



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE CIENCIAS EXACTAS, NATURALES Y AMBIENTALES
CARRERA DE CIENCIA DE DATOS

Análisis comparativo técnico y operativo de la arquitectura Lakehouse frente a modelos
tradicionales de gestión de datos.

Monografía previa a la obtención del título de: Ingeniería en Ciencia de Datos

Dylan Ariel Flores Muñoz

Quito, 2026

DEDICATORIA

A mi familia, especialmente a mis padres y a mi hermana por haberme dado las herramientas para llegar hasta aquí y por acompañarme en cada etapa de este proceso.

TABLA DE CONTENIDOS

Contenido

1.	RESUMEN	5
2.	ABSTRACT	7
3.	INTRODUCCIÓN	9
4.	OBJETIVOS	12
4.1	OBJETIVO GENERAL	12
4.2	OBJETIVOS ESPECÍFICOS	12
5.	DESARROLLO TEÓRICO	13
5.1	ANTECEDENTES Y EVOLUCIÓN DE LAS ARQUITECTURAS DE DATOS	13
5.1.1	EL CONTEXTO DE LA ERA DEL BIG DATA	13
5.1.2	PRIMERA GENERACIÓN: EL APOGEO Y LA CRISIS DEL DATA WAREHOUSE	14
5.1.3	SEGUNDA GENERACIÓN: EL DATA LAKE (DL) Y EL SURGIMIENTO DE LOS PANTANOS DE DATOS	17
5.1.4	LA CRISIS DE LAS ARQUITECTURAS HÍBRIDAS (ARQUITECTURAS DE DOS NIVELES)	21
5.2	FUNDAMENTOS DE LA ARQUITECTURA DATA LAKEHOUSE	24
5.2.1	DEFINICIÓN Y LOS CINCO DESAFÍOS TÉCNICOS (C1-C5)	24
5.2.2	LOS OCHO REQUISITOS TÉCNICOS FUNDAMENTALES DEL LAKEHOUSE	27
5.2.3	CAPAS ARQUITECTÓNICAS DEL MODELO LAKEHOUSE	29
5.2.4	CARACTERÍSTICAS AVANZADAS DE GESTIÓN Y EVOLUCIÓN DE ESQUEMAS	33
5.3	TECNOLOGÍAS HABILITADORAS Y FORMATOS DE TABLA	35
5.3.1	DELTA LAKE	36
5.3.2	APACHE ICEBERG	36
5.3.3	APACHE HUDI	37
5.3.4	ANÁLISIS COMPARATIVO Y DE RENDIMIENTO	38
5.4	ANÁLISIS COMPARATIVO: GOBERNANZA, ESCALABILIDAD, RENDIMIENTO Y COSTOS	39
5.4.1	GOBERNANZA, EVOLUCIÓN DE ESQUEMAS Y CONTROL DE CALIDAD 39	
5.4.2	ESCALABILIDAD, FORMATOS ABIERTOS Y PREVENCIÓN DE LAS DEPENDENCIAS TECNOLÓGICAS	41

5.4.3	RENDIMIENTO, OPTIMIZACIÓN ALGORÍTMICA Y REDUCCIÓN DE COSTOS	43
5.4.4	UNIFICACIÓN DE CARGAS DE TRABAJO: INTELIGENCIA DE NEGOCIOS E INTELIGENCIA ARTIFICIAL	44
5.5	CASOS DE ESTUDIO Y APLICABILIDAD	46
5.5.1	SECTOR DE MANUFACTURA Y COMPONENTES AUTOMOTRICES	47
5.5.2	SECTOR DE SALUD PÚBLICA Y EPIDEMIOLOGÍA	47
5.5.3	SECTOR MARÍTIMO Y MONITORIZACIÓN DE EMBARCACIONES (LOGÍSTICA).....	48
5.5.4	SECTOR DE INTELIGENCIA GEOESPACIAL Y CIUDADES INTELIGENTES	48
5.5.5	INVESTIGACIÓN BIOMÉDICA Y MEGA-BIOBANCOS	49
6.	CONCLUSIONES	50
7.	RECOMENDACIONES.....	53
8.	REFERENCIAS BIBLIOGRÁFICAS	55

1. RESUMEN

La explosión del volumen, variedad y velocidad de los datos en la era del Big Data evidenció las limitaciones estructurales de las arquitecturas tradicionales de gestión de información. Los modelos de Data Warehouse, aunque eficaces para el análisis estructurado, resultaron inflexibles ante datos semiestructurados y no estructurados, costosos de escalar y cerrados a las cargas de trabajo modernas de inteligencia artificial. El Data Lake, por su parte, ofreció flexibilidad y bajo costo de almacenamiento, pero careció de garantías transaccionales, gobernanza efectiva y rendimiento analítico suficiente. Las arquitecturas híbridas de dos niveles que combinaban ambos modelos tampoco resolvieron el problema: introdujeron complejidad operativa, datos desactualizados y duplicación de costos. Ante este escenario, la presente monografía analiza la arquitectura Data Lakehouse para determinar sus ventajas técnicas y operativas frente a estos modelos tradicionales, a partir de una revisión documental comparativa orientada a entornos con alta exigencia de análisis y procesamiento de datos. El desarrollo teórico aborda, en primer lugar, los principios, componentes y tecnologías fundamentales que sustentan el modelo Lakehouse, incluyendo su capa transaccional de metadatos, sus ocho requisitos técnicos y las tres tecnologías habilitadoras principales: Delta Lake, Apache Iceberg y Apache Hudi. En segundo lugar, se comparan las características funcionales y estructurales de las tres arquitecturas en términos de gobernanza, escalabilidad, rendimiento y costos, identificando las ventajas diferenciales del Lakehouse. Finalmente, se examinan cinco casos documentados de implementación en sectores estratégicos: manufactura automotriz, salud pública, logística marítima, inteligencia geoespacial e investigación biomédica. Los resultados del análisis confirman que el Data Lakehouse representa una evolución arquitectónica sustancial, capaz de unificar cargas de trabajo de inteligencia de negocios e inteligencia artificial sobre una única fuente de verdad, con mejoras verificables en rendimiento, reducción de costos y gobernanza de datos,

consolidándose como la plataforma de referencia para organizaciones con alta exigencia analítica.

Palabras clave: arquitectura de datos, Big Data, Data Lake, Data Lakehouse, Data Warehouse.

2. ABSTRACT

The explosion in volume, variety, and velocity of data during the Big Data era exposed the structural limitations of traditional data management architectures. Data Warehouse models, while effective for structured analysis, proved inflexible when handling semi-structured and unstructured data, costly to scale, and incompatible with modern artificial intelligence workloads. Data Lakes, on the other hand, offered flexibility and low storage costs, but lacked transactional guarantees, effective governance, and sufficient analytical performance. The two-tier hybrid architectures that combined both models failed to resolve these issues, instead introducing operational complexity, stale data, and duplicated costs. Against this backdrop, this monograph analyzes the Data Lakehouse architecture to determine its technical and operational advantages over these traditional models, based on a comparative documentary review focused on environments with high analytical and data processing demands. The theoretical development first addresses the fundamental principles, components, and technologies underpinning the Lakehouse model, including its transactional metadata layer, its eight core technical requirements, and the three main enabling technologies: Delta Lake, Apache Iceberg, and Apache Hudi. Second, the functional and structural characteristics of the three architectures are compared in terms of governance, scalability, performance, and costs, identifying the differential advantages of the Lakehouse. Finally, five documented implementation cases are examined across strategic sectors: automotive manufacturing, public health, maritime logistics, geospatial intelligence, and biomedical research. The findings confirm that the Data Lakehouse represents a substantial architectural evolution, capable of unifying business intelligence and artificial intelligence workloads on a single source of truth, with verifiable improvements in performance, cost reduction, and data governance, establishing itself as the reference platform for organizations with high analytical demands.

Keywords: Big Data, Data Lake, Data Lakehouse, Data Warehouse, data architecture.

3. INTRODUCCIÓN

En los últimos años, con la llegada de la era digital y el acceso a más fuentes disponibles, los datos y la información se han convertido en uno de los activos más importantes para las empresas alrededor del mundo (Janssen et al., 2024). Las organizaciones se enfrentan a grandes volúmenes de datos provenientes de distintas fuentes y en varios formatos. Para generar predicciones, llegar a conclusiones y tomar decisiones fundamentadas, se requiere procesar y analizar toda esa información, lo que demanda plataformas capaces de ofrecer las capacidades y funciones necesarias (Bentaib et al., 2024).

Tradicionalmente, las arquitecturas de Data Warehouse o Data Lake han sido las más utilizadas para construir estas plataformas, siendo adoptadas por industrias de todo tipo (Thalpati, 2024). Sin embargo, los Data Warehouses presentan limitaciones para manejar datos semiestructurados o no estructurados, implican altos costos de implementación y mantenimiento, y presentan dificultades de escalabilidad. Por otro lado, los Data Lakes, aunque permiten almacenar datos en bruto de forma flexible y a bajo costo, carecen de funciones nativas para manejar transacciones, controlar versiones o garantizar calidad, lo que dificulta la gobernanza y el análisis en tiempo real (Janssen et al., 2024). En consecuencia, se busca un enfoque que combine las ventajas de ambos modelos (Gade, 2022).

En respuesta, surge la arquitectura Data Lakehouse, que integra en un solo entorno la flexibilidad y bajo costo de los Data Lakes con la eficacia y rendimiento de los Data Warehouses (Cherradi y El Haddadi, 2024). En el panorama actual, se evidencia una transición masiva hacia su adopción, con más del 70% de profesionales de datos proyectándola como su arquitectura primaria para unificar cargas de trabajo de IA y analítica (Commvault, 2025), superando así las limitaciones de los modelos híbridos anteriores (Mezzoudj et al., 2025).

Esta evolución tecnológica se ha materializado mediante capas de gestión transaccional (ACID) y metadatos sobre formatos de tabla abiertos como Delta Lake, Iceberg y Hudi (Thomas et al., 2025), estructuradas bajo arquitecturas multi-zona (ingesta, procesamiento y acceso) que previenen la degradación de los repositorios en “pantanos de datos” (Azzabi et al., 2024). Según benchmarks técnicos, aunque existen disyuntivas de rendimiento (hasta 10 veces de diferencia entre estrategias Copy-on-Write y Merge-on-Read) (Jain et al., 2023), las implementaciones exitosas han reportado reducciones del 40–70 % en costos de almacenamiento y aceleración de hasta 8,5 veces en consultas analíticas complejas respecto a arquitecturas heredadas (Chippada et al., 2025).

Sabiendo que no existen muchos estudios comparativos sobre la arquitectura Lakehouse frente a los modelos anteriores, ya que se trata de un concepto relativamente reciente (Begoli et al., 2021), surge la siguiente pregunta: ¿Qué componentes, principios técnicos y ventajas operativas caracterizan a la arquitectura Data Lakehouse frente a los modelos tradicionales de Data Lake y Data Warehouse, en entornos con alta exigencia de análisis y procesamiento de datos?

En este contexto, las plataformas de datos juegan un papel central al permitir la recolección, procesamiento y análisis de datos y metadatos provenientes de múltiples fuentes. La adopción del modelo Lakehouse, además de proyectarse como sustituto de arquitecturas tradicionales como Data Warehouse (Zaharia et al., 2021), ofrece mejoras notables en costos y rendimiento analítico (Schneider et al., 2023), y proporciona un marco integral capaz de afrontar fortalezas y debilidades de los enfoques previos (Cherradi y El Haddadi, 2024). No obstante, su reciente aparición deja aún vacíos en definiciones precisas, criterios de implementación, tecnologías habilitadoras (Schneider et al., 2024) y estudios comparativos con modelos anteriores (Janssen et al., 2024).

