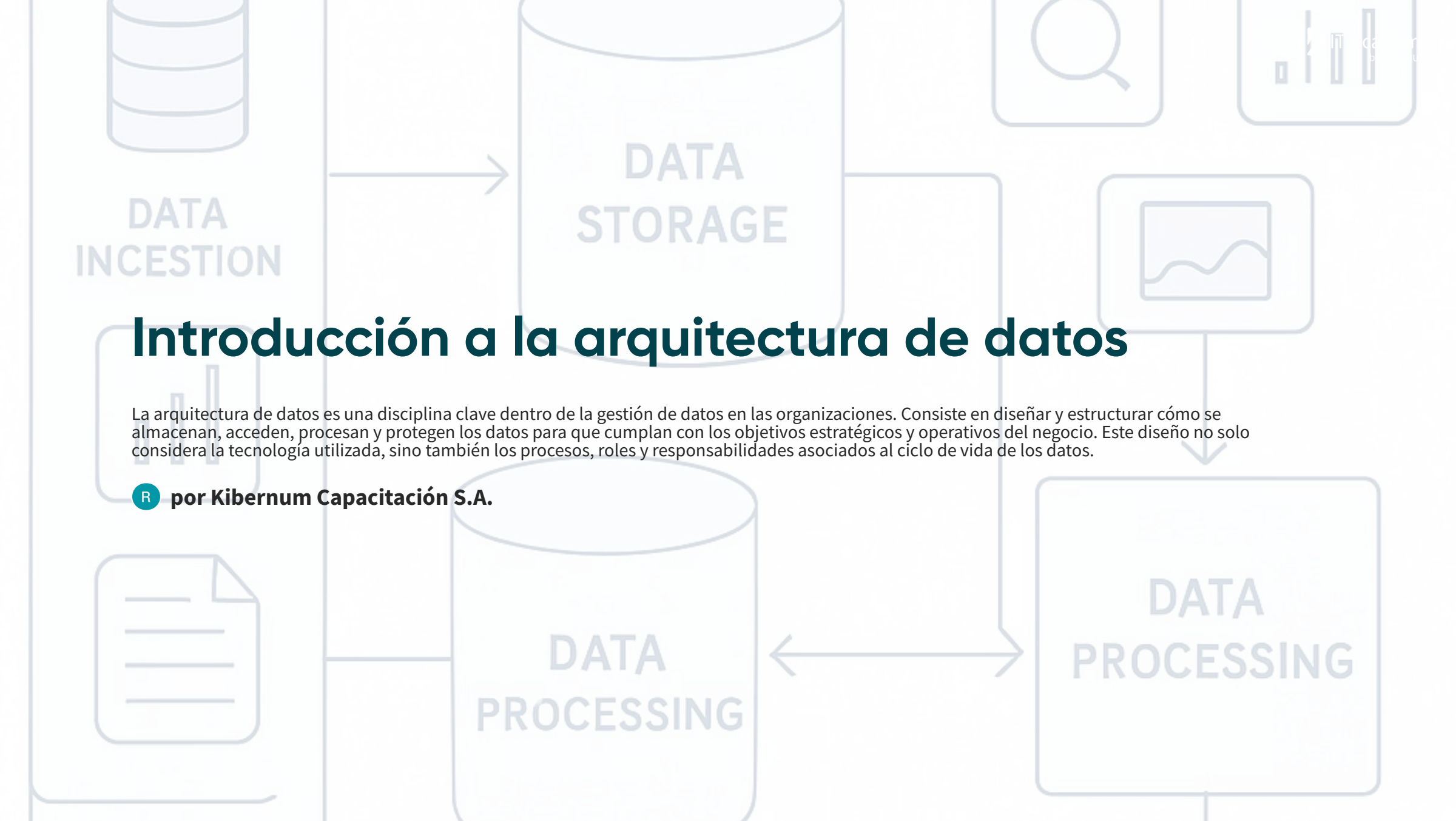


Introducción a la arquitectura de datos

La arquitectura de datos es una disciplina clave dentro de la gestión de datos en las organizaciones. Consiste en diseñar y estructurar cómo se almacenan, acceden, procesan y protegen los datos para que cumplan con los objetivos estratégicos y operativos del negocio. Este diseño no solo considera la tecnología utilizada, sino también los procesos, roles y responsabilidades asociados al ciclo de vida de los datos.

R **por Kibernum Capacitación S.A.**

¿Qué es la arquitectura de datos?

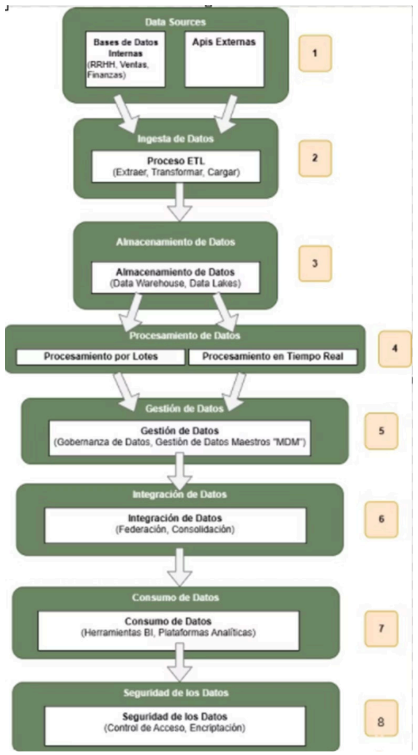
La arquitectura de datos es una disciplina clave dentro de la gestión de datos en las organizaciones. Consiste en diseñar y estructurar cómo se almacenan, acceden, procesan y protegen los datos para que cumplan con los objetivos estratégicos y operativos del negocio. Este diseño no solo considera la tecnología utilizada, sino también los procesos, roles y responsabilidades asociados al ciclo de vida de los datos.

