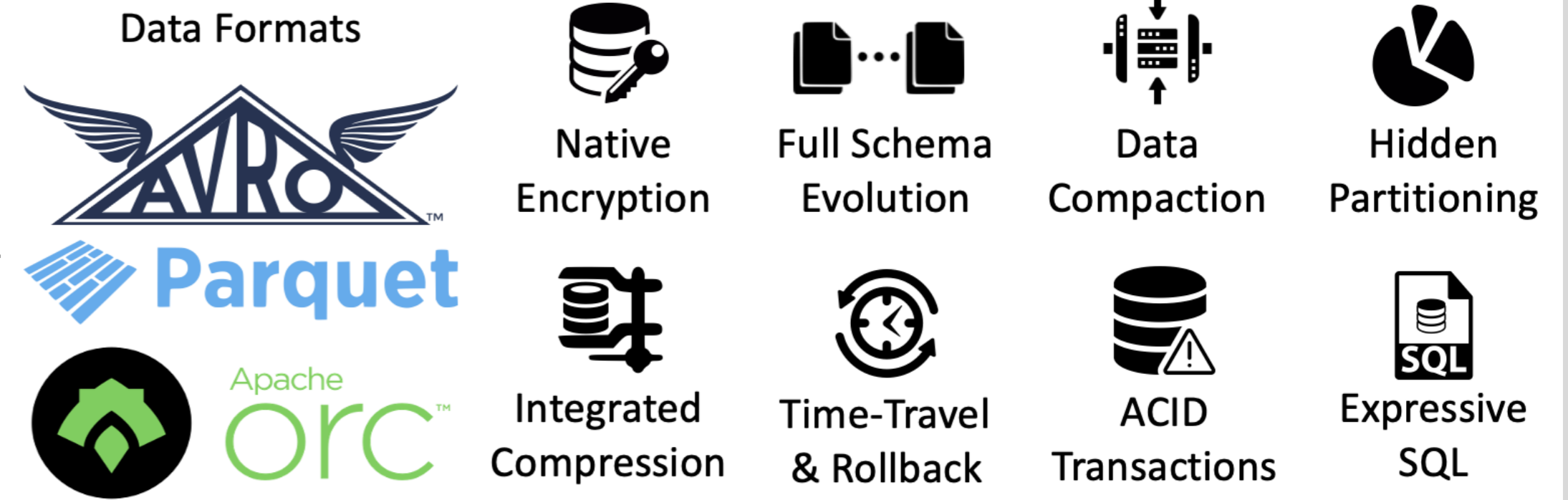


- 여러 런타임 엔진을 가속화 및 통합(C++)
- 벡터화된 실행(작업 및 명령 수준 모두)에 최적화됨
- 공통 기능: 함수, 조인, 필터 푸시다운, 정렬, 그룹화, 프로젝션 등
- 다양한 데이터 처리 프레임워크에서 사용
 - 범용 API
 - 고성능, 확장성
 - 배치, 대화형, 스트리밍, 분석, AI 전반에 걸쳐 일관된 의미 체계.

*Prestissimo : Presto on Velox

Apache Iceberg ICEBERG

- 초거대 데이터 분석을 위한 **고성능 개방형 테이블 Format**
- 빅 데이터 및 데이터 레이크에 **SQL의 단순성**을 제공
 - 엔진이 해당 사양에 따라 테이블과 상호 작용할 수 있는 API 및 라이브러리 집합
- **다양한 컴퓨팅 엔진** 사용이 가능한 유연성
 - Spark / Dremio / Trino / Presto / Flink / ...
- 완전한 **오픈 소스**
- 빠르게 업계 표준으로 부상
- **주요 기능 :**



- ✓ **Time travel** : Time travel을 통해 정확히 동일한 테이블 **Snapshot**을 사용하는 **재현 가능한 쿼리를 활성화**하거나 사용자가 변경 사항을 쉽게 검사. **버전 롤백**을 통해 사용자는 테이블을 양호한 상태로 재설정하여 문제를 신속하게 수정 가능
- ✓ **Schema evolution** : 열을 추가해도 "좀비" 데이터가 다시 표시되지 않음. 열 이름을 바꾸고 순서변경이 가능. 무엇보다도 스키마 변경 시 테이블을 다시 작성할 필요가 없음.
- ✓ **Partition evolution(Hidden Partitioning)** : 테이블의 행에 대한 파티션 값을 생성하는 오류가 발생하기 쉬운 작업을 처리하고, 불필요한 파티션과 파일을 자동으로 skip. 빠른 쿼리를 위해 추가 필터가 필요하지 않으며 데이터나 쿼리가 변경되면 테이블 레이아웃 업데이트를 수행
- ✓ **Transactions** : 최종적으로 일관된 클라우드 개체 저장소의 정확성 문제를 해결하도록 설계(ACID 보장)
- ✓ **Performance (테이블 메타데이터를 사용한 정리)** : 단일 테이블에 수십 페타바이트의 데이터가 포함될 수 있고 이러한 거대한 테이블도 분산 SQL 엔진 없이 읽을 수 있는 프로덕션에 사용 가능