Dado este escenario, la presente investigación adquiere relevancia al analizar cómo esta arquitectura permite integrar entornos fragmentados en una solución unificada. Se propone como una evolución sustancial capaz de fortalecer la toma de decisiones estratégicas y mejorar la calidad analítica y operativa en sectores con alta exigencia de procesamiento (Gade, 2022). Los beneficios esperados incluyen una comprensión técnica y conceptual más clara del modelo Lakehouse, identificando sus diferencias frente a enfoques tradicionales. El análisis de sus componentes y de casos documentados permitirá contextualizar su aplicación real y sus implicaciones prácticas. Asimismo, este trabajo brindará una sistematización clara y rigurosa de un modelo en consolidación, ofreciendo un recurso académico y profesional útil para futuras investigaciones, comparaciones técnicas y procesos de formación en torno a las arquitecturas modernas de gestión de datos.

4. OBJETIVOS

4.1 OBJETIVO GENERAL

Analizar la arquitectura Data Lakehouse para determinar sus ventajas técnicas y operativas frente a modelos tradicionales como Data Lake y Data Warehouse a partir de una revisión documental comparativa orientada a entornos con alta exigencia de análisis y procesamiento de datos.

4.2 OBJETIVOS ESPECÍFICOS

- 4.2.1 Describir los principios, componentes y tecnologías fundamentales que sustentan la arquitectura Data Lakehouse, considerando su origen y evolución frente a modelos previos.
- 4.2.2 Comparar las características funcionales y estructurales de las arquitecturas Data Lake, Data Warehouse y Data Lakehouse, identificando sus diferencias en términos de gobernanza, escalabilidad, rendimiento y costos.
- 4.2.3 Examinar casos documentados de implementación del modelo Lakehouse en sectores estratégicos para evaluar su aplicabilidad práctica y sus implicaciones operativas.

5. DESARROLLO TEÓRICO

5.1 ANTECEDENTES Y EVOLUCIÓN DE LAS ARQUITECTURAS DE DATOS

5.1.1 EL CONTEXTO DE LA ERA DEL BIG DATA

El volumen de datos generados por empresas e instituciones científicas creció de forma tan acelerada que obligó a replantear los fundamentos mismos de la gestión de la información (Thomas et al., 2025). En sus inicios, durante los años noventa, ese cúmulo de datos era visto con preocupación en organizaciones como la NASA: las limitaciones físicas de los discos duros de la época convertían cualquier acumulación masiva en un problema más que en un recurso (Cappa et al., 2025). Con el tiempo, esa percepción cambió por completo. Para 2021, el volumen global de datos generados superaba los 79 zettabytes, reflejo de un aumento del 5.000 % en las interacciones de datos durante la década previa (Mezzoudj et al., 2025). Las proyecciones apuntaban a que, para 2024, esa cifra alcanzaría los 149 zettabytes, confirmando lo que ya muchos anticipaban: que los datos se convertirían en la "nueva materia prima de los negocios" (Cappa et al., 2025).

Este desbordamiento informativo dio lugar al paradigma del Big Data, definido no solo por su cantidad, sino por un conjunto de cinco dimensiones conocidas como las "5 V": Volumen, Variedad, Velocidad, Veracidad y Valor (Mezzoudj et al., 2025; Cappa et al., 2025). Las organizaciones empezaron a producir información por canales muy distintos: plataformas digitales, dispositivos IoT, redes sociales, registros de máquina y aplicaciones SaaS (Sundar et al., 2022). Ante esa diversidad creciente, donde la mayor parte de la información corporativa pasó a ser semiestructurada o no estructurada (texto, audio, video, imágenes),

los sistemas de almacenamiento convencionales mostraron sus límites, impulsando el surgimiento de nuevas generaciones tecnológicas (Mezzoudj et al., 2025).

La literatura actual reconoce que gestionar esta complejidad ya no es opcional: dominarla es un prerequisite para la innovación. El uso efectivo del Big Data permite a las organizaciones acortar sus ciclos de desarrollo, adaptarse con mayor agilidad a mercados volátiles y transferir nuevas tecnologías con mayor eficiencia (Cappa et al., 2025). Sin embargo, esa abundancia tiene también un lado problemático. Cuando los datos no se procesan en plataformas adecuadas, las organizaciones incurren en costos elevados de gestión y seguridad, y caen en lo que se denomina "info-obesidad": un estado donde el exceso de información no procesable entorpece, en lugar de facilitar, la toma de decisiones (Cappa et al., 2025). Precisamente esa tensión entre el valor potencial del Big Data y los riesgos de su mal manejo fue la que impulsó la evolución hacia arquitecturas de datos más robustas.

5.1.2 PRIMERA GENERACIÓN: EL APOGEO Y LA CRISIS DEL DATA WAREHOUSE

La primera generación de plataformas analíticas estuvo marcada por el Data Warehouse (Almacén de Datos), cuyo concepto tomó forma a finales de la década de 1980 gracias a los investigadores de IBM Barry Devlin y Paul Murphy, con el propósito de resolver el flujo de información hacia los entornos de soporte de decisiones (Nambiar & Mundra, 2022). Bill Inmon, considerado el padre de esta arquitectura, la definió como una colección de datos orientada a temas, integrada, no volátil y variante en el tiempo, diseñada exclusivamente para apoyar las decisiones gerenciales (Nambiar & Mundra, 2022; Mezzoudj et al., 2025). En la práctica, esto significa que el Data Warehouse elimina información irrelevante para centrarse en un dominio específico, estandariza datos provenientes de múltiples fuentes, conserva el

historial para el análisis de tendencias y, una vez cargada la información, no la modifica con frecuencia, garantizando una base estable para el análisis (Mezzoudj et al., 2025).

A nivel estructural, el Data Warehouse no es un simple repositorio: está organizado en tres niveles interconectados por flujos de trabajo rígidos, como se aprecia en la Figura 1 (Mezzoudj et al., 2025; Nambiar & Mundra, 2022):

- **Nivel Inferior (Bottom Tier):** Actúa como el servidor de base de datos relacional. La información proveniente de bases de datos transaccionales (OLTP), sistemas ERP o fuentes externas debe ser extraída, limpiada, transformada y cargada mediante procesos ETL (Extract, Transform, Load) (Nambiar & Mundra, 2022; Grandhe, 2025). Este nivel opera bajo un esquema en la escritura (schema-on-write): los datos deben ajustarse a modelos dimensionales predefinidos (como esquemas en estrella o copo de nieve) antes de ser almacenados (Janssen et al., 2024; Schneider et al., 2024). Frecuentemente, la información centralizada se subdivide en Data Marts, repositorios más acotados que agrupan subconjuntos de datos por departamento o tema para agilizar las consultas (Janssen et al., 2024; Mezzoudj et al., 2025).
- **Nivel Medio (Middle Tier):** Alberga el servidor OLAP (Procesamiento Analítico en Línea), que reorganiza los datos en un formato multidimensional (cubos de datos) permitiendo a los usuarios extraer y consultar información desde distintas perspectivas (Janssen et al., 2024; Nambiar & Mundra, 2022).
- **Nivel Superior (Top Tier):** Es la capa de cliente, que incluye las herramientas de presentación de informes, minería de datos e inteligencia de negocios (BI) interactiva (Mezzoudj et al., 2025; Nambiar & Mundra, 2022).

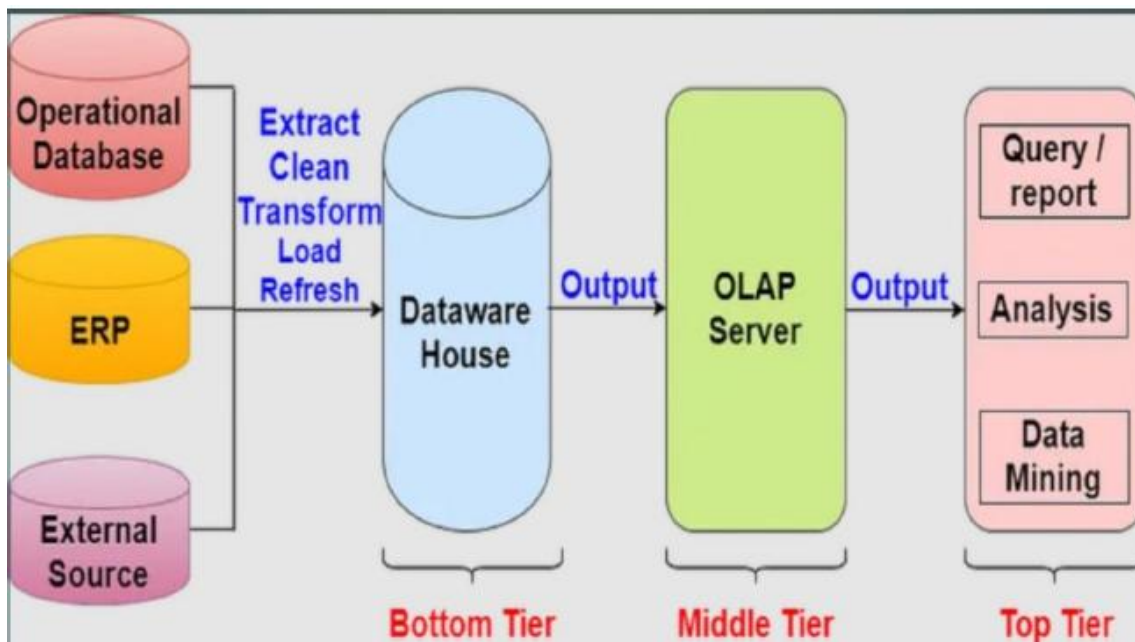


Figura 1. Arquitectura clásica de tres niveles de un almacén de datos.

Adaptado de Mezzoudj et al. (2025). ETL: Extract, Transform, Load; ERP: Enterprise Resource Planning; OLAP: Online Analytical Processing; OLTP: Online Transaction Processing.

Con la llegada del Big Data, sin embargo, esta arquitectura empezó a mostrar tres limitaciones estructurales que condujeron a su crisis operativa.

La primera fue su inflexibilidad ante la variedad de datos. Los Data Warehouses fueron concebidos para datos estructurados (Thomas et al., 2025). Al exigir un esquema previo a la escritura, carecen de la flexibilidad para almacenar información semiestructurada o no estructurada, como datos espaciales, fotografías, video, audio o texto plano (Bentaib et al., 2024; Janssen et al., 2024). Intentar adaptar esos formatos a esquemas relacionales requería un esfuerzo de ingeniería continuo y considerable (Armbrust et al., 2021).

A esto se sumaron límites de escalabilidad y sobrecostos extremos. Al acoplar el cómputo con el almacenamiento, las empresas se veían forzadas a escalar verticalmente adquiriendo hardware propietario costoso para absorber los picos de carga (Armbrust et al., 2021). Los datos son reveladores: las arquitecturas convencionales de Data Warehouse experimentaban una degradación del rendimiento de hasta un 42,7% cuando los volúmenes superaban los 7 petabytes (Chippada et al., 2025). Ese escalamiento ineficiente provocó que el 76,8% de las empresas excediera sus presupuestos de almacenamiento en un 312% durante los primeros 36 meses de operación (Chippada et al., 2025).

Finalmente, estas plataformas presentaban limitaciones críticas para la analítica avanzada. Al ser sistemas cerrados, no estaban diseñadas para procesar flujos de datos en tiempo real (streaming) ni para soportar de forma nativa las bibliotecas modernas de aprendizaje automático (Machine Learning). Los algoritmos de inteligencia artificial requieren procesar grandes volúmenes de datos con código no-SQL; extraerlos a través de conectores estándar (ODBC/JDBC) resultaba ineficiente y aislaba a los científicos de datos de los repositorios de negocio (Armbrust et al., 2021).