Esta arquitectura actúa como un plano o mapa de alto nivel que establece las políticas, estándares y reglas necesarias para el uso efectivo, seguro y eficiente de los datos en toda la organización. Una buena arquitectura de datos permite garantizar la trazabilidad, la calidad, la gobernanza y la disponibilidad de la información, facilitando así la toma de decisiones basada en datos y la generación de valor a partir de ellos.

Esquema conceptual de arquitectura de datos empresarial

Este diagrama representa de forma visual y ordenada cómo fluye la información en una arquitectura de datos empresarial moderna. Este tipo de esquemas se utiliza ampliamente en organizaciones para planificar, diseñar y entender cómo se gestionan los datos desde su origen hasta su uso final en la toma de decisiones.

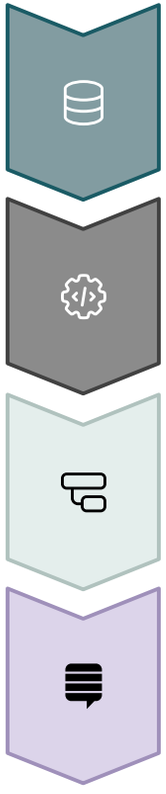
Cada caja del diagrama representa una etapa clave del recorrido que siguen los datos dentro de una empresa:



•

Componentes del flujo de datos empresarial

- Fuentes de Datos (Data Sources): Todo parte aquí. Los datos pueden provenir de sistemas internos como Recursos Humanos, Finanzas o Ventas, así como de APIs externas. Estas fuentes son variadas, y su correcta identificación es el primer paso en el diseño de una buena arquitectura.
- Ingesta de Datos (Data Ingestion): En esta etapa se realiza el famoso proceso ETL: Extraer, Transformar y Cargar los datos. Este paso prepara la información para que pueda ser almacenada y procesada de manera eficiente.



Almacenamiento de Datos (Data Storage)

Una vez transformados, los datos se almacenan en sistemas como Data Warehouses o Data Lakes, dependiendo del tipo y volumen de información. Este componente es esencial para tener una base sólida y estructurada que soporte los análisis posteriores.

Procesamiento de Datos (Data Processing)

Aquí ocurre la "magia". Los datos se procesan ya sea en lotes (Batch) o en tiempo real (Stream), según la necesidad del negocio. Por ejemplo, una empresa de logística podría usar procesamiento en tiempo real para rastrear envíos.

Gestión de Datos (Data Management)

Esta parte asegura que los datos estén bien organizados, actualizados y gobernados. Se incluyen prácticas como la gobernanza de datos y la gestión de datos maestros (MDM), fundamentales para mantener la calidad y coherencia en toda la organización.

Integración de Datos (Data Integration)

Una vez gestionados, los datos deben integrarse, es decir, combinarse desde distintas fuentes para construir una visión unificada y útil. Se utilizan técnicas como federación o consolidación de datos.

Consumo de Datos (Data Consumption)
Es el momento en que los usuarios finales (analistas, gerentes, áreas de negocio) acceden a los datos a través de herramientas como **BI (Business Intelligence)** o **plataformas analíticas**, para generar reportes, dashboards o tomar decisiones basadas en evidencia

Seguridad de los Datos (Data Security)
Finalmente, todo este sistema debe estar protegido. Aquí se aplican políticas de **control de acceso** y **encriptación** para asegurar que los datos estén disponibles solo para quienes los necesitan y se mantengan seguros frente a amenazas.

¿Qué es un Arquitecto de Datos?

Un arquitecto de datos es como el urbanista de la información: diseña la ciudad (los sistemas de datos), define las rutas (los flujos de información), y garantiza que todo esté bien conectado, con buena seguridad y normas claras.

Por lo tanto, una definición mas formal seria: El arquitecto de datos es el profesional encargado de diseñar, mantener y supervisar la arquitectura de datos en una organización.

Roles de un Arquitecto de datos

Función	Descripción práctica	Ejemplo
Visión Estratégica	Define cómo los datos pueden apoyar la toma de decisiones y la estrategia de negocio.	"Los datos de clientes deberían integrarse con los datos de ventas para predecir abandono."
Estándares Técnicos	Establece reglas para calidad, integración, interoperabilidad y seguridad.	"Usaremos formato JSON estandarizado para las APIs internas."
Gobernanza de Datos	Se asegura de que existan políticas claras sobre uso, propiedad y acceso a los datos.	"Cada departamento debe tener un Data Steward responsable de sus datos."
Escalabilidad	Garantiza que la arquitectura soporte el crecimiento de datos y usuarios.	"Esta plataforma en la nube puede crecer sin afectar el rendimiento."

En organizaciones data-driven, este rol es fundamental para conectar el mundo técnico (infraestructura, plataformas, herramientas) con el mundo del negocio (análisis, indicadores, estrategia).