빠르게 확산되고 있는 Iceberg

- 벤더 및 애플리케이션

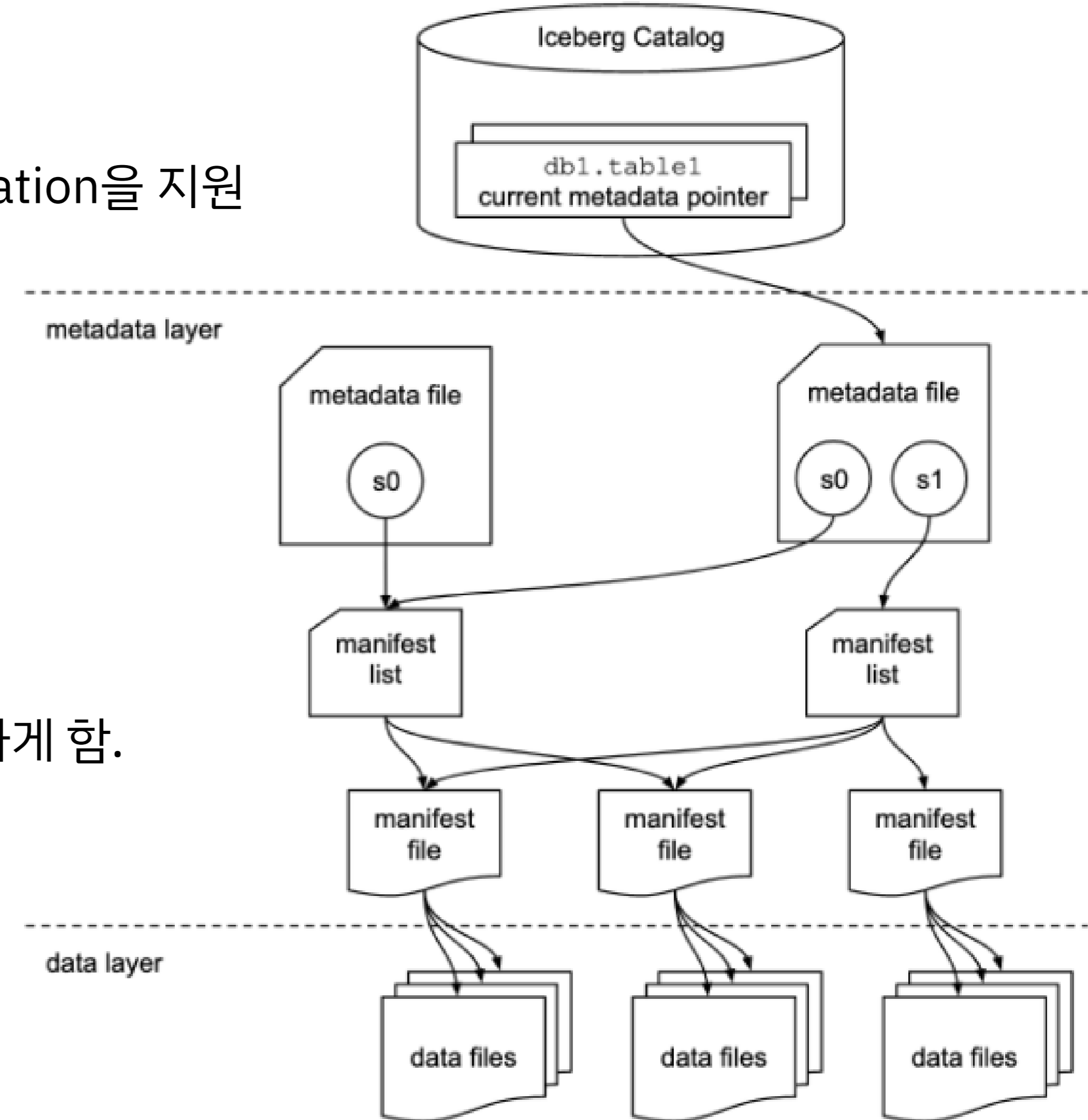
- 레이크하우스: Databricks Lakehouse, Snowflake, Microsoft Azure, AWS, Dremio, Presto, Trino, Google BigLake, **IBM watsonx.data**
- 데이터 플랫폼 : Spark, Cloudera Data Platform (Hive, Impala)
- 스트리밍 플랫폼 : Flink, Spark, Apache Beam
- 애플리케이션: Netflix, Adobe, Twitter, Apple, ...
 - Apple: several use cases in production, e. g. 2-3 PB of data per table, 2.5 mill files, < 5 sec response time, table management