5.1.3 SEGUNDA GENERACIÓN: EL DATA LAKE (DL) Y EL SURGIMIENTO DE LOS PANTANOS DE DATOS

Frente a las restricciones económicas, la escalabilidad limitada y la rigidez del Data Warehouse, la industria introdujo una segunda generación de plataformas: el Data Lake (Lago de Datos). El término fue acuñado en 2010 por James Dixon, director de tecnología de

Pentaho, para describir un repositorio centralizado capaz de ingerir y almacenar volúmenes prácticamente ilimitados de datos en bruto para su uso futuro (Bentaib et al., 2024).

A diferencia del Data Warehouse, que imponía una estructura desde el principio, el Data Lake adoptó el enfoque opuesto: el esquema en la lectura (schema-on-read) (Bentaib et al., 2024). Este paradigma permite ingerir información en su formato original, ya sea estructurada, semiestructurada (JSON, XML) o no estructurada (texto, imágenes, video, telemetría IoT) sin necesidad de modelarla previamente mediante procesos ETL (Azzabi et al., 2024). La estructuración se difiere al momento en que el analista o científico de datos extrae la información para responder a una consulta específica, lo que otorga una flexibilidad sin precedentes para cargas de trabajo de Machine Learning y minería de datos (Mezzoudj et al., 2025).

Para entender las limitaciones que llevarían al colapso de este modelo, conviene revisar cómo evolucionó su arquitectura interna. La literatura documenta tres etapas bien diferenciadas:

- **Arquitectura Mono-Zona:** Las primeras implementaciones de Data Lakes adoptaban un diseño plano con una única zona de almacenamiento, generalmente construida sobre el ecosistema Apache Hadoop (HDFS) (Azzabi et al., 2024). Todos los datos en crudo se volcaban en ese repositorio central. La simplicidad, sin embargo, fue su talón de Aquiles: sin estructura de directorios definida ni registro de actividades de usuario, la información pronto se volvió inmanejable (Azzabi et al., 2024).

- Arquitecturas de Procesamiento (Lambda y Kappa):** Para atender tanto datos históricos como flujos en tiempo real, surgieron diseños como la arquitectura Lambda, que separaba el procesamiento en una capa por lotes (*batch layer*) y una capa de velocidad (*speed layer*) (Azzabi et al., 2024). El problema fue que este modelo obligaba a los ingenieros a codificar y mantener la misma lógica dos veces, en sistemas separados, lo que elevaba considerablemente la complejidad y los costos operativos (Azzabi et al., 2024). Como respuesta a este inconveniente, se introdujo la arquitectura Kappa, una versión minimalista que elimina la capa por lotes y utiliza una única canalización unificada basada exclusivamente en la capa de velocidad, evitando así la necesidad de reescribir y duplicar el código (Azzabi et al., 2024). Sin embargo, la arquitectura Kappa también demostró tener limitaciones severas, ya que ejecutar tareas analíticas complejas (como el entrenamiento de modelos de Machine Learning o uniones de conjuntos masivos de datos) utilizando únicamente procesamiento de flujos (*streaming*) resulta extremadamente difícil (Schneider et al., 2024).
- Arquitectura Multi-Zona o Funcional:** Como respuesta a la desorganización, la industria maduró hacia una arquitectura estructurada en zonas lógicas o capas.

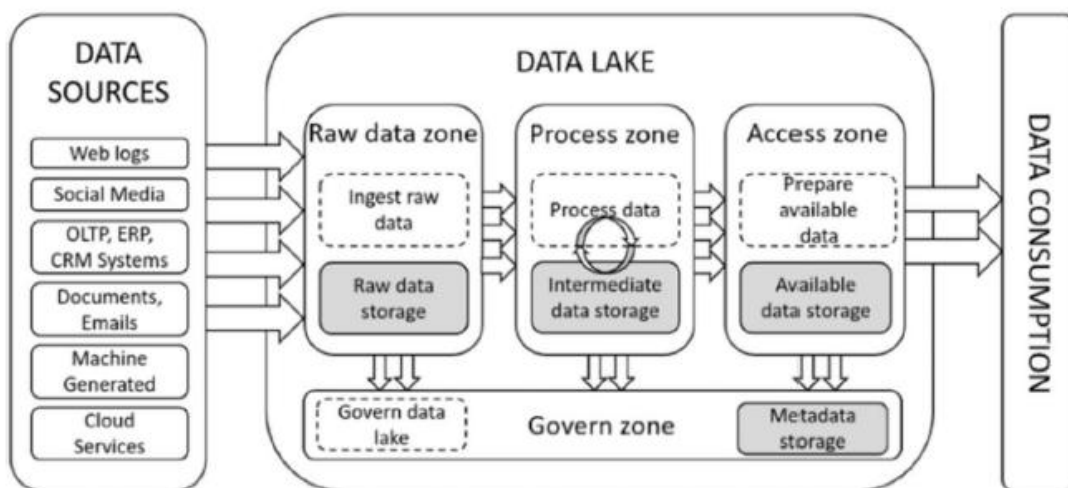


Figura 2. Arquitectura multizona de un Data Lake.

Adaptado de Mezzoudj et al. (2025). CRM: Customer Relationship Management; ERP: Enterprise Resource Planning; OLTP: Online Transaction Processing.

Como muestra la figura anterior, el estándar moderno del Data Lake se organiza en cuatro componentes (Mezzoudj et al., 2025):

- **Zona de Datos Crudos (Raw Data Zone):** Almacena los datos en su formato original exacto, funcionando como un registro histórico inmutable (Mezzoudj et al., 2025).
- **Zona de Procesamiento (Process Zone):** Es donde los datos crudos se transforman e integran para ser limpiados y preparados (Mezzoudj et al., 2025).
- **Zona de Acceso (Access Zone):** Contiene los datos ya procesados y disponibles para herramientas de BI y modelos avanzados de IA (Mezzoudj et al., 2025).
- **Zona de Gobernanza (Govern Zone):** Una capa conceptual que gestiona la seguridad, el ciclo de vida del dato y los metadatos (Mezzoudj et al., 2025).

A pesar de estos esfuerzos por organizar el Data Lake mediante zonas lógicas, la tecnología subyacente presentaba deficiencias estructurales que derivaron en tres crisis operativas. La más conocida fue el llamado "Pantano de Datos" (Data Swamp). En la práctica, una gestión rudimentaria de los metadatos y una gobernanza débil provocaron que la información se acumulara caóticamente. Los repositorios se degradaban en pantanos donde

los datos perdían contexto, se volvían inutilizables y dejaban de ser confiables para la toma de decisiones (Azzabi et al., 2024; Mezzoudj et al., 2025).

A ello se sumó la carencia de transacciones ACID. A diferencia de un Sistema de Gestión de Bases de Datos (DBMS), el Data Lake no ofrece garantías de consistencia, aislamiento ni atomicidad (Schneider et al., 2024). Cuando múltiples usuarios o procesos acceden simultáneamente para leer, actualizar o borrar datos, el sistema no puede aislar esas transacciones, lo que genera anomalías, lecturas sucias y corrupción de archivos (Schneider et al., 2024; Armbrust et al., 2021).

Finalmente, el Data Lake resultó ineficiente para la inteligencia de negocios (BI). Aunque el almacenamiento plano sin esquemas es adecuado para la ciencia de datos exploratoria, las consultas SQL complejas pueden tardar horas en ejecutarse. La imposibilidad de indexar correctamente la información o de aplicar optimizaciones propias de un motor de base de datos impide que el Data Lake cumpla las funciones de reporte rápido que el Data Warehouse sí ofrecía (Mezzoudj et al., 2025).

5.1.4 LA CRISIS DE LAS ARQUITECTURAS HÍBRIDAS (ARQUITECTURAS DE DOS NIVELES)

Dado que ni el Data Warehouse, restringido a datos estructurados y de alto costo, ni el Data Lake, carente de garantías ACID y de rendimiento para consultas rápidas, podían resolver por sí solos todas las demandas analíticas empresariales, la práctica estándar de la industria convergió hacia un patrón conocido como arquitectura de dos niveles o Two-Tier Architecture (Armbrust et al., 2021).

En este modelo híbrido, todos los datos en bruto de los sistemas operativos se ingresan primero al Data Lake, donde quedan disponibles para los científicos de datos. Luego, mediante complejas canalizaciones ETL, un subconjunto seleccionado se transfiere al Data Warehouse para satisfacer las cargas de trabajo OLAP y los reportes de inteligencia de negocios (Schneider et al., 2024). Los datos que dejan de ser necesarios en el almacén se devuelven al lago, buscando reducir costos de almacenamiento (Schneider et al., 2024).

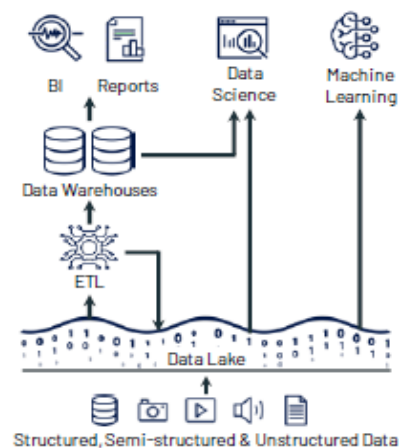


Figura 3. Arquitectura de gestión de datos de dos niveles.

Adaptado de Armbrust et al. (2021). BI: Business Intelligence; ETL: Extract, Transform, Load; ML: Machine Learning.

Como expone la figura anterior, mantener plataformas paralelas parecía una solución razonable, pero en la práctica resultó insostenible. La literatura especializada identifica cuatro crisis operativas derivadas de este enfoque.

La primera fue una grave complejidad y pérdida de fiabilidad. Mantener la consistencia de los datos entre el lago y el almacén exige un esfuerzo de ingeniería continuo y minucioso (Armbrust et al., 2021). Cada paso de movimiento o transformación de datos incorpora nuevos puntos de falla e introduce errores sutiles que deterioran la calidad de la información (Schneider et al., 2024). De forma relacionada, surgió el problema de la obsolescencia de los datos (Data Staleness): las canalizaciones ETL generan cuellos de botella que retrasan la disponibilidad de la información, dejando el Data Warehouse constantemente desactualizado frente al Data Lake (Armbrust et al., 2021). Estudios sectoriales revelan que el 86 % de los analistas termina utilizando datos desactualizados en sus reportes, lo que imposibilita la toma de decisiones en tiempo real (Armbrust et al., 2021).

Por otro lado, la fragmentación del ecosistema en silos aislados genera una pérdida masiva de productividad y una barrera para la IA. Los profesionales de datos destinan hasta el 82,7% de sus horas de trabajo a localizar, extraer, limpiar y sincronizar datos entre sistemas, en lugar de generar conocimiento útil (Chippada et al., 2025). Adicionalmente, si los equipos de ciencia de datos necesitaban utilizar la información estructurada que residía en el almacén, se veían obligados a exportar los datos nuevamente hacia archivos en el lago (añadiendo un tercer paso ETL), lo que multiplicaba la complejidad arquitectónica e incrementaba la latencia de la información.

Finalmente, este esquema multiplica el Costo Total de Propiedad (TCO). Las organizaciones pagan doble por almacenar los mismos datos redundantemente en ambos sistemas, y deben sostener una infraestructura dual con equipos de administración paralelos (Armbrust et al., 2021). A esto se añade el problema del bloqueo de proveedor (vendor lock-in) que imponen los formatos propietarios del Data Warehouse comercial, encareciendo cualquier migración futura (Armbrust et al., 2021).

Fue precisamente esta confluencia de ineficiencias (latencia de información, sobrecostos y complejidad creciente) la que motivó a investigadores y líderes de la industria a diseñar un modelo superador. El objetivo era claro: combinar el rendimiento analítico del Data Warehouse con la flexibilidad y bajo costo del Data Lake en una única plataforma abierta. Así nació la tercera generación de sistemas de gestión de datos: la arquitectura Data Lakehouse.