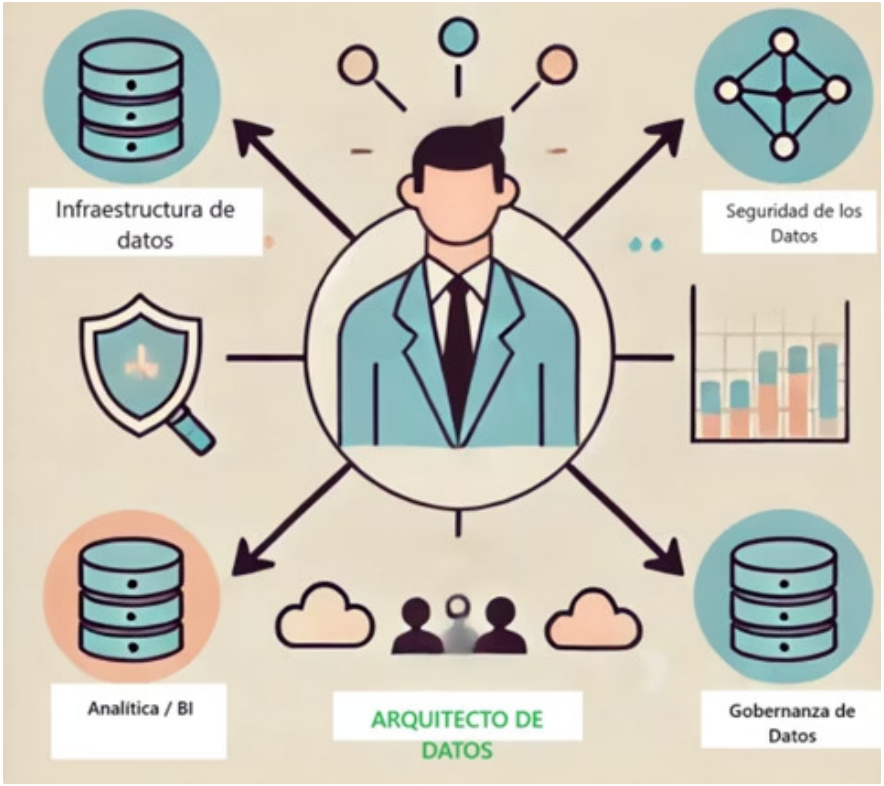
Relación entre Arquitecto e Ingeniero de Datos

"El Arquitecto de Datos define la estructura general y los estándares para la gestión de datos en una organización. Posteriormente, el Ingeniero de Datos se encarga de implementar y mantener esos flujos de datos, asegurando que funcionen eficientemente en los entornos definidos."

¿Qué pasa sin un arquitecto de datos?

- Las herramientas se implementan sin cohesión.
- Los equipos duplican esfuerzos y los datos se fragmentan.
- No hay estándares ni control de calidad.
- El análisis se vuelve lento e impreciso.

En organizaciones **data-driven,** este rol es fundamental para conectar el mundo técnico (infraestructura, plataformas, herramientas) con el mundo del negocio (análisis, indicadores, estrategia).



Diferencia con el Ingeniero de Datos

En este punto, es importante establecer algunas diferencias clave entre el Arquitecto de Datos y el Ingeniero de Datos. Profundizaremos en el rol del Ingeniero de Datos en el módulo 6.

Arquitecto de Datos

- Define el marco estratégico y tecnológico
- Diseña la arquitectura
- Se enfoca en estándares, interoperabilidad y gobernanza

Ingeniero de Datos

- Implementa y opera los flujos de datos
- Construye pipelines y modelos
- Se enfoca en eficiencia, rendimiento y automatización

Piensa en el arquitecto como el diseñador del plano, y el ingeniero como quien lo convierte en realidad.

Componentes básicos de la arquitectura de datos

A continuación, analizaremos en profundidad los seis componentes fundamentales que constituyen una arquitectura de datos moderna. Para una guía más completa de estos dominios, puedes consultar el marco de referencia DAMA-DMBOK, ampliamente utilizado en la industria para definir las buenas prácticas en la gestión de datos.

Cada uno cumple una función crítica dentro del ecosistema de gestión de datos, y entender su rol nos permite visualizar cómo fluye la información desde su origen hasta su consumo analítico. La correcta integración y coordinación entre estos componentes es clave para lograr una solución de datos robusta, escalable y segura.

La arquitectura de datos moderna se construye a partir de un conjunto de componentes que trabajan en conjunto para permitir la:

- Recolección
- Transformación
- Almacenamiento y
- Análisis de los datos.

Fuentes de datos

Toda arquitectura comienza con los datos en su forma más cruda, tal como se generan o capturan en los sistemas. Estos datos provienen de diversas fuentes, que son los lugares donde se originan y, por tanto, deben ser cuidadosamente identificadas y clasificadas exploremos un poco mas sobre fuentes internas y externas de datos.

Internas:

Provienen de dentro de la organización. Incluyen:

Sistemas ERP (Enterprise Resource Planning)

Función: Gestionar procesos empresariales integrados (finanzas, logística, etc.).

Ejemplos: SAP, Oracle ERP, Microsoft Dynamics, NetSuite.

Sistemas CRM (Customer Relationship Management)

Función: Administrar interacciones con clientes.

Ejemplos: Salesforce, HubSpot, Zoho CRM.

Bases de datos transaccionales

Función: Almacenar y procesar operaciones en tiempo real.

