

BASES DE DATOS PARA LA INGENIERÍA INDUSTRIAL

Fundamentos, diseño e implementación para Ingenieros Industriales



Ms. Julio César Álvarez Reyes

Bases de datos para la ingeniería industrial

Fundamentos, diseño e implementación para ingenieros industriales



Julio César Álvarez Reyes

<https://orcid.org/0000-0001-8465-8907>

Lima – Perú

2026

Bases de datos para la ingeniería industrial

Fundamentos, diseño e implementación para ingenieros industriales

© **Julio César Álvarez Reyes**
<https://orcid.org/0000-0001-8465-8907>

Editada por:

© Professionals On Line SAC. (FEPOL) - Fondo Editorial.
Dirección: Av. General Jose de San Martin nro. 790 dpto. 402 urb. el Carmen Lima - Lima - Pueblo Libre (magdalena vieja)
professionalsonline.net@gmail.com
Teléf. móvil: +51 999 140 920
Web: <https://professionalsonline.org/>

Primera edición digital: Febrero 2026
Libro digital disponible en <https://editorialfondo.com/>
Hecho el Depósito Legal en la Biblioteca Nacional del Perú N° 2026-01635
ISBN: 978-612-49279-9-7

Corrección de estilo: Luis Pablo Diaz Tito
luisp.diaz@upsjb.edu.pe / Tel. de contacto: +51 955 129 801

Diseño y Diagramación: Gráfica “imagen”
Corrección de estilo: Luis Pablo Diaz Tito
luisp.diaz@upsjb.edu.pe / Tel. de contacto: +51 955 129 801

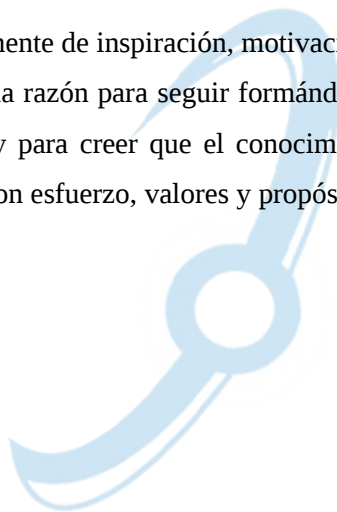
Libro resultado de Investigación y con revisión por pares doble ciego.
Sello editorial: Fondo Editorial (978-612-48981)

Este libro está dedicado a:

A mis estudiantes de Ingeniería Industrial, quienes con curiosidad, disciplina y espíritu crítico me recuerdan cada día que el conocimiento no es un destino, sino un proceso continuo de mejora.

A mis padres, quienes con su esfuerzo silencioso, valores firmes y apoyo incondicional sentaron las bases más importantes de mi formación personal y profesional.

A mis hijos y esposa, fuente permanente de inspiración, motivación y aprendizaje. En ustedes encuentro la razón para seguir formándome, para enseñar con responsabilidad y para creer que el conocimiento tiene sentido cuando se construye con esfuerzo, valores y propósito.



Prefacio

En un entorno industrial marcado por la globalización, la automatización y la acelerada transformación digital, la capacidad de convertir datos en conocimiento se ha consolidado como una competencia esencial del ingeniero industrial. Las organizaciones ya no compiten únicamente por eficiencia operativa, sino por su habilidad para capturar, estructurar y analizar información confiable que sustente decisiones oportunas y estratégicas.

Este libro ha sido concebido específicamente para estudiantes y profesionales de Ingeniería Industrial, con el propósito de abordar el diseño de bases de datos desde una perspectiva aplicada a los procesos productivos, logísticos y de gestión. Lejos de tratar las bases de datos como un componente meramente tecnológico, la obra las presenta como una infraestructura crítica para la mejora continua, la trazabilidad operativa y la gestión basada en datos.

Una base de datos bien diseñada trasciende la simple organización de información. Permite garantizar la consistencia de los registros, eliminar duplicidades, reducir la dependencia de hojas de cálculo dispersas y facilitar el acceso a información actualizada en tiempo real. Esto no solo contribuye al uso eficiente de los recursos —incluyendo la reducción del uso de papel—, sino que habilita la generación de reportes confiables que ofrecen una visión inmediata del desempeño organizacional, apoyando el control de procesos, la

detección temprana de desviaciones y la toma de decisiones fundamentadas.

A lo largo de esta obra se explora el ciclo completo de los datos en el contexto industrial: desde su generación en los procesos operativos hasta su estructuración en arquitecturas de información robustas que garantizan integridad, disponibilidad y trazabilidad. En un escenario donde la intuición resulta insuficiente frente a la complejidad de los sistemas productivos modernos, la calidad de los datos históricos se convierte en el insumo indispensable para la optimización, la analítica avanzada y la sostenibilidad organizacional.