5.2 FUNDAMENTOS DE LA ARQUITECTURA DATA LAKEHOUSE

5.2.1 DEFINICIÓN Y LOS CINCO DESAFÍOS TÉCNICOS (C1-C5)

El Data Lakehouse surge como un paradigma arquitectónico orientado a integrar las fortalezas analíticas del Data Warehouse con la flexibilidad de esquemas del Data Lake (Cherradi y El Haddadi, 2024). Formalmente, se define como una plataforma de gestión de datos de próxima generación basada en almacenamiento de acceso directo y bajo costo en formatos abiertos, pero que ofrece las características de rendimiento, gestión y calidad propias de los sistemas de bases de datos relacionales, unificando la inteligencia de negocios con la analítica avanzada en un único entorno (Armbrust et al., 2021). No se trata de una base de datos convencional: es una arquitectura capaz de procesar eficientemente cargas de trabajo multimodales de forma simultánea (Azzabi et al., 2024).

El Lakehouse no es una evolución teórica, sino una respuesta de ingeniería a los fallos concretos de las generaciones anteriores (Thomas et al., 2025). La investigación académica

identifica cinco desafíos técnicos críticos (C1 a C5) que los Data Lakes y las arquitecturas híbridas no lograron resolver, y que el Lakehouse se propone eliminar (Schneider et al., 2024):

- **C1 – Arquitectura Compleja (Complex Architecture):** Los ecosistemas híbridos dependían de múltiples sistemas de almacenamiento y motores de procesamiento desconectados entre sí. Esto obligaba a las empresas a realizar movimientos y transformaciones de datos constantes entre plataformas, generando altos costos operativos y una sobrecarga de mantenimiento considerable para los ingenieros (Schneider et al., 2024).
- **C2 – Combinación ineficiente de lotes y flujos (Combining Batch and Stream Processing):** Las arquitecturas previas requerían bases de código separadas y lógicas duplicadas para gestionar tanto los datos históricos (por lotes) como los flujos en tiempo real (streaming), complicando y encareciendo la operación analítica (Schneider et al., 2024).
- **C3 – Imposibilidad de procesamiento incremental (Incremental Processing):** Los motores de procesamiento trataban los archivos de datos como entidades inmutables. Sin capacidad para actualizar o eliminar registros individuales sin reescribir archivos completos, el corregir errores, purgar datos o evolucionar esquemas resultaba extremadamente costoso (Schneider et al., 2024).
- **C4 – Ausencia de cumplimiento de consistencia (Consistency Enforcement):** Los repositorios de archivos planos carecían de mecanismos para validar la estructura de los datos almacenados. Sin garantías automáticas sobre los esquemas, los

desarrolladores debían escribir cantidades masivas de código de validación manual en cada etapa del proceso (Schneider et al., 2024).

- C5 – Carencia de atomicidad y aislamiento (Assurance of Atomicity and Isolation):** En entornos donde múltiples procesos leen y escriben datos de forma concurrente, la ausencia de propiedades transaccionales ACID nativas podía generar anomalías de concurrencia, inconsistencias masivas e información parcial o corrupta que comprometía las decisiones de negocio (Schneider et al., 2024).

La propuesta central del Lakehouse es resolver estos cinco desafíos mediante un marco lógico que aplica capas de transacciones y metadatos directamente sobre el almacenamiento en bruto, sin necesidad de mover la información a sistemas separados (Armbrust et al., 2021).

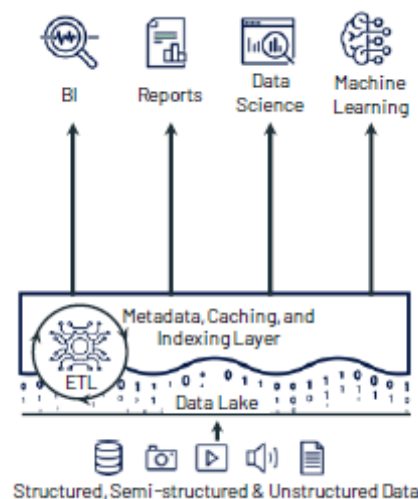


Figura 4. Plataforma unificada de la arquitectura Data Lakehouse.

Adaptado de Armbrust et al. (2021). BI: Business Intelligence; ETL: Extract, Transform, Load; ML: Machine Learning.

Como ilustra el diagrama anterior, el Lakehouse añade una capa unificada de metadatos, indexación y almacenamiento en caché directamente sobre el lago de datos. Esta estructura permite que todas las cargas de trabajo operen de manera simultánea y segura sobre los mismos archivos (Armbrust et al., 2021).

5.2.2 LOS OCHO REQUISITOS TÉCNICOS FUNDAMENTALES DEL LAKEHOUSE

Para superar los cinco desafíos operativos (C1 a C5), la implementación de un Data Lakehouse debe regirse por principios arquitectónicos que orienten su diseño antes de traducirse en requisitos técnicos concretos. De acuerdo con Menon (2022), estos principios establecen que el sistema debe ser disciplinado en su núcleo de almacenamiento pero flexible en sus capas de procesamiento; debe desacoplar el cómputo del almacenamiento para que ambos escalen de forma independiente; debe priorizar la funcionalidad de cada componente por encima de la dependencia hacia un producto específico; debe construirse de forma modular para incorporar nuevas capacidades sin interrupciones; y debe ejecutar una catalogación activa y continua que prevenga la degradación del repositorio en un pantano de datos inmanejable. Estos principios no son requisitos verificables en sí mismos, sino la filosofía de diseño que da coherencia al conjunto. A partir de ellos, la comunidad científica ha formalizado ocho requisitos técnicos estrictos (R1 a R8) que un sistema debe cumplir para ser considerado un genuino Lakehouse (Schneider et al., 2024):

- **(R1) Mismo tipo de almacenamiento y formato de datos:** Todos los datos y metadatos deben residir en un único tipo de almacenamiento altamente escalable,

utilizando un formato de archivo abierto y estandarizado como Apache Parquet (Schneider et al., 2023; Schneider et al., 2024).

- **(R2) Soporte CRUD para todos los tipos de datos:** La plataforma debe soportar operaciones de Creación, Lectura, Actualización y Eliminación (CRUD) a nivel de registro individual para cualquier variedad de información (Schneider et al., 2024; Schneider et al., 2023).
- **(R3) Colecciones de datos relacionales:** El sistema debe proveer conceptos lógicos que permitan agrupar datos estructurados, físicamente dispersos, dentro de un contexto relacional cohesivo en forma de tablas (Schneider et al., 2023; Schneider et al., 2024).
- **(R4) Lenguaje de consulta:** La arquitectura debe ofrecer una interfaz declarativa estándar, típicamente SQL, para acceder y manipular la información, democratizando su uso (Armbrust et al., 2021).
- **(R5) Garantías de consistencia:** El sistema debe incorporar mecanismos integrados para verificar y hacer cumplir la consistencia estructural de los datos, ya sea en el momento de su escritura o de su lectura (Schneider et al., 2024; Schneider et al., 2023).
- **(R6) Aislamiento y atomicidad:** El Lakehouse debe garantizar propiedades transaccionales ACID sobre las colecciones de datos, asegurando que las

operaciones concurrentes no interfieran entre sí (Schneider et al., 2024; Schneider et al., 2023).

- **(R7) Acceso directo de lectura:** Los motores de procesamiento avanzado deben poder leer eficientemente los archivos directamente desde el almacenamiento subyacente, sin pasar por interfaces de bases de datos propietarias (Schneider et al., 2024; Schneider et al., 2023).
- **(R8) Procesamiento unificado de Batch y Streaming:** La plataforma debe permitir que las mismas colecciones de datos actúen simultáneamente como fuentes y destinos para cargas de trabajo históricas por lotes y flujos en tiempo real (Schneider et al., 2024).

3.2.3 CAPAS ARQUITECTÓNICAS DEL MODELO LAKEHOUSE

La arquitectura Data Lakehouse puede comprenderse desde dos perspectivas complementarias: una visión lógica empresarial que abarca el ciclo de vida completo del dato, y una visión técnica que describe el motor interno responsable de cumplir los ocho requisitos fundamentales. Desde la perspectiva empresarial, el ecosistema se organiza en siete capas lógicas interconectadas que estructuran cómo fluye la información desde su origen hasta su consumo analítico (Menon, 2022):

- **Capa de Ingesta de Datos (Data Ingestion Layer):** Es el punto de entrada para los distintos proveedores de datos (sistemas OLTP, archivos de texto, flujos en tiempo real, dispositivos IoT o datos multimedia) (Menon, 2022). Funciona mediante servicios

de ingesta periódica por lotes (batch) y servicios de ingesta continua en tiempo real (streaming) (Menon, 2022).

- **Capa de Lago de Datos (Data Lake Layer):** Corresponde a la zona de almacenamiento físico. Se subdivide en repositorios de datos crudos (raw) que preservan el formato original, datos intermedios para acelerar el procesamiento, datos procesados listos para el consumo, y datos archivados a largo plazo para retención de bajo costo (Menon, 2022).
- **Capa de Procesamiento de Datos (Data Processing Layer):** Transforma la información ingerida a través de motores de computación distribuida, gestionando tanto los procesos por lotes como los flujos ininterrumpidos en tiempo real (Menon, 2022).
- **Capa de Servicio de Datos (Data Serving Layer):** Actúa como puente hacia las aplicaciones y consumidores aguas abajo. Entrega los datos procesados a través de almacenes relacionales, sistemas NoSQL mediante APIs, servicios en tiempo real y plataformas de intercambio estructurado (Data Share) (Menon, 2022).
- **Capa de Analítica de Datos (Data Analytics Layer):** Es el entorno donde se genera el conocimiento. Incluye espacios de experimentación (sandboxes) para analistas y científicos de datos, servicios de Inteligencia Artificial y Aprendizaje Automático (AI/ML), y servicios de Inteligencia de Negocios (BI) para la elaboración de reportes (Menon, 2022).

- **Capa de Gobernanza de Datos (Data Governance Layer):** Capa transversal que previene la degradación del sistema en un pantano de datos inmanejable. Se ocupa de la gestión de políticas, la catalogación activa con linaje de los activos de datos y el control de calidad, exactitud y completitud de la información (Menon, 2022).
- **Capa de Seguridad de Datos (Data Security Layer):** Entorno transversal que protege toda la plataforma a través de la Gestión de Identidad y Acceso (IAM), servicios de encriptación de datos en reposo y en tránsito, enmascaramiento dinámico de datos sensibles y seguridad estricta de red (Menon, 2022).

Dado que la Capa de Seguridad de Datos actúa de forma transversal sobre todas las demás, vale detallar su estructura. Menon (2022) la describe como cuatro componentes interconectados: la Gestión de Identidad y Acceso (IAM), que aplica el Principio de Privilegio Mínimo bajo una arquitectura de "Confianza Cero" (Zero-Trust) para autenticar cada solicitud; la Encriptación de Datos, que protege la información tanto en reposo como en tránsito; el Enmascaramiento de Datos (Data Masking), que oculta la Información de Identificación Personal (PII) permitiendo que los analistas trabajen con los datos sin exponer los valores reales; y la Seguridad de Red, implementada mediante Redes Virtuales (VNET), cortafuegos y puertas de enlace VPN que protegen el perímetro del ecosistema y sus comunicaciones con fuentes externas o infraestructuras on-premises.