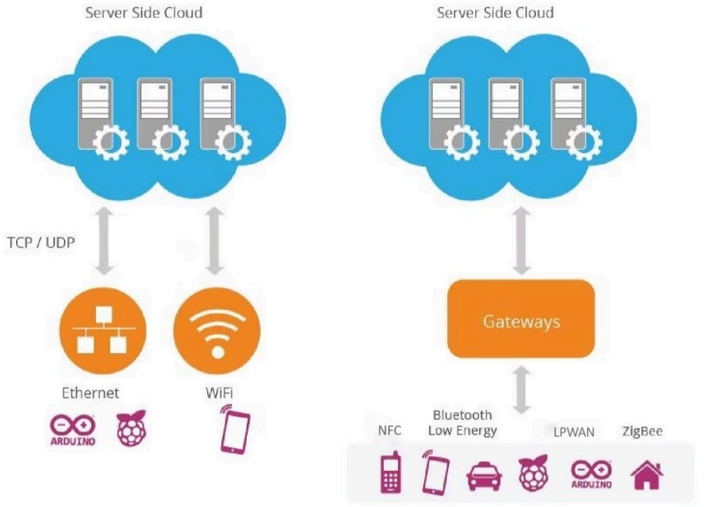
Ejemplos: Oracle Database, Microsoft SQL Server, PostgreSQL.

Más fuentes de datos internas

Sensores IoT (Internet of Things)

Función: Recopilar datos de dispositivos conectados.

Ejemplos: Sensores industriales, wearables, sensores ambientales.



Aplicaciones internas de gestión

Función: Automatizar procesos específicos.

Ejemplos:

- Personalizadas: Software de inventario o RRHH.
- Low-code: Power Apps, OutSystems.

Tecnología	Función principal	Ejemplos comunes
Sistemas ERP	Integrar procesos empresariales	SAP, Oracle ERP, Microsoft Dynamics
Sistemas CRM	Gestionar relaciones con clientes	Salesforce, HubSpot, Zoho CRM
Bases de datos transaccionales	Almacenar datos operacionales	Oracle Database, SQL Server, MySQL
Sensores IoT	Monitorear activos y ambientes	Sensores industriales, wearables
Aplicaciones internas	Automatizar procesos específicos	Software de BI, Power Apps, soluciones a medida

Fuentes de datos externas

Se refieren a datos que provienen de terceros. Algunos ejemplos son APIs de servicios públicos, portales de datos abiertos del gobierno, redes sociales como Twitter, proveedores de datos financieros o meteorológicos, entre otros.

Observemos con un poco mas de detalle los ejemplos mencionados.

APIs de servicios públicos

Función: Obtener datos estandarizados de entidades externas.

Ejemplos: APIs de bancos, sistemas de transporte, servicios de pago (Stripe, PayPal).

Portales de datos abiertos

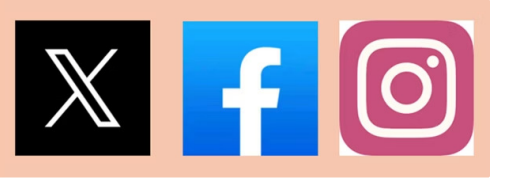
Función: Acceder a datasets gubernamentales o institucionales.

Ejemplos: data.gov (EE.UU.), datos.gob.es (España), World Bank Open Data.

Redes sociales

Función: Recopilar interacciones y tendencias de usuarios.

Ejemplos: Twitter (X), Facebook, LinkedIn, Instagram.



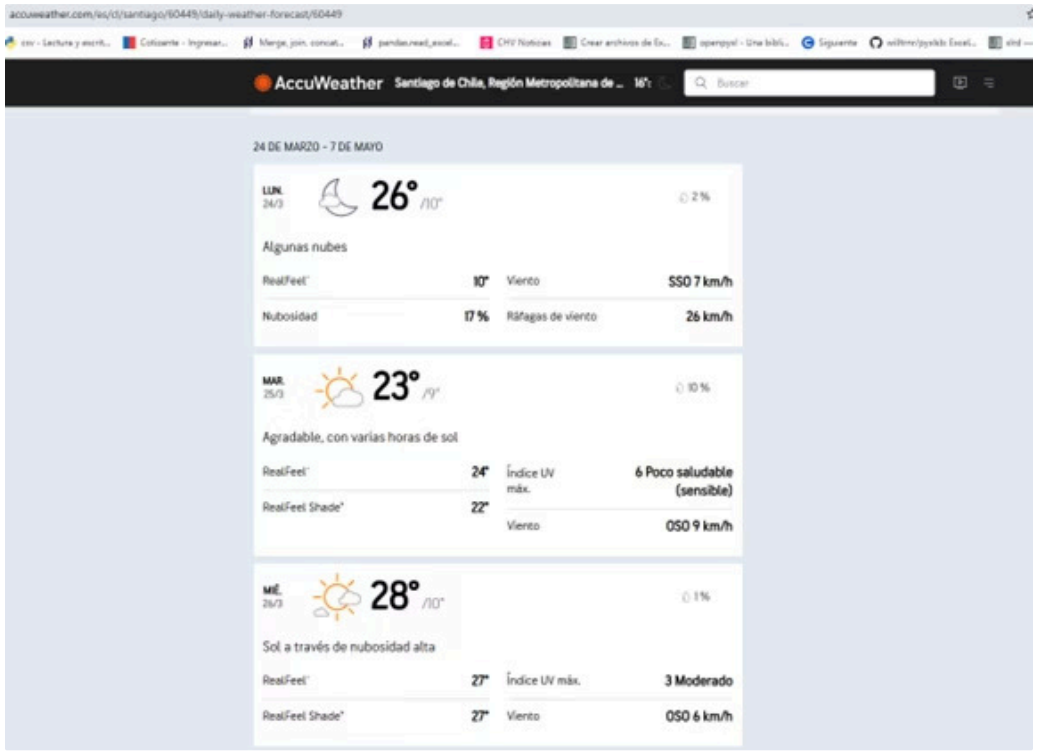
Recurso adicional: Para conjuntos de datos de ejemplo y casos prácticos, se recomienda explorar Kaggle, plataforma que alberga miles de datasets públicos para análisis y machine learning.