El libro se estructura de manera progresiva y coherente. En una primera parte, se analizan los fundamentos conceptuales y el valor estratégico de los datos y de los sistemas de gestión de bases de datos en la industria. Posteriormente, se desarrolla el modelado conceptual, donde los procesos reales se traducen en entidades y relaciones que representan fielmente la operación. Finalmente, se profundiza en el diseño lógico y físico, transformando estos modelos en estructuras relacionales eficientes, normalizadas y preparadas para soportar las exigencias de sistemas industriales modernos.

Esta obra no pretende formar programadores ni especialistas en sintaxis SQL. Su objetivo es formar ingenieros industriales capaces de diseñar, evaluar y gestionar arquitecturas de datos alineadas con los procesos del negocio, entendiendo que una base de datos correctamente concebida es, en esencia, una extensión lógica de la

planta, del almacén y de la cadena de suministro. Dominar estos fundamentos ya no es una ventaja competitiva opcional, sino un requisito indispensable para liderar la industria del presente y del futuro.



Tabla de contenido

Parte I: Fundamentos y el rol de los datos en la industria	9
Capítulo I: La información como activo estratégico	9
Capítulo 2: Conceptos fundamentales de Bases de Datos para sistemas industriales	21
Capítulo 3: Levantamiento de Requerimientos y Planificación de la Base de Datos	26
Parte II: Diseño Conceptual. Comprendiendo los procesos	34
Capítulo 4: Modelado Diagrama Entidad-Relación (DER)	37
Parte III: Diseño Lógico y Normalización. Del modelo de negocio al modelo de datos.....	59
Capítulo 5: El Modelo Relacional	63
Parte IV: Modelo Físico de Datos. De la teoría a la infraestructura operativa.....	84
Capítulo 6: Especificación técnica y diccionario de datos.....	84
Capítulo 7: Índices y estrategias de optimización.....	88

Parte I: Fundamentos y el rol de los datos en la industria

Capítulo I: La información como activo estratégico

En la actualidad, las organizaciones industriales operan en un entorno caracterizado por la globalización, la acelerada transformación digital y la necesidad constante de mejorar su competitividad. En este escenario, la información ha dejado de ser un recurso secundario para convertirse en el *activo estratégico más valioso*, capaz de generar ventajas sostenibles y orientar la toma de decisiones en todos los niveles de la empresa.

Cuando se gestiona de manera adecuada, la información permite transformar datos dispersos en conocimiento accionable. Este conocimiento constituye la base para optimizar procesos, reducir costos, mejorar la calidad, innovar en productos y servicios, y responder con agilidad a las demandas del mercado.

Considerar la información como un activo estratégico implica reconocer que su valor no reside únicamente en su acumulación, sino en su uso *inteligente y contextualizado*. Hoy, este uso inteligente está impulsado por tecnologías como la *Inteligencia Artificial (IA)* y el *Machine Learning (ML)*, que permiten analizar grandes volúmenes de datos para predecir fallos en equipos, optimizar cadenas de suministro o personalizar la producción. Las empresas que logran integrar sus datos en plataformas de análisis avanzado, sistemas de gestión y herramientas de inteligencia de negocios pueden anticipar tendencias,

identificar oportunidades y minimizar riesgos de forma proactiva. En este sentido, la información se ha convertido en un recurso tan crítico como el capital financiero o humano, y su administración efectiva es clave para la sostenibilidad y el crecimiento organizacional.

1.1 Diferencia entre dato, información y conocimiento.

Comprender la denominada pirámide del conocimiento constituye el primer paso para gestionar la información como un activo estratégico en la industria. Este modelo describe la trazabilidad que existe desde el dato hasta el conocimiento y, al mismo tiempo, proporciona un marco conceptual para el diseño de sistemas de información capaces de capturar, procesar y transformar datos en soporte efectivo para la toma de decisiones.

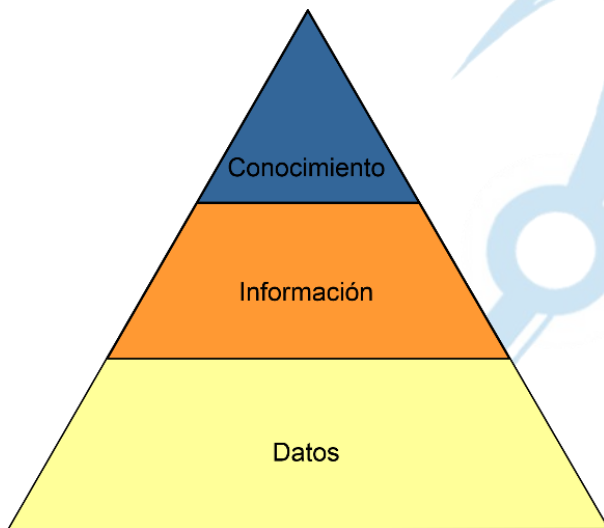


Figura 1: Pirámide del conocimiento

a. Dato

Es el registro elemental y objetivo de un hecho, valor o evento observable, capturado sin interpretación ni contexto propio. El dato representa la observación directa de la realidad operativa y constituye la materia prima sobre la cual se construyen los sistemas de información.