- 핵심 기능

- Flexible compute
 - 데이터를 이동 없음. 여러 엔진(JVM뿐만 아니라 여러 언어)이 원활하게 작동
- **Iceberg FileIO (S3FileIO)**
 - **핵심 Iceberg 라이브러리와 기본 스토리지 간의 인터페이스**
 - **S3FileIO는 최적화된 보안 및 성능을 위해 최신 S3 기능을 채택**
- Maintaining Iceberg Tables
 - 테이블 마이그레이션: 현재 위치 마이그레이션 또는 Iceberg로의 새도 마이그레이션
 - 스냅샷 관리 및 메타데이터 관리: expire_snapshots, rewrite_data_files 등
 - 압축(여러 개의 작은 파일을 가져와서 더 적은 수의 큰 파일로 다시 쓰기), 스냅샷 만료, 분리된 파일 제거

Iceberg 아키텍처

- Architecture: 3 layers
 - **Iceberg Catalog**
 - Iceberg 테이블에 대한 현재 메타데이터 포인터
 - 현재 메타데이터 포인터를 업데이트하기 위한 Atomic operation을 지원
 - **Metadata layer** : 3 layers
 - Metadata : 테이블을 정의
 - Manifest List : 매니페스트 목록으로 Snapshot을 정의
 - Manifest : 데이터에 대한 메타데이터(최소/최대값)를 가짐
 - **Data Layer**
 - data format의 데이터
- Benefits, Capabilities, and Observations
 - 테이블 추상화는 물리적 최적화(숨겨진 분할, 압축, 진화)를 가능하게 함.
 - 메타데이터 트리를 통한 Snapshot 격리
 - 최적의 동시성 제어
 - 규모에 맞게 더 빠른 쿼리 계획 및 실행
 - 테이블에 쓰면 새 Snapshot을 생성
 - 만들기, 삽입, 병합, 행 수준 업데이트 및 삭제 지원



IBM watsonx.data feature highlights



왜 IBM 레이크하우스 인가?