Proveedores de datos especializados

Función: Suministrar información sectorial (financiera, meteorológica, etc.).

Mejores alternativas para datos sectoriales:

- Datos financieros: Bloomberg Professional Services
- Datos climáticos: AccuWeather
- Datos de mercados: NielsenIQ



En esta imagen podemos apreciar un reporte típico de datos meteorológicos sectoriales, como los que provee AccuWeather para empresas. Incluye métricas esenciales para la toma de decisiones logísticas, agrícolas o de seguridad:

Variables visibles:

- Temperatura actual (26°C, 23°C, 28°C en diferentes ubicaciones).
- Condiciones atmosféricas ("Algunas nubes", "Soleado", "Radiación alta").
- Velocidad del viento (580.7 km/h, 26 km/h).
- Índice UV ("6 - Riesgo saludable", "3 - Moderado").

Aplicaciones empresariales:
✓ Optimización de rutas de transporte.
✓ Planificación de eventos al aire libre.
✓ Gestión de riesgos en agricultura.

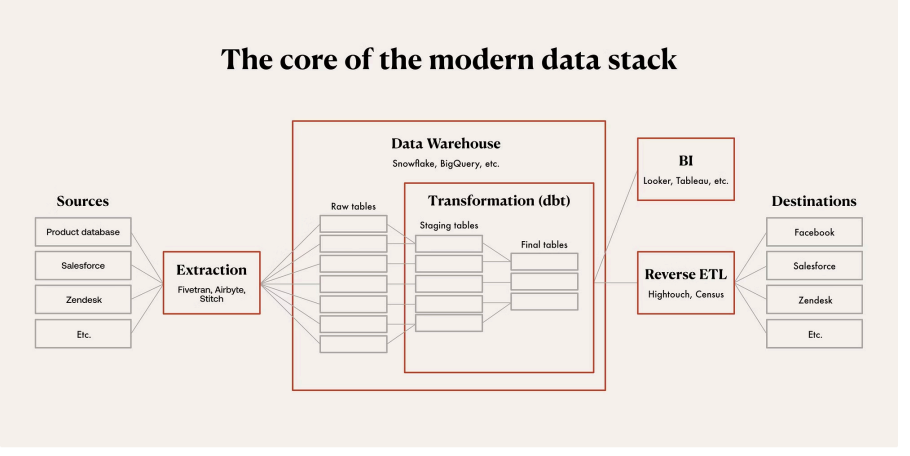
Plataformas de terceros y almacenamiento de datos

Plataformas de terceros

Función: Integrar datos de socios comerciales o SaaS.

Ejemplos: Shopify (e-commerce), Google Analytics (tráfico web), Salesforce (datos compartidos por clientes).

Es crucial considerar la confiabilidad, frecuencia de actualización y estructura de estos datos para integrarlos eficazmente al ecosistema organizacional de la información que será procesada.



🔗 Referencia: modern data stack

Almacenamiento de datos

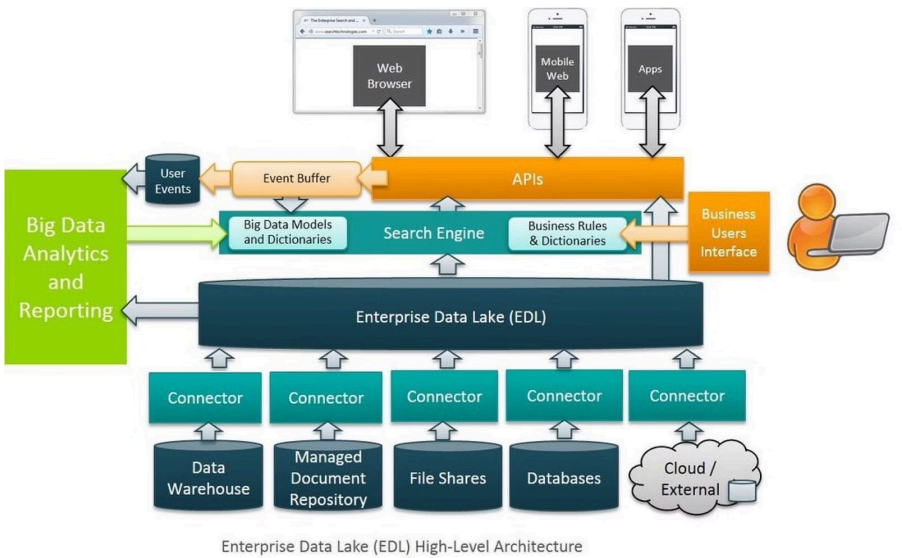
Una vez que los datos son capturados desde las fuentes, deben ser almacenados de manera segura y estructurada para permitir su posterior procesamiento y análisis. El almacenamiento puede ser:

- Relacional (SQL): como PostgreSQL, MySQL u Oracle. Ideal para datos estructurados y relaciones bien definidas. Muy útil para aplicaciones OLTP (procesamiento de transacciones en línea).
- No Relacional (NoSQL): como MongoDB, Cassandra o Redis. Adecuado para datos semiestructurados o que cambian rápidamente. Se adapta bien a aplicaciones web, logs o análisis en tiempo real.

Tipos de almacenamiento de datos

Data Lake

Espacio centralizado en la nube que permite guardar grandes volúmenes de datos en su forma original, sin necesidad de transformarlos previamente. Ideal para análisis exploratorios y uso de datos no estructurados.