Una vez establecida la visión empresarial, es posible descender al motor técnico que hace viable esta arquitectura. Internamente, el Lakehouse desacopla el almacenamiento del procesamiento mediante tres capas principales, permitiendo que ambos recursos escalen de forma independiente para optimizar costos y eficiencia (Janssen et al., 2024; Armbrust et al., 2021):

- **Capa de Almacenamiento (Storage Layer):** Utiliza sistemas de almacenamiento de objetos en la nube de bajo costo, alta durabilidad y elasticidad (Thomas et al., 2025). Toda la información estructurada, semiestructurada o no estructurada se guarda en formatos de archivo columnares abiertos y estandarizados como Apache Parquet o Apache ORC, que incorporan mecanismos avanzados de compresión y codificación para optimizar el escaneo de datos y reducir los costos de retención a escala de petabytes (Armbrust et al., 2021; Thomas et al., 2025).
- **Capa Transaccional y de Metadatos (Metadata and Transaction Layer):** Es la innovación central que distingue al Lakehouse de un simple lago de datos (Armbrust et al., 2021). Esta capa lógica reside directamente sobre los archivos en bruto y mantiene un registro de transacciones inmutable que rastrea qué archivos físicos pertenecen a cada versión de la tabla (Thomas et al., 2025). Gracias a ello, el sistema habilita transacciones ACID, versionado de datos y aplicación de esquemas estrictos sin necesidad de mover la información de su almacenamiento original (Armbrust et al., 2021; Thomas et al., 2025).
- **Capa de Procesamiento y Consumo (Processing and Consumption Layer):** Dado que el almacenamiento y los metadatos están desacoplados y basados en estándares abiertos, múltiples motores de procesamiento distribuido pueden conectarse simultáneamente a la misma plataforma (Thomas et al., 2025). Tecnologías como Apache Spark, Trino o Photon ejecutan cargas de trabajo de Inteligencia de Negocios mediante SQL, mientras que las bibliotecas de aprendizaje automático leen directamente los archivos Parquet a través de interfaces declarativas como los DataFrames (Armbrust et al., 2021; Thomas et al., 2025).

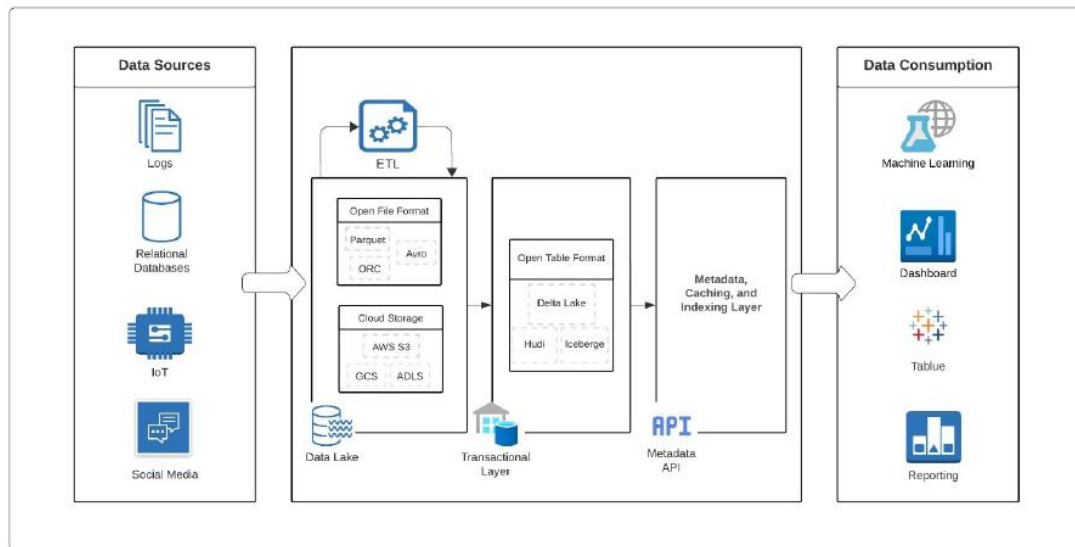


Figura 5. Arquitectura basada en capas de un Data Lakehouse.

Adaptado de Azzabi et al. (2024). ADLS: Azure Data Lake Storage; API: Application Programming Interface; AWS S3: Amazon Web Services Simple Storage Service; ETL: Extract, Transform, Load; GCS: Google Cloud Storage.

La integración de estas dos perspectivas elimina la necesidad de infraestructuras paralelas o canalizaciones redundantes. La capa transaccional actúa como puente entre los archivos en bruto y los motores de consumo, garantizando que tanto las herramientas de BI como los algoritmos de ciencia de datos operen sobre una única fuente de verdad en tiempo real (Armbrust et al., 2021; Thomas et al., 2025).

5.2.4 CARACTERÍSTICAS AVANZADAS DE GESTIÓN Y EVOLUCIÓN DE ESQUEMAS

Más allá de unificar el almacenamiento y el procesamiento, el Lakehouse introduce capacidades críticas para el ciclo de vida del dato, entre ellas la evolución de esquemas y el

linaje inmutable (Thomas et al., 2025). A través de la capa de metadatos transaccionales, plataformas como Delta Lake, Apache Iceberg y Apache Hudi permiten aplicar esquemas estrictos para mantener la calidad de la información y, al mismo tiempo, facilitan la evolución dinámica de dichos esquemas conforme cambian las necesidades del negocio, previniendo que el ecosistema degenere en un pantano de datos inmanejable (Gade, 2022). Esta capa también mantiene un registro de transacciones y capturas de estado que proporciona un linaje de datos en tiempo de ejecución, fundamental para documentar la trazabilidad desde la ingesta hasta el consumo analítico final (Thomas et al., 2025). Ese linaje automatizado no solo posibilita la auditabilidad continua del sistema, sino que establece una base sólida para el análisis de impacto y para garantizar la confianza en los modelos analíticos o de aprendizaje automático generados por la organización (Thomas et al., 2025).

Otra de las capacidades más relevantes es el versionado de datos, conocido en la industria como "viaje en el tiempo" o Time Travel (Gade, 2022). Esta funcionalidad permite acceder a versiones históricas anteriores de los datos almacenados, lo que resulta especialmente valioso en entornos colaborativos donde múltiples equipos necesitan trabajar con capturas de información precisas para auditorías, investigaciones o cumplimiento normativo (Gade, 2022). Al anclar las consultas a estados inmutables de las tablas en un punto específico del tiempo, el viaje en el tiempo garantiza la reproducibilidad analítica y previene la pérdida de información ante sobreescrituras accidentales o corrupciones en los procesos de ingesta (Gade, 2022; Thomas et al., 2025). En consecuencia, los científicos de datos pueden auditar los modelos de inteligencia artificial utilizando exactamente las mismas entradas con las que fueron entrenados originalmente, asegurando la reproducibilidad técnica de los experimentos (Gade, 2022).

Todas estas capacidades convergen además con los principios FAIR, que establecen que los datos deben ser Encontrables, Accesibles, Interoperables y Reutilizables (Findable,

Accessible, Interoperable, and Reusable) tanto para humanos como para sistemas automatizados (Begoli et al., 2021). El Lakehouse facilita la adopción nativa de estos principios al incorporar infraestructuras lógicas que permiten recuperar metadatos mediante protocolos estandarizados y rastrear la procedencia exacta de la información mediante ontologías de linaje, incluso cuando los datos físicos originales han sido purgados (Begoli et al., 2021). La implementación de esquemas en formatos abiertos y la asociación de metadatos detallados durante cada etapa del procesamiento garantizan que los usuarios puedan navegar, comprender y reutilizar la información para investigación a gran escala sin intervención manual (Begoli et al., 2021). Con ese nivel de gobernanza, catalogación y transparencia estructural, el Lakehouse se consolida como una plataforma unificada que cumple las exigencias de seguridad y cumplimiento normativo de los ecosistemas empresariales modernos (Gade, 2022).

5.3 TECNOLOGÍAS HABILITADORAS Y FORMATOS DE TABLA

La viabilidad técnica del Data Lakehouse descansa en los formatos de tabla abiertos, que funcionan como una capa de gestión intermedia sobre el almacenamiento en la nube (Armbrust et al., 2021; Thomas et al., 2025). Esta capa transaccional es la responsable de proporcionar garantías ACID sobre sistemas de almacenamiento de objetos que, en su estado natural, solo ofrecen consistencia eventual, resolviendo así problemas históricos como la corrupción por escrituras fallidas (Armbrust et al., 2020). El ecosistema actual está dominado por tres tecnologías que operan bajo un control de concurrencia multiversión para gestionar transacciones y aplicar esquemas: Delta Lake, Apache Iceberg y Apache Hudi (Jain et al., 2023; Parimi, 2025). Estas herramientas ofrecen una abstracción uniforme que aísla a ingenieros y analistas de la complejidad física del sistema de archivos subyacente (Thomas et al., 2025).

5.3.1 DELTA LAKE

Desarrollado originalmente para optimizar canalizaciones de datos estructurados, Delta Lake se ha consolidado como una implementación de referencia para la arquitectura Lakehouse (Schneider et al., 2024). Su arquitectura interna se basa en un registro de transacciones de solo adición (transaction log) alojado junto con los archivos de datos en formato Apache Parquet (Bentaib et al., 2024; Parimi, 2025). Este registro almacena una secuencia de objetos JSON que detallan cada operación atómica, combinados con archivos de puntos de control (checkpoints) periódicos para limitar el costo de planificación de consultas y evitar la lectura iterativa del historial completo (Armbrust et al., 2020). El cumplimiento transaccional ACID se asegura mediante un control de concurrencia optimista, apoyado en las operaciones atómicas nativas de los proveedores de nube o en servicios de coordinación ligeros (Armbrust et al., 2020). Las evaluaciones empíricas destacan su "simplicidad operativa" y su integración nativa con Apache Spark, lo que lo hace especialmente atractivo para organizaciones que priorizan la facilidad de mantenimiento sobre la escalabilidad extrema de escrituras (Parimi, 2025). Delta Lake también incluye capacidades de optimización del almacenamiento físico, como el ordenamiento Z (Z-ordering), un algoritmo que reorganiza los registros a lo largo de múltiples dimensiones para maximizar el salto de archivos irrelevantes (data skipping) en consultas altamente selectivas (Armbrust et al., 2020). Sin embargo, bajo condiciones de escritura masiva concurrente, el registro de transacciones centralizado puede convertirse en un cuello de botella por contención de red (Parimi, 2025).

5.3.2 APACHE ICEBERG

A diferencia del enfoque basado en listar directorios o en un único registro central, Apache Iceberg es un formato de tabla abierto diseñado desde cero para gestionar conjuntos masivos de datos analíticos a escala de petabytes (Bentaib et al., 2024; Parimi, 2025). Su principal innovación radica en una estructura jerárquica compuesta por archivos de metadatos, listas de manifiestos y archivos de manifiesto individuales que rastrean la ubicación física y las estadísticas operativas de cada archivo de datos (Parimi, 2025). Este diseño desacoplado permite a los motores de consulta identificar archivos específicos con precisión, eliminando la latencia que genera la inspección de directorios completos en el almacenamiento en la nube (Jain et al., 2023). Su enfoque basado en instantáneas (*snapshots*) ofrece ventajas significativas en portabilidad entre múltiples motores y facilita ecosistemas descentralizados que adoptan el paradigma *Data Mesh* (Mishra, 2022). Las pruebas de rendimiento posicionan a Iceberg como el líder para cargas de trabajo con alta proporción de lecturas, gracias a su eficiencia en la planificación de consultas y la poda jerárquica de particiones (Kasireddy, 2023; Parimi, 2025). Además, permite la evolución completa de esquemas y particiones mediante operaciones que solo alteran los metadatos lógicos, sin requerir la reescritura física de los archivos (Mishra, 2022; Thomas et al., 2025). Su principal limitación aparece en escenarios de concurrencia de escritura excepcionalmente alta, donde la coordinación de las listas de manifiestos sufre contención y exhibe degradación exponencial en la latencia de escritura (Parimi, 2025).