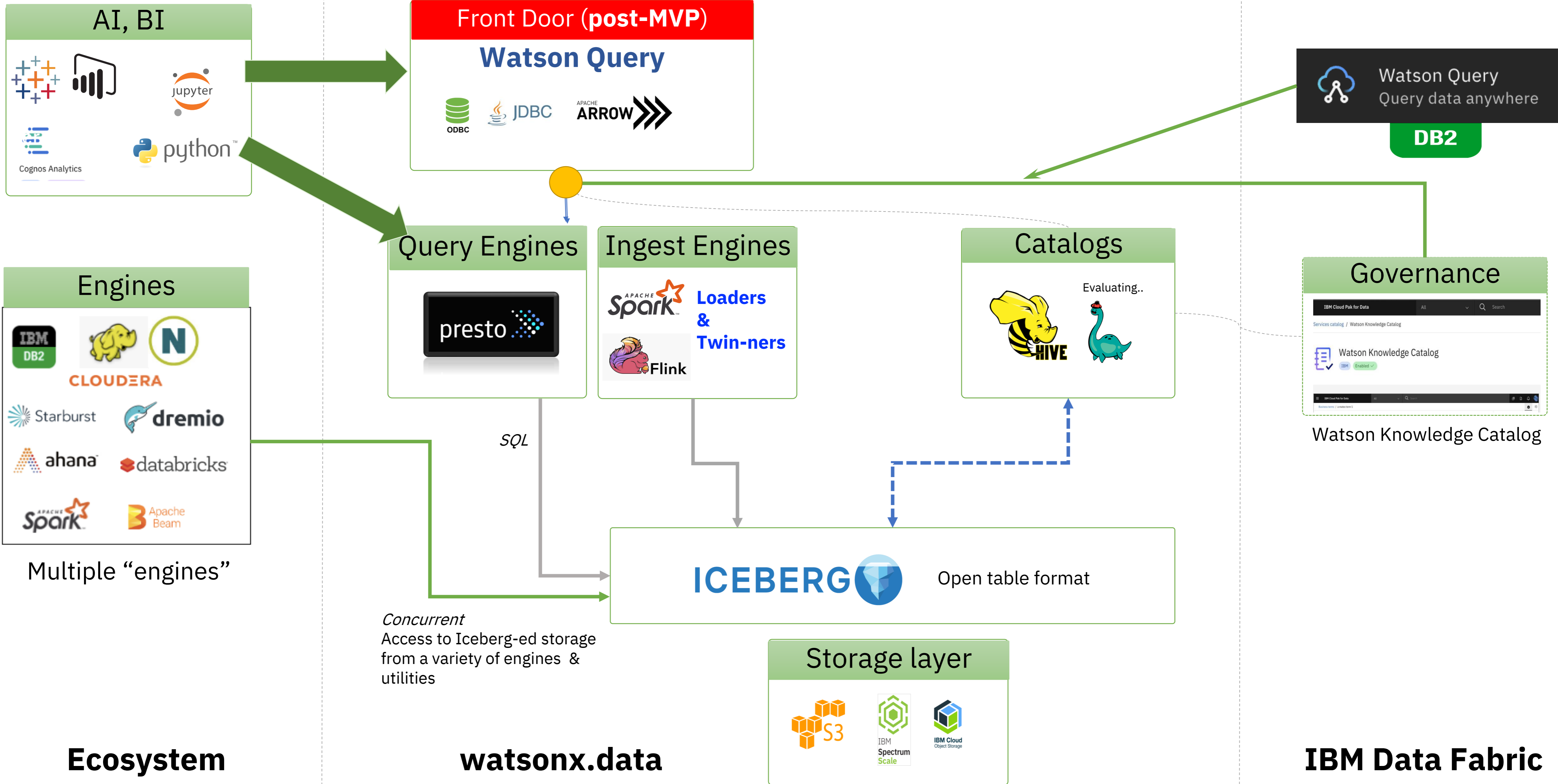
Lakehouse – 구현의 어려움

- 데이터 리포지토리가 고정되어 있습니다.
 - 기존 창고 및 호수에서의 마이그레이션은 복잡합니다.
 - 교체에 대한 내성
- 어디서나 데이터 사용
 - 데이터를 다른 사이트 및 클라우드로 이동하는 것이 항상 가능한 것은 아닙니다
 - 세이프 하버 및 기타 규정 준수는 무시할 수 없습니다
- 가격/성능
 - Compute Elasticity - 수요에 따라 확장 및 축소
 - 저렴하지만 느린 스토리지
- 응용 프로그램 리팩터링
 - 새로운 프로토콜로 응용 프로그램 및 공급업체 도구를 다시 구현하는 데 비용이 많이 듭니다.
- 운영 복잡성
 - 휴대성 – 다양한 폼 팩터의 어느 곳에서나 실행할 수 있는 기능
 - 진정한 탄력적 및 “on-demand” Lakehouse 인스턴스의 수명 주기 관리

IBM Data Lakehouse의 차별성

- 데이터 수집을 통한 손쉬운 데이터 통합 지원*
 - 새 레이크하우스로 확장하는 동안 기존 리포지토리를 유지합니다.
 - 기존 레이크하우스나 새 레이크하우스 또는 둘 다 사용
 - 분석을 가속화하는 지속적인 데이터 복제의 단순성
 - ❖ * 데이터 복사를 방지하기 위해 Virtual Lakehouses에 대한 federated 쿼리.
- 엔진 내 정책 시행 및 **데이터 주권**
 - 수집 중 데이터 버전 관리 및 적용.
 - 관리되는 데이터의 구체화/캐싱
- **성능 최적화**
 - Watson Query 분석 최적화, 쿼리 재작성, 푸시다운, OLAP 및 기타 분석 최적화(IBM IP) 워크로드 관리 및 격리
 - Velox, Prestissimo 및 IBM 확장을 통한 실행 가속화
- SQL 및 드라이버 유선 프로토콜 호환성
 - IBM Db2, IBM NZ와 대화하는 기존 앱 유지
 - PostgreSQL 또는 MySQL 방언, URL만 전환
- 개방형 모듈식 레이크하우스, 레이크하우스에 적용된 "마이크로 서비스" 아키텍처.
 - 다중 클라우드의 IBM SaaS
 - Red Hat OpenShift – 프라이빗 클라우드를 위한 운영
 - ISV 및 솔루션과 여러 엔진/공급업체를 위한 준비 – 워크로드에 적합한 크기