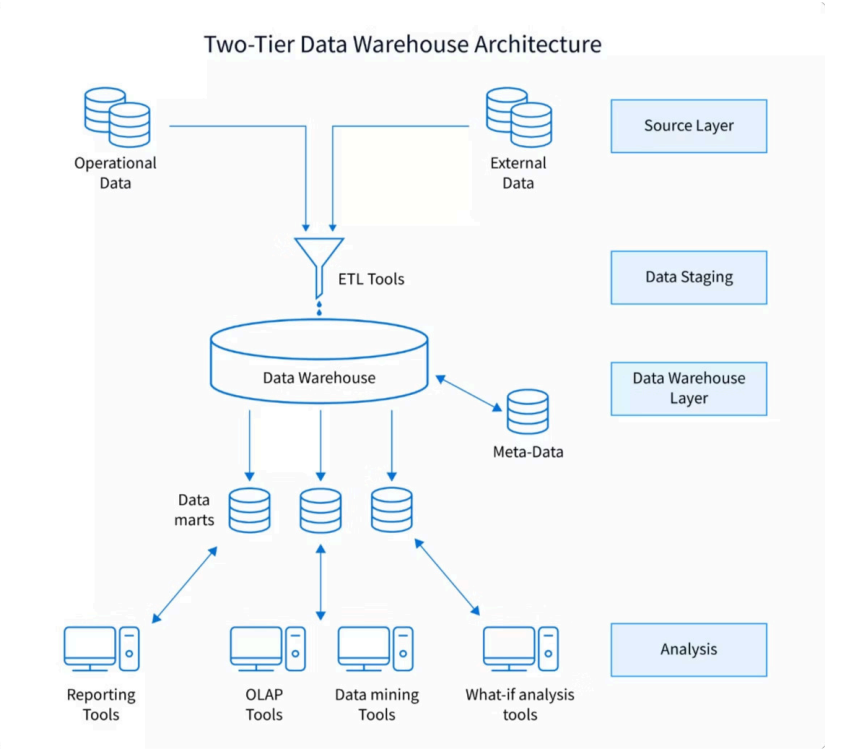


Enterprise Data Lake (EDL) High-Level Architecture

Video: Data Lakes: Características básicas y tecnologías

Data Warehouse

Sistemas diseñados específicamente para análisis de negocio (OLAP). Permiten consultas rápidas sobre grandes volúmenes de datos consolidados, como Snowflake o BigQuery.



Procesamiento de datos

El procesamiento es la fase en la que los datos brutos se transforman en información valiosa. Es un paso esencial para limpiar, validar, enriquecer y transformar los datos según las necesidades del negocio.

ETL (Extract, Transform, Load)

Se extraen los datos de las fuentes, se transforman (limpieza, estandarización, agregación) y se cargan a un destino como un Data Warehouse.

ELT (Extract, Load, Transform)

Primero se cargan los datos al destino (como un Data Lake) y luego se transforman, lo que permite aprovechar el poder computacional del sistema de destino.

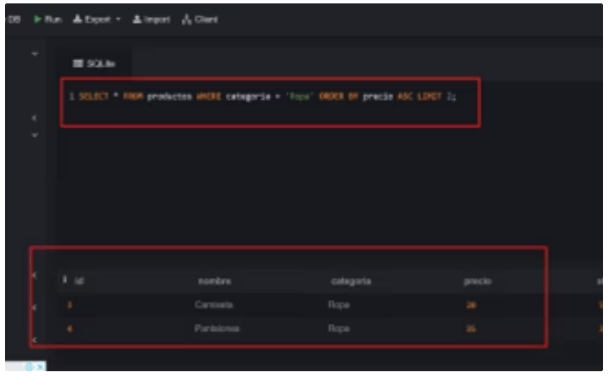
Frameworks y herramientas:

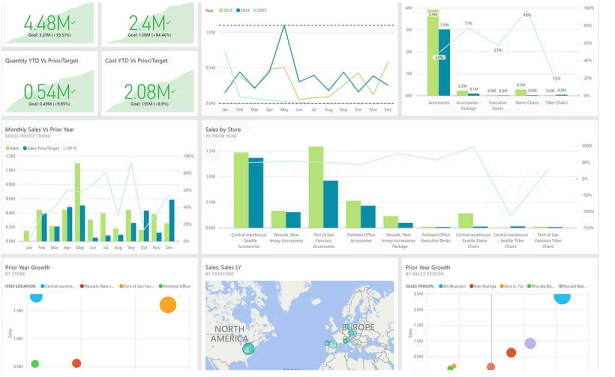
- Apache Spark: Procesamiento distribuido (batch/streaming). Ejemplo: Análisis de logs en tiempo real.
- Hadoop: Batch a gran escala (HDFS + MapReduce). Caso de uso: Procesar petabytes históricos.
- dbt: Transformaciones SQL + versionamiento. Típico: Modelado analítico en BigQuery.
- Airflow: Orquestación con DAGs (Python). Ejemplo: Programar pipelines diarios.

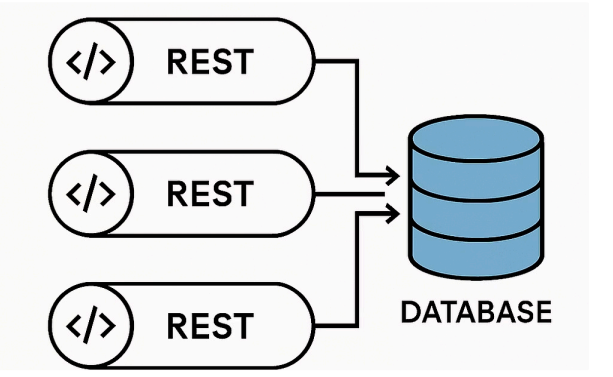
Batch vs Tiempo real: el procesamiento puede realizarse en bloques (batch) o de forma continua y en tiempo real (streaming), dependiendo del caso de uso.