5.3.3 APACHE HUDI

Apache Hudi (Hadoop Upserts Deleted Incrementals) fue diseñado específicamente para acelerar el procesamiento de cargas incrementales y optimizar la ingestión de flujos de datos continuos sobre sistemas de almacenamiento de objetos (Bentaib et al., 2024). Su arquitectura captura cambios de datos con muy baja latencia mediante un servicio de línea

de tiempo (timeline service) que registra con versionado instantáneo todas las acciones y estados realizados sobre la tabla (Parimi, 2025; Thomas et al., 2025). Para lograrlo en entornos altamente dinámicos, Hudi implementa dos estrategias de actualización: Copia en Escritura (Copy-On-Write o CoW) y Fusión en Lectura (Merge-On-Read o MoR) (Jain et al., 2023; Kasireddy, 2023). En el modo CoW, el sistema reescribe los archivos completos en cada actualización, lo que genera alta amplificación de escritura pero mantiene un rendimiento de lectura analítica veloz (Jain et al., 2023). El modo MoR, en cambio, registra los cambios en archivos log auxiliares y pospone la reconciliación al momento de la consulta, logrando la latencia de escritura más baja de las tres opciones para la ingestión continua de flujos (Kasireddy, 2023; Parimi, 2025). Las evaluaciones operativas confirman que Hudi maneja mejor que sus competidores la alta concurrencia de escritores gracias a su eficiente detección de conflictos a nivel de archivo, aunque exige madurez técnica considerable para afinar continuamente los trabajos de compactación asíncrona entre los registros en flujo y los datos base (Kasireddy, 2023; Parimi, 2025).

5.3.4 ANÁLISIS COMPARATIVO Y DE RENDIMIENTO

La evaluación conjunta de estas tres tecnologías deja claro que no existe un formato universalmente superior. Cada una refleja decisiones arquitectónicas profundas en la gestión de metadatos que favorecen distintos tipos de cargas de trabajo (Kasireddy, 2023; Parimi, 2025). Los estudios empíricos con simulaciones estandarizadas como TPC-DS revelan que Iceberg sobresale en análisis masivos de lectura cruzando múltiples motores independientes; Delta Lake ofrece el mayor rendimiento en escritura masiva con una simplicidad operativa sin igual en ecosistemas Spark; y Hudi domina en escenarios intensivos de actualizaciones y captura de datos de cambio (CDC) continuos (Kasireddy, 2023; Parimi, 2025). Bajo métricas de estrés transaccional, Iceberg y Delta Lake muestran degradación de latencia por cuellos

de botella en la coordinación de manifiestos y el registro centralizado, respectivamente, mientras que Hudi exhibe la mayor estabilidad y la menor tasa de conflictos bajo concurrencia extrema de escritura (Parimi, 2025). Más allá de estas diferencias, el estado del arte refleja una tendencia acelerada hacia la convergencia de características y la interoperabilidad plena, lo que reduce significativamente los riesgos de dependencia de un solo proveedor y permite a las empresas estructurar arquitecturas Lakehouse sólidas, gobernadas y rentables (Kasireddy, 2023; Parimi, 2025).

5.4 ANÁLISIS COMPARATIVO: GOBERNANZA, ESCALABILIDAD, RENDIMIENTO Y COSTOS

A diferencia de las arquitecturas tradicionales, que segregan las cargas de trabajo en sistemas aislados, el Data Lakehouse integra las fortalezas del Data Warehouse y el Data Lake en una única plataforma unificada (Armbrust et al., 2021). Al evaluarlo bajo los criterios de gobernanza, escalabilidad, rendimiento operativo y soporte algorítmico, se ponen de manifiesto ventajas estructurales y financieras que explican su creciente adopción empresarial (Janssen et al., 2024).

5.4.1 GOBERNANZA, EVOLUCIÓN DE ESQUEMAS Y CONTROL DE CALIDAD

Una de las críticas más frecuentes a los lagos de datos tradicionales es la ausencia de gestión de metadatos y garantías transaccionales, lo que los convierte con facilidad en "pantanos de datos" con información poco fiable (Janssen et al., 2024). El Lakehouse aborda esta debilidad incorporando una capa robusta de gestión de metadatos que habilita transacciones ACID, versionado de datos y cumplimiento normativo (Janssen et al., 2024).

Sin embargo, la gobernanza en un ecosistema Lakehouse no depende únicamente de la tecnología subyacente, sino de una estructuración organizativa holística. Para implementarla de manera efectiva, la literatura propone el "marco 3-3-3 de gobernanza de datos", compuesto por tres objetivos, tres pilares y tres componentes fundamentales (Menon, 2022):

- **Tres objetivos principales:** Disponibilidad de datos (garantizar que la plataforma entregue la información correcta en el momento oportuno), Visibilidad de datos (mantener un inventario intuitivo de los activos) y Calidad de datos (asegurar la exactitud, consistencia y completitud a lo largo del ciclo de vida) (Menon, 2022).
- **Tres pilares fundamentales:** Personas (con roles organizativos claros como el patrocinador ejecutivo, líderes de gobernanza, propietarios de datos, administradores de datos o Data Stewards y administradores de publicación), Políticas (reglas estandarizadas para el descubrimiento, uso y monitoreo de la información) y Tecnología (los servicios y herramientas en la nube para ejecutar dichas reglas) (Menon, 2022).
- **Tres componentes de la capa de gobernanza:** La gestión integral de políticas de datos, los servicios de calidad para prevenir anomalías, y los servicios de catalogación y curación activa (Menon, 2022). Este último organiza un inventario visual completo que consolida metadatos funcionales (glosarios, modelos lógicos) y metadatos técnicos (linaje de datos, propiedades de almacenamiento) de todo el Lakehouse (Menon, 2022).

La combinación de este marco organizativo con la gestión técnica de metadatos transaccionales es lo que permite al ecosistema adaptarse a las exigencias empresariales sin fracturarse. En entornos modernos, las organizaciones enfrentan un promedio de 412 cambios de esquema por mes, lo que afecta seriamente los procesos analíticos dependientes (Chippada et al., 2025). La capacidad del Lakehouse para gestionar esa evolución y mantener datos versionados reduce en un 92,1% las interrupciones en los procesos analíticos posteriores (Chippada et al., 2025). El versionado, anclado a las políticas de gobernanza, garantiza una continuidad del 99,9% en las cargas de trabajo, evitando que los cambios estructurales impacten la lógica de negocio y traduciéndose en ahorros considerables al prevenir la disrupción operativa (Chippada et al., 2025).

5.4.2 ESCALABILIDAD, FORMATOS ABIERTOS Y PREVENCIÓN DE LAS DEPENDENCIAS TECNOLÓGICAS

En términos de escalabilidad, el Lakehouse supera al Data Warehouse tradicional al desacoplar por completo la capacidad de almacenamiento de la capacidad de procesamiento (Janssen et al., 2024). Esa separación estructural permite escalar los recursos informáticos de forma elástica bajo demanda, adaptándose rentablemente a los picos de carga y a las necesidades cambiantes del negocio (Janssen et al., 2024). Además, el uso de formatos de tabla abiertos como Apache Parquet, Delta Lake, Iceberg o Hudi estandariza el acceso a la información y previene el bloqueo del proveedor (vendor lock-in) (Chippada et al., 2025). Las evaluaciones financieras muestran que las organizaciones que aprovechan estos formatos obtienen un ahorro promedio de 3,74 millones de dólares en costos de migración al cambiar de proveedor o plataforma en la nube, reduciendo los plazos de transición de 14,7 a tan solo 3,8 meses (Chippada et al., 2025). Esta flexibilidad arquitectónica es considerada un pilar

estratégico por el 73,6 % de las organizaciones multinacionales al ejecutar sus estrategias multinube (Chippada et al., 2025).

Para potenciar esta separación estructural, las implementaciones modernas incorporan marcos de escalado dinámico automatizado (autoscaling). Las organizaciones que implementan este enfoque logran reducir sus costos de infraestructura en un 57,3 % frente al aprovisionamiento estático tradicional, alcanzando tasas de utilización de recursos informáticos del 87,4 % en comparación con un 23,7 % en entornos estáticos (Chippada et al., 2025). Adicionalmente, al aprovechar la planificación y ejecución distribuida de consultas, el Lakehouse alcanza escalabilidad lineal para el 92,7 % de las cargas de trabajo analíticas, procesando conjuntos de datos a escala de petabytes sin degradación del rendimiento (Chippada et al., 2025).

En la frontera de la flexibilidad arquitectónica, la evolución de los formatos abiertos está eliminando el riesgo de dependencia tecnológica a través de la interoperabilidad omni-motor (omni-engine). Tecnologías de traducción bidireccional de metadatos como el proyecto XTable y características nativas como UniForm de Delta Lake habilitan lecturas en formato dual con mínima duplicación de datos (Thomas et al., 2025; Parimi, 2025). Esta convergencia permite que múltiples motores de ejecución distribuidos como Apache Spark, Trino o Presto interactúen simultáneamente con las mismas tablas sin incompatibilidades (Parimi, 2025). Al independizar los datos del motor de procesamiento, las organizaciones obtienen la agilidad necesaria para seleccionar la herramienta de cómputo más adecuada para cada caso de uso, maximizando la portabilidad en ecosistemas multinube (Thomas et al., 2025).

5.4.3 RENDIMIENTO, OPTIMIZACIÓN ALGORÍTMICA Y REDUCCIÓN DE COSTOS

Para igualar o superar la velocidad de consulta de los almacenes tradicionales, el Lakehouse implementa técnicas avanzadas de indexación, almacenamiento en caché y poda de particiones (partition pruning) directamente sobre los archivos almacenados en la nube (Armbrust et al., 2021). La aplicación de estrategias sofisticadas de particionamiento ha demostrado reducir los tiempos promedio de ejecución de consultas en un 87,3%, disminuyendo además los requerimientos de escaneo de datos en un 97,4% al omitir bloques de información irrelevantes (Chippada et al., 2025). De igual forma, el uso de marcos de enrutamiento inteligente de consultas permite reducir la latencia promedio en un 83,7% mientras incrementa el rendimiento general del clúster de procesamiento en un 572% mediante una asignación eficiente de recursos (Chippada et al., 2025).

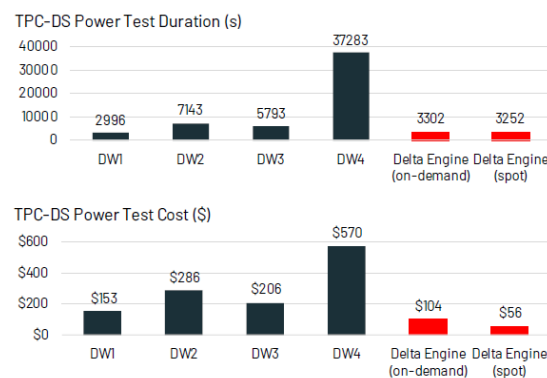


Figura 6. Puntuación de potencia de consulta y costo operativo a gran escala.

Adaptado de Armbrust et al. (2021). TPC-DS: Transaction Processing Performance Council Decision Support.

La Figura 6 ilustra estos resultados: los gráficos comparan el tiempo de ejecución y el costo económico del motor analítico del Lakehouse frente a cuatro almacenes de datos en la

nube tradicionales, evidenciando su superioridad bajo el conjunto de pruebas de rendimiento TPC-DS.

Para complementar estas optimizaciones algorítmicas, el Lakehouse alcanza su máxima rentabilidad a través de estrategias automatizadas de almacenamiento por niveles (storage tiering) y gestión del ciclo de vida del dato. Evaluaciones recientes a gran escala demuestran que las organizaciones que aplican mecanismos de estratificación inteligente logran clasificar, en promedio, el 91,3 % de sus volúmenes de datos en clases de almacenamiento apropiadas según su frecuencia de acceso, temperatura y criticidad comercial (Chippada et al., 2025). Este enfoque dinámico permite reservar el cómputo de alto rendimiento exclusivamente para el 9,7 % de los datos que concentran el 87,2 % de las consultas analíticas diarias, traduciéndose en una reducción directa del 53,7 % en los costos de almacenamiento en la nube frente a estrategias de nivel único (Chippada et al., 2025).