IBM watsonx.data 오픈 소스 아키텍처



Ecosystem

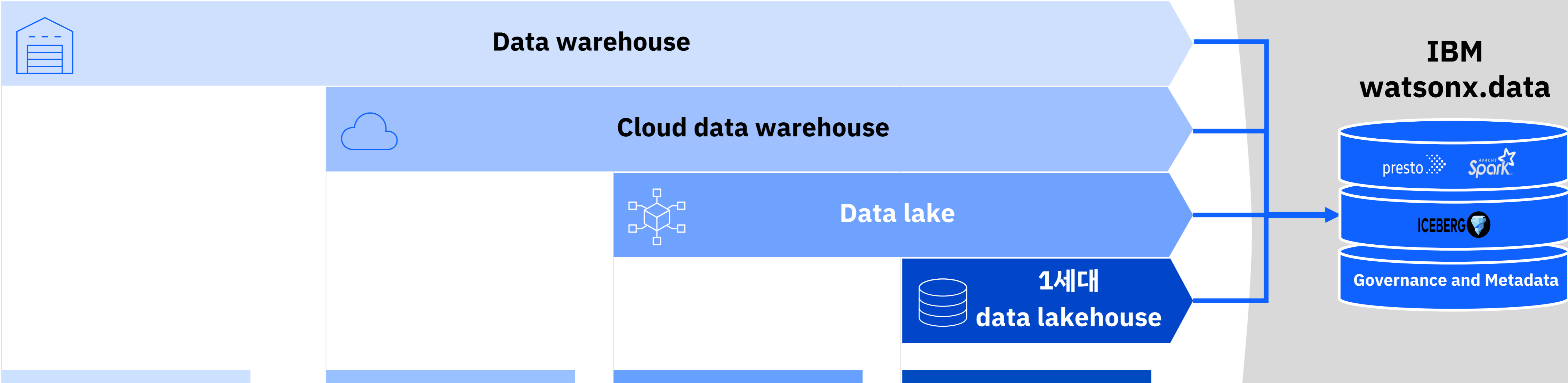
watsonx.data

IBM Data Fabric

IBM watsonx.data는 기업들이 직면하고 있는 비용과 복잡성을 극복할 수 있도록 설계된 차세대 레이크 하우스

하이브리드 클라우드에서
관리되는 데이터 및 AI 워크로드에
최적화된 유일한 개방형 데이터
저장소

Late 90s Early 2000s Present



높은 초기 비용
구조화된 데이터만
ETL 필요
공급업체 종속
제한된 확장성

높은 복잡성
열악한 데이터 품질
제한된 성능
유지 관리 비용 과다

데이터 마이그레이션
공급업체 종속
높은 비용
제한된 AI/ML 사용 사례

제한된 사용 사례(예: BI
전용 또는 AI/ML 전용)