Acceso a datos

Una vez procesados, los datos deben ser accesibles para quienes los necesitan, ya sean personas o sistemas. Este componente se encarga de presentar los datos de manera comprensible y oportuna.







Consultas directas

Los analistas pueden acceder directamente a los datos a través de SQL u otros lenguajes, dependiendo del sistema de almacenamiento.

Imagen: Consulta SQL en SQLite extrayendo datos filtrados y ordenados de una tabla 'productos'

Dashboards y reportes

Herramientas como Power BI, Tableau o Looker permiten visualizar indicadores clave, gráficos y métricas de forma interactiva.

Imagen: "Dashboard en Power BI con análisis de compras: $4.13M en ventas, 499 transacciones, distribución por cliente/proveedor y rendimiento trimestral con filtros interactivos."

APIs de datos

Permiten que aplicaciones o servicios externos consuman datos automáticamente, fomentando la integración entre plataformas.

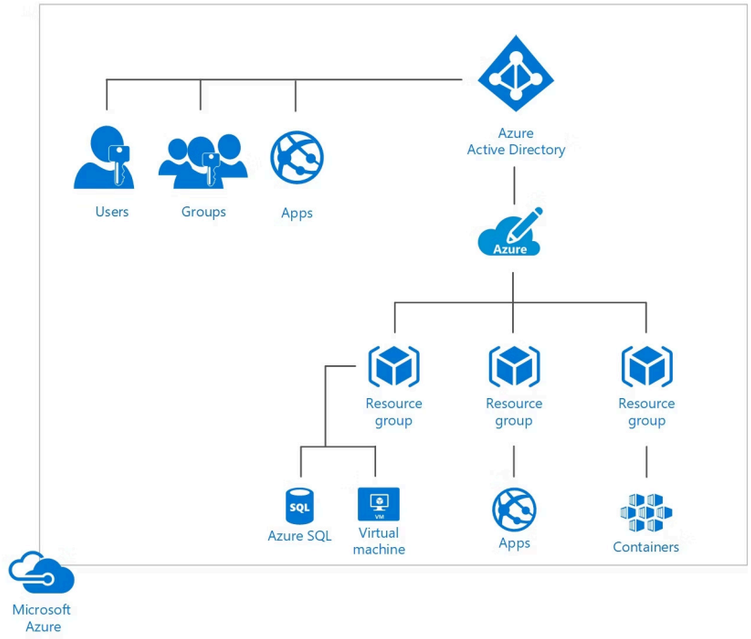
La capa de acceso debe estar bien diseñada para que no comprometa el rendimiento ni la seguridad del sistema.

Seguridad y gobernanza de datos

Seguridad

La seguridad de los datos no es un componente aislado, sino un aspecto transversal que debe implementarse en cada etapa del ciclo de vida del dato. Incluye:

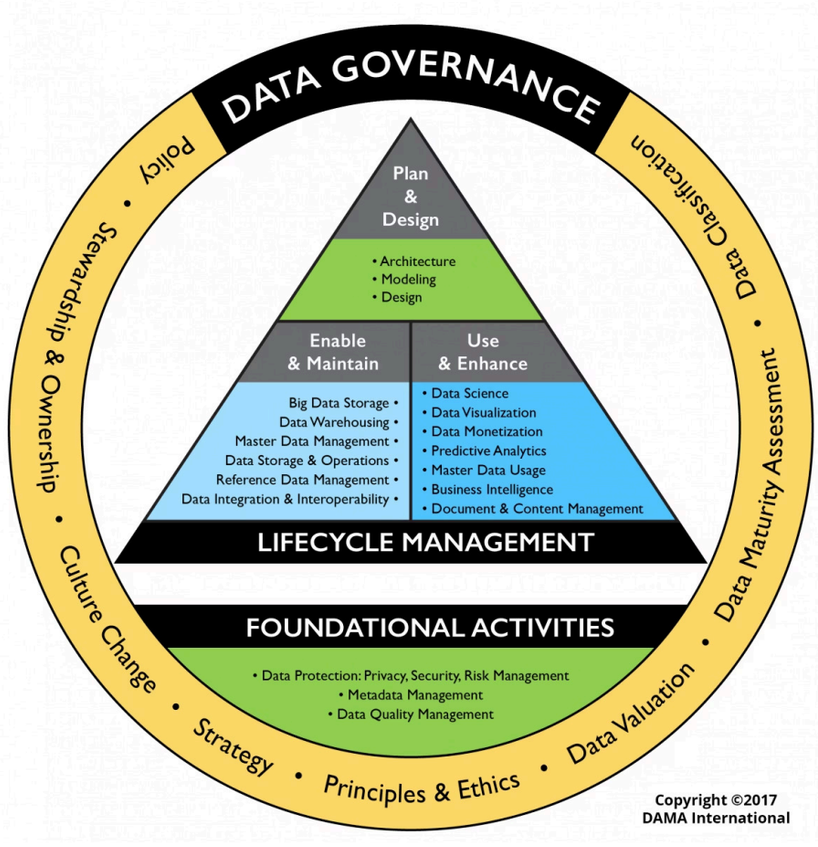
- Gestión de accesos y roles: establecer quién puede acceder a qué datos y con qué permisos.
- Autenticación multifactor (MFA): capa adicional para prevenir accesos no autorizados.
- Encriptación: proteger los datos tanto en tránsito (cuando se mueven por la red) como en reposo (cuando están almacenados).
- Auditoría y monitoreo: seguimiento de actividades sospechosas y revisión de logs de acceso.