Adicionalmente, la arquitectura Lakehouse facilita la ejecución de políticas de retención altamente granulares. Las empresas con implementaciones maduras logran archivar proactivamente hasta un 78,3 % de su volumen total hacia niveles de almacenamiento en frío de bajo costo sin perder la capacidad de consulta transparente, lo que se traduce en una disminución del 72,4 % en los gastos de mantenimiento de almacenamiento en comparación con los enfoques tradicionales de retención indefinida (Chippada et al., 2025).

5.4.4 UNIFICACIÓN DE CARGAS DE TRABAJO: INTELIGENCIA DE NEGOCIOS E INTELIGENCIA ARTIFICIAL

Durante años, la arquitectura de dos niveles obligó a las empresas a mantener datos en un Data Lake para el entrenamiento de modelos de Machine Learning y a extraer un subconjunto transformado hacia un Data Warehouse para las tareas de Inteligencia de Negocios (BI), incrementando la latencia, la duplicación y la complejidad (Armbrust et al., 2021; Mezzoudj et al., 2025). El Lakehouse elimina ese cuello de botella al consolidar ambas cargas de trabajo sobre un mismo repositorio (Janssen et al., 2024). Dado que las bibliotecas de inteligencia artificial y ciencia de datos procesan grandes volúmenes con código imperativo (no SQL), la lectura directa desde formatos columnares abiertos evita las penalizaciones de rendimiento que surgen al extraer datos a través de conectores ODBC/JDBC desde un almacén tradicional (Armbrust et al., 2021).

Más allá de evitar esos cuellos de botella, el Lakehouse optimiza activamente las cargas de trabajo de aprendizaje automático. Los sistemas modernos de ciencia de datos como TensorFlow o Spark MLlib interactúan con la información a través de interfaces declarativas conocidas como DataFrames (Armbrust et al., 2021). Al operar sobre un ecosistema Lakehouse, estas interfaces permiten que las aplicaciones de inteligencia artificial aprovechen automáticamente las optimizaciones internas del motor de datos (almacenamiento en caché, índices y poda de particiones) sin necesidad de alterar el código del usuario, acelerando la fase de preparación de datos para los algoritmos (Armbrust et al., 2021).

A nivel operativo, esta unificación convierte al Lakehouse en la base natural para el ciclo de vida completo del aprendizaje automático mediante la integración de almacenes de características (Feature Stores) y prácticas de MLOps (Sundar et al., 2022). Al utilizar las tablas del Lakehouse como un Feature Store centralizado con aislamiento de instantáneas (snapshot isolation), las organizaciones garantizan que los modelos consuman características

estandarizadas y listas para producción, asegurando una alineación consistente entre el entorno de entrenamiento por lotes y el entorno de predicción en tiempo real (Thomas et al., 2025; Sundar et al., 2022). Esto elimina el sesgo o desviación de los datos predictivos, uno de los problemas más frecuentes en entornos de producción con pipelines de datos fragmentados (Sundar et al., 2022).

Al integrar el acceso rápido a nivel de archivo con la inmutabilidad histórica del versionado transaccional, el Lakehouse proporciona además la trazabilidad y el linaje de datos necesarios para garantizar la reproducibilidad precisa de los modelos predictivos y el cumplimiento de auditorías en entornos regulados (Mezzoudj et al., 2025). En la frontera más reciente, los formatos de tabla abiertos están incorporando extensiones vectoriales y tensoriales que permiten ejecutar búsquedas de similitud directamente sobre los archivos almacenados, habilitando patrones avanzados como la Generación Aumentada por Recuperación (RAG) sin necesidad de exportar la información hacia bases de datos vectoriales aisladas (Thomas et al., 2025). Esta evolución consolida al Lakehouse como la plataforma unificada de referencia tanto para la analítica tradicional como para las cargas de trabajo de inteligencia artificial generativa.

5.5 CASOS DE ESTUDIO Y APLICABILIDAD

La arquitectura Data Lakehouse no es únicamente un modelo teórico, sino una solución tecnológica validada empíricamente en industrias de alta exigencia. El análisis de implementaciones documentadas permite constatar cómo esta plataforma resuelve cuellos de botella operativos reales mediante métricas concretas de rendimiento, reducción de costos y soporte avanzado de cargas de trabajo.

5.5.1 SECTOR DE MANUFACTURA Y COMPONENTES AUTOMOTRICES

La gestión de datos a escala industrial presenta retos de integración considerables. Un caso documentado analiza a un fabricante de componentes automotrices de nivel 1 (Tier-1) con ingresos superiores a 12.700 millones de dólares, que procesaba 9,2 terabytes de datos nuevos al día y acumulaba 4,3 petabytes de datos históricos distribuidos en 35 fuentes desconectadas (sistemas ERP, plataformas CRM y sensores IoT) (Chippada et al., 2025). Al unificar toda esa infraestructura bajo una arquitectura Lakehouse, los resultados operativos fueron verificables: los defectos de fabricación cayeron un 8,7%, los costos de mantenimiento de inventario se redujeron un 16,9% y el rendimiento de entrega a tiempo mejoró un 11,2% (Chippada et al., 2025). A nivel técnico, la latencia promedio de las consultas bajó un 83,7% y los fallos de consulta disminuyeron un 89,3% frente a los sistemas fragmentados anteriores (Chippada et al., 2025).

5.5.2 SECTOR DE SALUD PÚBLICA Y EPIDEMIOLOGÍA

La pandemia de COVID-19 generó volúmenes masivos de datos epidemiológicos que requerían procesamiento en tiempo real para análisis comparativos internacionales. En ese contexto, la implementación empírica de un Data Lakehouse basado en almacenamiento de objetos y en el motor de virtualización de datos Dremio demostró ser la solución más adecuada para la ingesta continua y la unificación de registros (N'Guessan et al., 2023). Al activar los mecanismos de aceleración de consultas propios del Lakehouse (reflejos de Dremio), el tiempo de renderizado de mapas de calor globales se redujo de 33.196 milisegundos a tan solo 391 milisegundos, una mejora del 98,82% (N'Guessan et al., 2023).

Las consultas para calcular el total de casos confirmados pasaron de 29.194 milisegundos a 411 milisegundos (mejora del 98,59%), eliminando la latencia que los sistemas tradicionales imponían a las autoridades sanitarias (N'Guessan et al., 2023).

5.5.3 SECTOR MARÍTIMO Y MONITORIZACIÓN DE EMBARCACIONES (LOGÍSTICA)

El rastreo logístico y la monitorización de flotas plantean un reto extremo en términos de volumen de ingesta. Para abordarlo, se diseñó un Data Lakehouse específico para un Sistema de Monitorización de Embarcaciones (VMS), capaz de gestionar más de 354,8 millones de registros dinámicos del Sistema de Identificación Automática (AIS), con un peso superior a los 31 gigabytes solo en su capa transaccional dinámica (Park et al., 2023). Centralizar la información en este entorno no solo resolvió la fragmentación preexistente, sino que permitió integrar directamente modelos de inteligencia artificial para la predicción de rutas (como Random Forest y redes neuronales) sin necesidad de extraer o duplicar la información hacia sistemas aislados, optimizando así las labores de control y vigilancia marítima (Park et al., 2023).

5.5.4 SECTOR DE INTELIGENCIA GEOESPACIAL Y CIUDADES INTELIGENTES

El análisis de datos geoespaciales, indispensable tanto para el desarrollo de ciudades inteligentes como para la manufactura automotriz, supone un reto computacional severo. Para resolverlo, corporaciones del grupo Fortune 500 han adoptado implementaciones de Lakehouse como GeoLake, que extienden los formatos abiertos para soportar de forma nativa algoritmos de ordenamiento espacial (Zhang et al., 2023). En evaluaciones empíricas con conjuntos de datos reales a gran escala (rastreo de flotas e infraestructura urbana), la

agrupación de coordenadas bajo este formato redujo los requerimientos de almacenamiento físico, mientras que el empuje de predicados espaciales sobre los archivos permitió descartar bloques de datos irrelevantes de manera anticipada, acelerando los tiempos de lectura frente a bases de datos relacionales tradicionales (Zhang et al., 2023).

5.5.5 INVESTIGACIÓN BIOMÉDICA Y MEGA-BIOBANCOS

En el ámbito clínico, consolidar datos multimodales respetando estrictamente la privacidad del paciente es un requisito ineludible. Un ejemplo de alta complejidad es el mega-biobanco genómico gestionado por el Laboratorio Nacional de Oak Ridge (ORNL) (Begoli et al., 2021). Para gestionar datos médicos y de veteranos sensibles, esta arquitectura Lakehouse aplica etiquetas de control de acceso de grano fino directamente en el nivel de metadatos personalizados de los archivos Parquet, filtrando y restringiendo la información disponible para cada grupo de investigación (Begoli et al., 2021). La infraestructura unificada está orientada a facilitar la adopción de los principios FAIR (Encontrable, Accesible, Interoperable y Reutilizable), garantizando la seguridad normativa sin sacrificar la capacidad analítica (Begoli et al., 2021).

6. CONCLUSIONES

1. La arquitectura Data Lakehouse surge como respuesta directa a las limitaciones estructurales de sus predecesores. El Data Warehouse demostró ser inflexible ante la variedad de datos modernos, costoso de escalar y cerrado a las cargas de trabajo de inteligencia artificial. El Data Lake, aunque resolvió la flexibilidad y el costo de almacenamiento, careció de garantías transaccionales, gobernanza efectiva y rendimiento analítico adecuado. Las arquitecturas híbridas de dos niveles que intentaron combinar ambos modelos introdujeron complejidad operativa, datos desactualizados y duplicación de costos insostenibles. El Lakehouse resuelve estas deficiencias mediante una capa transaccional de metadatos sobre almacenamiento en formatos abiertos, habilitando transacciones ACID, versionado de datos y soporte simultáneo de cargas de trabajo multimodales sin necesidad de mover la información entre sistemas.
2. El modelo Lakehouse se sustenta en ocho requisitos técnicos formalizados por la comunidad científica, que garantizan un almacenamiento unificado, soporte CRUD completo, consistencia transaccional y procesamiento simultáneo de lotes y flujos en tiempo real. Su viabilidad práctica descansa en tres tecnologías habilitadoras principales: Delta Lake, destacado por su simplicidad operativa e integración con Apache Spark; Apache Iceberg, superior en cargas de trabajo analíticas masivas de lectura; y Apache Hudi, óptimo para la ingestión continua de flujos y la alta concurrencia de escritura. Cada tecnología refleja decisiones arquitectónicas distintas que favorecen diferentes perfiles de carga de trabajo, y la tendencia actual apunta hacia su convergencia e interoperabilidad plena, reduciendo el riesgo de dependencia tecnológica.