비용을 50% 절감하고,
데이터에 100% 액세스하고,
10분 안에 시작 가능

IBM watsonx.data의 주요 구성요소 개요

: 멀티 엔진, 개방형 테이블 형식 및 기본 제공 엔터프라이즈 거버넌스



watsonx.data

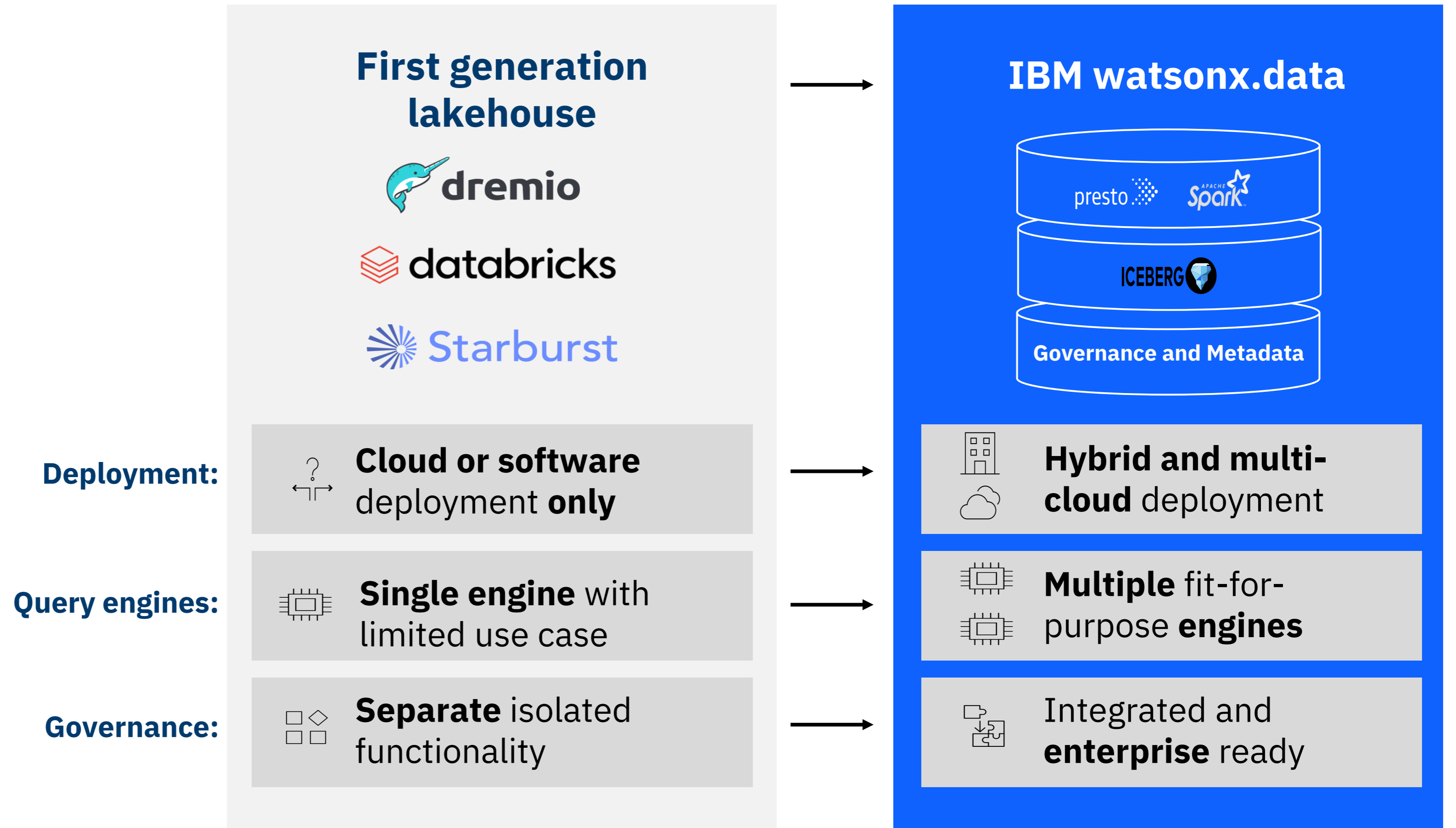
어디서나 모든 데이터에 대해 AI 워크로드 확장

Data and AI 워크로드를 위한 개방형 레이크하우스 아키텍처 기반의 데이터 저장소

IBM watsonx.data는 1세대 레이크하우스에서 진화한 차세대 기술입니다.

레이크 하우스 (Lakehouse)

레이크하우스는 데이터 레이크와 웨어하우스를 최고 품질의 단일 통합 플랫폼으로 결합하여, 매우 복잡한 데이터 변환과 다양한 사용 사례를 지원



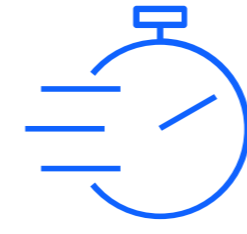
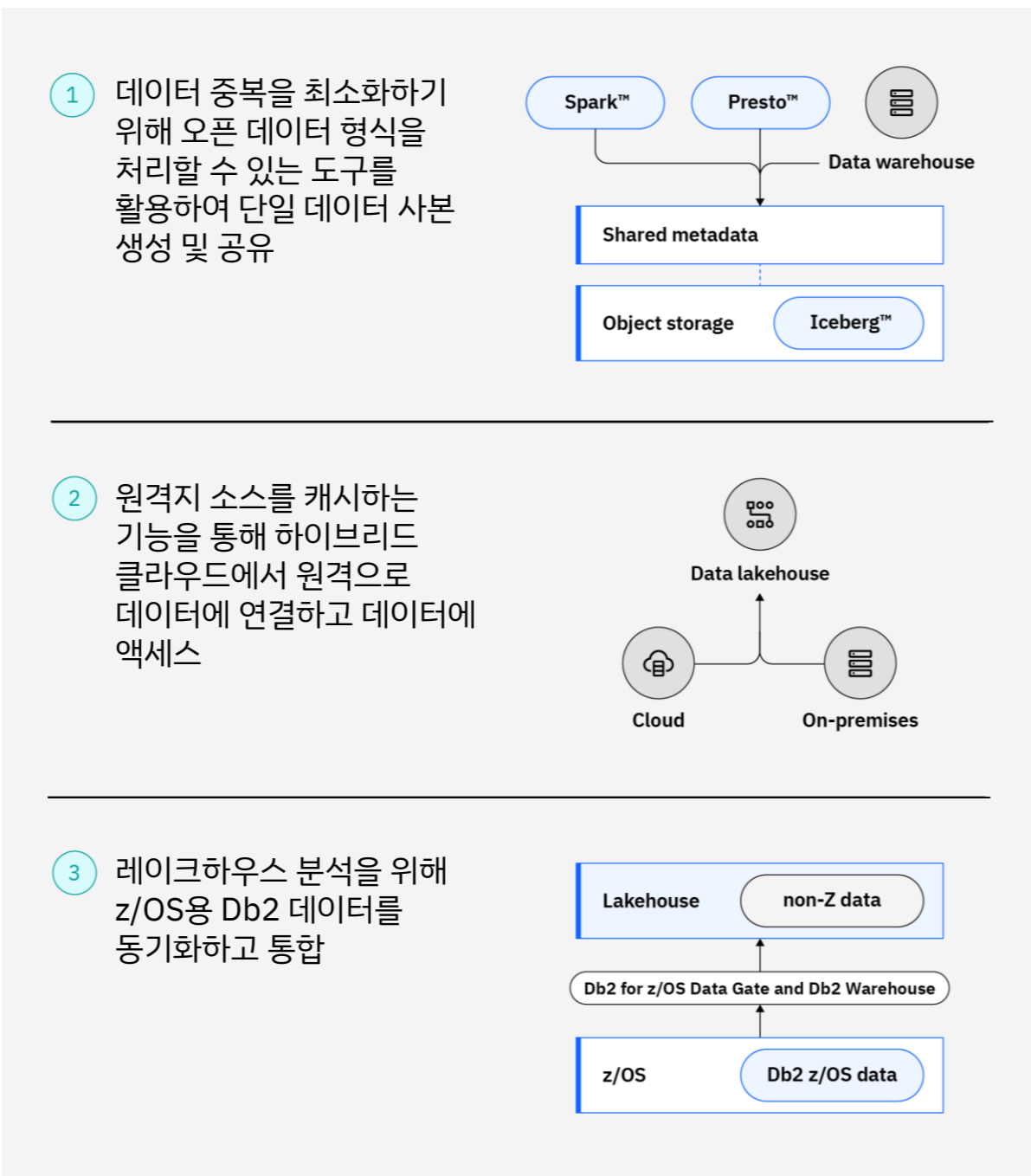
- IBM watsonx.data는 **Multi 쿼리 엔진**을 갖춘 유일한 레이크하우스로, 고객이 **워크로드를 최적의 엔진과 매칭하여 비용과 성능을 최적화**
- **단일 창**에서 모든 워크로드를 실행하여 비용과 성능을 향상시키면서 **편리함까지 향상**
- **하이브리드 클라우드 및 멀티클라우드 환경에 대한 완벽한 지원**으로 어디에나 배포
- **여러 엔진에서 메타데이터를 공유**하여, 카탈로그를 다시 작성할 필요가 없이 가치 창출 시간을 단축하는 동시에 거버넌스를 보장하고, **비용이 많이 드는 구현 노력을 절약**