Gobernanza de datos

La gobernanza de datos establece el marco de reglas, responsabilidades y procesos que aseguran que los datos sean gestionados como un activo estratégico. Algunos elementos clave son:

- Catálogo de datos: repositorio centralizado que permite conocer qué datos existen, dónde están y cómo se utilizan.
- Diccionario de datos: define los términos clave, estructuras y formatos para evitar ambigüedades.
- Políticas de calidad: aseguran que los datos sean precisos, completos, actualizados y consistentes.
- Gestión de roles: define quiénes son los data stewards, custodios y propietarios del dato, y cuáles son sus responsabilidades.



Principios y mejores prácticas

Los principios que guían el diseño de una arquitectura de datos efectiva incluyen:

Centralización o federación del control de datos, según el tamaño y necesidades de la organización.

Documentación clara y accesible sobre flujos y estructuras de datos.

Escalabilidad para crecer sin comprometer el rendimiento.

Flexibilidad para adaptarse a nuevas tecnologías o requisitos del negocio.

Seguridad y privacidad por diseño.

Calidad de datos como prioridad, aplicando validaciones, deduplicación, y monitoreo.

DAMA-DMBOK: Marco de referencia

El **Data Management Body of Knowledge (DAMA-DMBOK)** es uno de los marcos de referencia más importantes y ampliamente aceptados a nivel mundial para la gestión integral de datos. Desarrollado por la asociación DAMA International, este cuerpo de conocimiento proporciona una estructura clara y detallada que permite a las organizaciones gestionar sus datos como un activo estratégico, promoviendo prácticas estandarizadas, sostenibles y escalables.

Este marco identifica **11 áreas clave de conocimiento**, todas interconectadas, que permiten comprender y estructurar adecuadamente una estrategia de datos efectiva:

Arquitectura de datos: define cómo se estructuran y conectan los sistemas de datos dentro de una organización.

Gobernanza de datos: establece políticas, estándares y roles claros para la toma de decisiones relacionadas con los datos.

Calidad de datos: asegura que los datos sean confiables, completos, exactos y relevantes para el negocio.

Seguridad de datos: protege los datos frente a accesos no autorizados, pérdida o uso indebido.

Modelado y diseño de datos: crea representaciones conceptuales, lógicas y físicas de los datos y sus relaciones.

Además, contempla otras áreas como almacenamiento, integración, operaciones, metadatos, gestión documental, entre otras.

Por lo tanto, DAMA-DMBOK es una **herramienta fundamental para comprender cómo los distintos componentes de la arquitectura de datos se alinean con buenas prácticas reconocidas internacionalmente**. Usarlo como guía permite establecer una base sólida sobre la cual construir soluciones robustas, adaptables y orientadas al valor del negocio.

Consideraciones clave en la arquitectura de datos

Para asegurar que una arquitectura de datos sea efectiva a largo plazo, deben considerarse:

Escalabilidad: ¿Soporta el crecimiento futuro en volumen y variedad de datos?

Flexibilidad: ¿Permite cambios sin rediseñar toda la solución?

Interoperabilidad: ¿Se integra con otras herramientas o plataformas?

Seguridad y privacidad: ¿Protege los datos sensibles y cumple con regulaciones?

Calidad de datos: ¿Existe control sobre la integridad, consistencia y disponibilidad?

Una arquitectura robusta permite a las organizaciones tomar decisiones basadas en datos confiables, seguros y bien estructurados.

Ciclo de vida del dato y arquitectura empresarial

Ciclo de vida del dato

El ciclo de vida del dato describe las etapas que atraviesa un dato desde su creación hasta su eliminación. La arquitectura de datos interviene en cada una de estas etapas asegurando que los datos sean gestionados adecuadamente:

1. Captura: Definición de formatos y validaciones desde el origen.
2. Almacenamiento: Elección del sistema adecuado según el tipo y uso del dato.
3. Procesamiento: Aplicación de reglas de negocio, integración y transformación.
4. Distribución: Acceso a usuarios y sistemas mediante capas de servicio (APIs, BI).
5. Consumo: Uso en análisis, reportes, modelos de IA.
6. Mantenimiento: Revisión, depuración, actualización.
7. Archivado o eliminación: Políticas de retención de datos y cumplimiento normativo.

Este enfoque permite comprender que la arquitectura no es estática, sino un componente vivo que se adapta continuamente al contexto tecnológico y organizacional.

Ejemplo práctico de arquitectura empresarial

Contexto: Una cadena de supermercados con múltiples sucursales físicas y ventas en línea.

Arquitectura de datos propuesta:

- Fuentes de datos: Punto de venta (POS), sitio web, aplicación móvil, inventario, redes sociales.
- Almacenamiento: Base de datos relacional para el stock e inventario, Data Lake en la nube para datos semiestructurados, Data Warehouse para consolidar información.
- Procesamiento: Proceso ETL nocturno que integra datos de ventas, inventario y logística.
- Acceso: Paneles BI en Power BI para el área comercial y consultas SQL para los analistas.
- Seguridad: Accesos controlados por rol, autenticación multifactor, encriptación en reposo.
- Gobernanza: Diccionario de datos, políticas de acceso y revisión de calidad de datos semanal.