3. La comparación estructural entre las tres arquitecturas evidencia que el Lakehouse no simplemente mejora a sus predecesores en dimensiones aisladas, sino que resuelve simultáneamente las limitaciones que ninguno de los dos modelos anteriores pudo superar por separado. En gobernanza, su capa transaccional de metadatos reduce en un 92,1 % las interrupciones causadas por cambios de esquema, dotando al ecosistema de una continuidad operativa del 99,9 % que los lagos de datos tradicionales no podían garantizar. En escalabilidad, el desacoplamiento estructural del cómputo y el almacenamiento elimina el vendor lock-in que caracterizaba a los almacenes propietarios, generando ahorros promedio de 3,74 millones de dólares en costos de migración. En rendimiento, las técnicas de indexación y poda de particiones reducen los tiempos de consulta hasta un 87,3 % frente a los sistemas tradicionales, con una reducción del escaneo de datos del 97,4 %. En costos, el ROI promedio documentado alcanza el 427 %, combinando la reducción del almacenamiento redundante con la eliminación de las canalizaciones ETL intermedias que duplicaban tanto los datos como la infraestructura.
4. Los cinco casos de implementación documentados confirman la aplicabilidad transversal del modelo Lakehouse en sectores de alta exigencia operativa. En manufactura automotriz, la unificación de 35 fuentes desconectadas bajo una arquitectura Lakehouse redujo los defectos de fabricación un 8,7 % y la latencia de consultas un 83,7 %. En salud pública, el procesamiento de datos epidemiológicos del COVID-19 mejoró el rendimiento de consultas en un 98,82 %, eliminando la latencia crítica que los sistemas tradicionales imponían a las autoridades sanitarias. En logística marítima, la gestión de más de 354 millones de registros AIS permitió integrar modelos predictivos de inteligencia artificial directamente sobre el repositorio sin duplicación de datos. En inteligencia geoespacial, corporaciones Fortune 500 adoptaron implementaciones Lakehouse para soportar nativamente algoritmos de ordenamiento espacial a escala urbana. En investigación biomédica, el Laboratorio

Nacional de Oak Ridge implementó control de acceso de grano fino sobre metadatos Parquet para gestionar datos genómicos sensibles bajo los principios FAIR, garantizando seguridad normativa sin sacrificar capacidad analítica.

7. RECOMENDACIONES

1. Se recomienda a futuras investigaciones desarrollar evaluaciones empíricas comparativas de los tres formatos de tabla abiertos (Delta Lake, Apache Iceberg y Apache Hudi) bajo condiciones de infraestructura controladas y reproducibles, incorporando simultáneamente cargas de trabajo de aprendizaje automático, procesamiento de flujos en tiempo real y consultas analíticas masivas. La literatura actual evalúa estas tecnologías de forma parcial o bajo condiciones heterogéneas que dificultan la comparación objetiva, lo que limita la capacidad de los equipos de ingeniería para seleccionar el formato más adecuado a su perfil operativo específico.
2. Se recomienda investigar la viabilidad técnica y económica de implementar arquitecturas Lakehouse en organizaciones de mediana escala en economías emergentes, particularmente en América Latina, donde las barreras de infraestructura en la nube, costos de licenciamiento y disponibilidad de talento especializado difieren significativamente de los contextos documentados en la literatura internacional. Identificar configuraciones mínimas viables y estrategias de migración graduales adaptadas a estas realidades permitiría democratizar el acceso a este modelo arquitectónico más allá de las grandes corporaciones.
3. Se recomienda desarrollar estudios longitudinales que midan el impacto operativo real del modelo Lakehouse en organizaciones que hayan completado migraciones desde arquitecturas tradicionales, con seguimiento de métricas de gobernanza, calidad del dato y costos totales de propiedad durante períodos superiores a tres años. La mayoría de los casos documentados actualmente corresponden a evaluaciones post-implementación inmediata, lo que impide conocer el comportamiento del modelo a

largo plazo, su curva de madurez operativa y los factores que determinan su éxito o fracaso sostenido en producción.

4. Se recomienda examinar el impacto de la incorporación de extensiones vectoriales y capacidades de inteligencia artificial generativa directamente sobre la capa transaccional del Lakehouse, evaluando cómo estas integraciones afectan el rendimiento de las cargas de trabajo analíticas tradicionales, los esquemas de gobernanza existentes y los costos operativos. Dado que esta convergencia entre el Lakehouse y la IA generativa representa la frontera más activa del campo, su estudio sistemático permitirá anticipar los requisitos arquitectónicos de la próxima generación de plataformas de datos empresariales.

8. REFERENCIAS BIBLIOGRÁFICAS

- Armbrust, M., Das, T., Sun, L., Yavuz, B., Zhu, S., Murthy, M., Torres, J., van Hovell, H., Ionescu, A., Łuszczak, A., Świtakowski, M., Szafranski, M., Li, X., Ueshin, T., Mokhtar, M., Boncz, P., Ghodsi, A., Paranjpye, S., Senster, P., ... Zaharia, M. (2020). Delta Lake: High-performance ACID table storage over cloud object stores. *Proceedings of the VLDB Endowment*, 13(12), 3411–3424. <https://doi.org/10.14778/3415478.3415560>
- Armbrust, M., Ghodsi, A., Xin, R. y Zaharia, M. (2021). Lakehouse: A new generation of open platforms that unify data warehousing and advanced analytics. *11th Annual Conference on Innovative Data Systems Research (CIDR '21)*.
- Azzabi, S., Alfughi, Z. y Ouda, A. (2024). Data lakes: A survey of concepts and architectures. *Computers*, 13(7), 183. <https://doi.org/10.3390/computers13070183>
- Begoli, E., Goethert, I. y Knight, K. (2021). A Lakehouse Architecture for the Management and Analysis of Heterogeneous Data for Biomedical Research and Mega-biobanks. *2021 IEEE International Conference On Big Data (Big Data)*, 4643-4652. <https://doi.org/10.1109/bigdata52589.2021.9671534>
- Bentaib, M., Ettaoufik, A., Tragha, A. y Azzouazi, A. (2024). Storage structures in the era of big data: From data warehouse to lakehouse. *Journal of Theoretical and Applied Information Technology*, 102(6), 2428–2441.

Cappa, F., Acciarini, C., Oriani, R. y Boccardelli, P. (2025). Big Data and innovation: The state of the art and future research directions. *R&D Management*.
<https://doi.org/10.1111/radm.70004>

Cherradi, M. y Haddadi, A. E. (2024). Data Lakehouse: Next generation Information System. *Seminars In Medical Writing And Education*, 3, 67. <https://doi.org/10.56294/mw202467>

Chippada, S. S., Agrawal, S. y Vats, R. (2025). Data lakehouse implementation: A journey from traditional data warehouses. *World Journal of Advanced Engineering Technology and Sciences*, 15(1), 311–332.
<https://doi.org/10.30574/wjaets.2025.15.1.0224>

Commvault. (2025, September 30). *Commvault delivers industry-first cyber resilience for AI Data Lakehouses on AWS with support for Apache Iceberg tables* [Press release]. Commvault.

Gade, K. R. (2022). Data lakehouses: Combining the best of data lakes and data warehouses. *Journal of Computational Innovation*, 2(1).
<https://researchworkx.com/index.php/jci/article/view/3/3>

Gentner, T., Neitze, T., Schulze, J., Gerschner, F. y Theissler, A. (2023). Data lakes in healthcare: Applications and benefits from the perspective of data sources and players. *Procedia Computer Science*, 225, 1302–1311.
<https://doi.org/10.1016/j.procs.2023.10.118>

- Grandhe, K. (2025). From chaos to clarity: Building a modern data lakehouse for SAP financial data. *International Journal for Multidisciplinary Research*, 7(4).
<https://doi.org/10.36948/ijfmr.2025.v07i04.54379>
- Isnaeni, N., Putranto, B. P. D., Andriyani, W. y Khomsah, S. (2025). Comprehensive lakehouse data architecture model for college accreditation. *Journal of Dinda: Data Science, Information Technology, and Data Analytics*, 5(1), 45–51.
<https://doi.org/10.20895/dinda.v5i1.1759>
- Jain, P., Kraft, P., Power, C., Das, T., Stoica, I. y Zaharia, M. (2023). Analyzing and comparing Lakehouse storage systems. *Proceedings of the Conference on Innovative Data Systems Research (CIDR 2023)*.
<https://vldb.org/cidrdb/2023/analyzing-and-comparing-lakehouse-storage-systems.html>
- Janssen, N., Ilayperuma, T., Jayasinghe, J., Bukhsh, F. y Daneva, M. (2024). The evolution of data storage architectures: examining the secure value of the Data Lakehouse. *Journal of Data Information And Management*, 6(4), 309-334. <https://doi.org/10.1007/s42488-024-00132-1>
- Kasireddy, J. R. (2023). Operationalizing lakehouse table formats: A comparative study of Iceberg, Delta, and Hudi workloads. *International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)*, 6(2).
<https://doi.org/10.15662/IJRPETM.2023.0602002>
- Menon, P. (2022). *Data Lakehouse in action: Architecting a modern and scalable data analytics platform*. Packt Publishing.

- Mezzoudj, S., Khelifa, M. y Saadna, Y. (2025). Data lakes versus data warehouses: Choosing the right approach for big data analytics. *Journal of Electrical Systems and Information Technology*, 12, 89. <https://doi.org/10.1186/s43067-025-00275-0>
- Mishra, S. (2022). Comparing Apache Iceberg and Databricks in building data lakes and mesh architectures. *International Journal of AI, BigData, Computational and Management Studies*, 3(4), 37–48. <https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I4P105>
- Nambiar, A. y Mundra, D. (2022). An overview of data warehouse and data lake in modern enterprise data management. *Big Data and Cognitive Computing*, 6(4), 132. <https://doi.org/10.3390/bdcc6040132>
- N'Guessan, B. G., Assie, B. I., Wamba, S. F. y Achiepo, O. Y. M. (2023). Implementation of a data lakehouse for efficient recovery and processing of massive data. *SSRG International Journal of Computer Science and Engineering*, 10(11), 7–12. <https://doi.org/10.14445/23488387/IJCSE-V10I11P102>
- Parimi, S. (2025). A comparative performance & metadata study of open table formats: Iceberg vs Delta vs Hudi at scale. *Journal of Computer Science and Technology Studies*, 7(12), 513–520. <https://doi.org/10.32996/jcsts.2025.7.12.56>
- Park, S., Yang, C.-S. y Kim, J. (2023). Design of vessel data lakehouse with big data and AI analysis technology for vessel monitoring system. *Electronics*, 12(8), 1943. <https://doi.org/10.3390/electronics12081943>

- Schneider, J., Gröger, C., Lutsch, A., Schwarz, H. y Mitschang, B. (2024). The Lakehouse: State of the Art on Concepts and Technologies. *SN Computer Science*, 5(5). <https://doi.org/10.1007/s42979-024-02737-0>
- Schneider, J., Gröger, C., Lutsch, A., Schwarz, H. y Mitschang, B. (2023). Assessing the Lakehouse: Analysis, requirements and definition. *Proceedings of the 25th International Conference on Enterprise Information Systems – Volume 1 (ICEIS)* (pp. 44–56). <https://doi.org/10.5220/0011840500003467>
- Sundar, D., Jayaram, Y. y Bhat, J. (2022). A comprehensive cloud data lakehouse adoption strategy for scalable enterprise analytics. *International Journal of Emerging Research in Engineering and Technology*, 3(4), 92–103. <https://doi.org/10.63282/3050-922X.IJERET-V3I4P111>
- Thomas, M., Gonsalez, L., Jacob, R. y Mathew, T. (2025). Data lakehouse architecture: Bridging the gap between data lakes and data warehouses. *IRE Journals*, 9(4). <https://doi.org/10.64388/IREV9I4-1711258-3166>
- Zhang, Y., Peng, B., Du, Y. y Su, J. (2023). GeoLake: Bringing geospatial support to lakehouses. *IEEE Access*, 11, 143037–143049. <https://doi.org/10.1109/ACCESS.2023.3343953>

Firma del estudiante y fecha:

Dylan Ariel Flores Muñoz

Quito, 07 de enero de 2026

Firma del director de monografía y fecha:

Mgtr. Kevin Ricardo Rojas Satián

Quito, 07 de enero de 2026

Firma del Coordinador de la Escuela y fecha:

Mgtr. Kevin Ricardo Rojas Satián

Quito, 07 de enero de 2026