어디서나 모든 데이터에 대해 AI 워크로드 확장

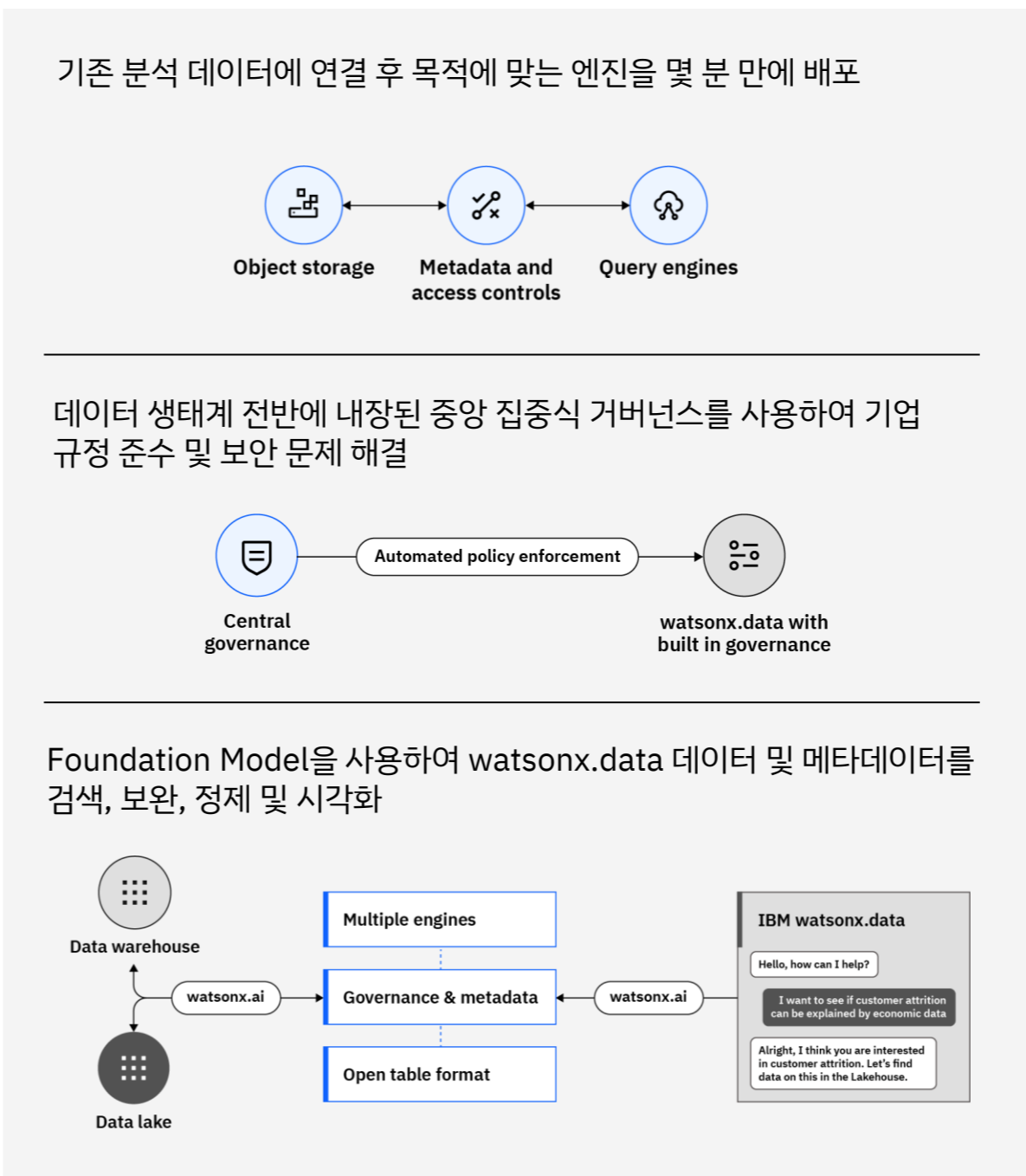
Data and AI 워크로드를 위한 개방형 레이크하우스 아키텍처 기반의 데이터 저장소



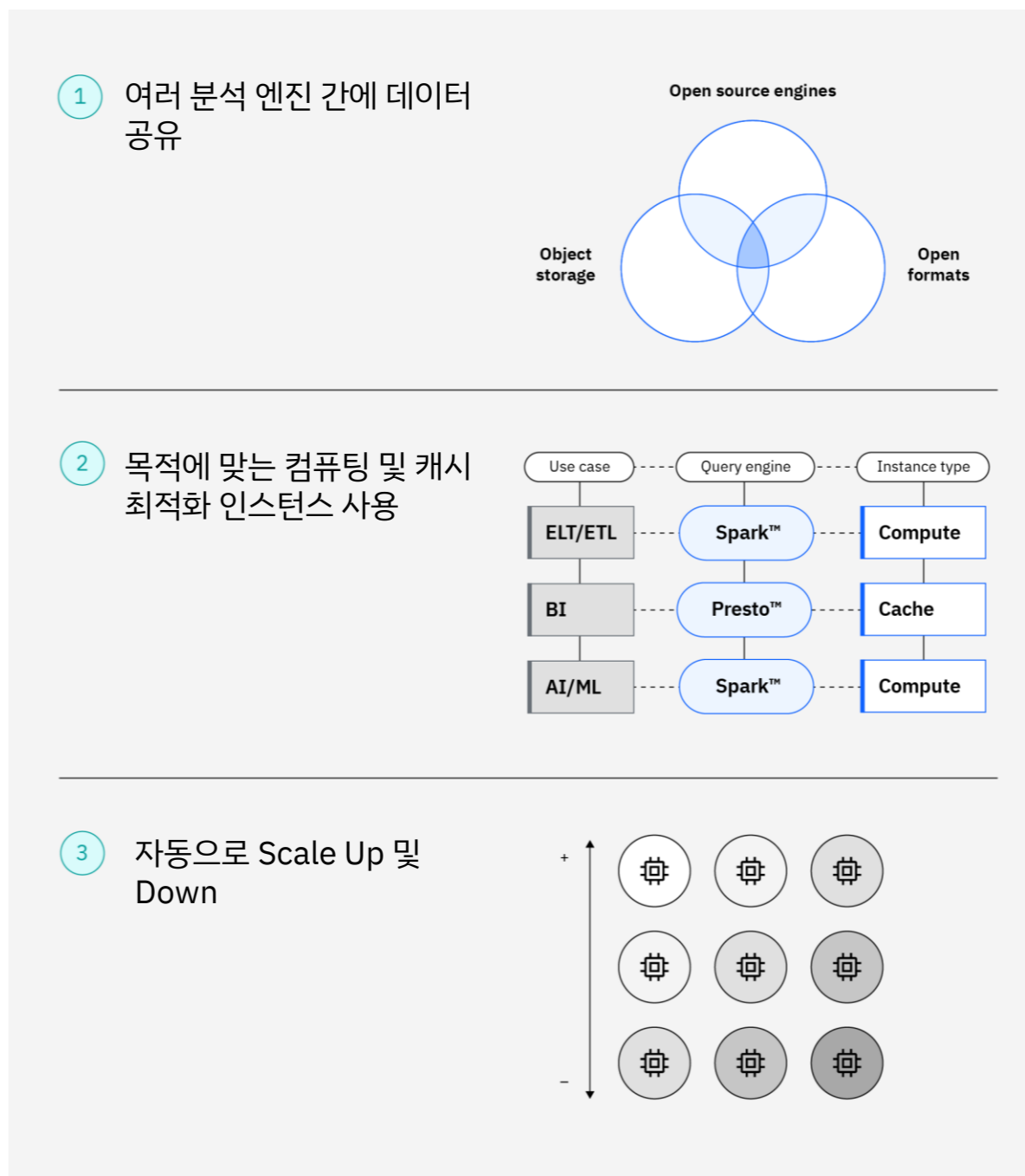
모든 데이터에 대한 single point 액세스



내장된 거버넌스, 보안 및 자동화로 몇 분 만에 사용



데이터 웨어하우스 비용을 최대 50%* 절감



*When comparing published 2023 list prices normalized for VPC hours of IBM watsonx.data to several major cloud data warehouse vendors. Savings may vary depending on configurations, workloads and vendors.

IBM