En los entornos industriales, los datos se generan de forma continua a partir de sistemas transaccionales, equipos de control, sensores y dispositivos de registro. Su valor es potencial y latente: por sí solos no permiten la toma de decisiones, pero al ser organizados y relacionados dentro de una base de datos, se convierten en el insumo fundamental para producir información confiable y conocimiento aplicable.

Ejemplo

El registro “2025-05-01 08:20:30 a. m.” capturado por un sistema biométrico de control de acceso al momento del ingreso de un operario a la planta de producción.



Figura 2: Lector biométrico de huella para marcación de asistencia

b. Información

La información surge cuando los datos son procesados, organizados y contextualizados, adquiriendo significado y utilidad para responder preguntas básicas del proceso: qué ocurrió, quién participó, cuándo y dónde.

Ejemplo

“El operario Julio Álvarez Reyes (ID: 6669), del área de ensamblado final, ingresó a la planta el 1 de mayo de 2025 a las 08:20:30, registrando un retraso de 20 minutos respecto a su horario de turno establecido (08:00:00).”

En este nivel, la base de datos permite estructurar y relacionar los datos para generar reportes operativos. Si este mismo evento se repite cuatro veces durante una semana para el mismo operario, la información

puede agregarse y transformarse en un indicador de desempeño, como un KPI de asistencia que evidencie un nivel de puntualidad del 20% en el período analizado.

c. Conocimiento

El conocimiento emerge de la síntesis de la información, combinada con la experiencia, el criterio profesional y la comprensión de las relaciones causa–efecto dentro del sistema productivo. En este nivel, la información deja de ser descriptiva y se convierte en soporte para decisiones estratégicas.

Ejemplo

Al analizar la base de datos de los últimos tres meses, el ingeniero jefe del área de ensamblado identifica que los retrasos de aproximadamente 20 minutos no son un fenómeno aislado, sino que afectan al 40% del personal que reside en la zona norte de la ciudad, especialmente los días miércoles y jueves.

A partir de este análisis, se genera el siguiente conocimiento:

“Los retrasos no obedecen a problemas de disciplina individual, sino a una saturación recurrente en la ruta de transporte público de la zona norte, causada por una feria local que opera esos días. Esta situación genera un cuello de botella en el inicio de la jornada, incrementando los costos por horas extras necesarias para compensar la producción perdida.”

Con base en este conocimiento, el ingeniero propone soluciones de nivel superior, como el ajuste escalonado de los horarios de ingreso del personal afectado, compensando la salida al final del turno, con el objetivo de mitigar el impacto del tráfico y mejorar la eficiencia global del proceso.

Cierre conceptual

Esta secuencia —dato, información y conocimiento— evidencia que el verdadero valor de una base de datos industrial no reside en el almacenamiento de registros, sino en su capacidad para preservar la memoria operativa de la organización y convertirla en decisiones informadas. Diseñar correctamente esta estructura es una de las responsabilidades centrales del ingeniero industrial en la gestión moderna de los procesos.

1.2 Ciclo de vida del dato en la cadena de suministro

En la ingeniería industrial tradicional, la cadena de suministro se visualizaba principalmente como un flujo lineal de materiales: desde la materia prima hasta el producto terminado en manos del cliente. Sin embargo, en la era digital, por cada unidad física que se mueve, se generan cientos de datos digitales que describen su estado, ubicación, condición y destino.

En este contexto, los datos se convierten en el insumo esencial para coordinar actividades, optimizar recursos y garantizar la trazabilidad de los productos desde el proveedor hasta el cliente final.

Comprender el ciclo de vida del dato permite visualizar cómo la información fluye y se transforma en conocimiento útil para la toma de decisiones estratégicas.

a. Generación y captura

Los datos se originan en múltiples puntos de la cadena: pedidos de clientes, registros de inventario, sensores de producción, sistemas de transporte, entre otros.

Ejemplo: un escáner de código de barras registra la entrada de materias primas en el almacén.

b. Almacenamiento y organización

Los datos capturados deben ser almacenados en sistemas confiables, como Bases de Datos relacionales o plataformas en la nube.

La organización adecuada garantiza su disponibilidad y evita redundancias.

Ejemplo: un ERP (Enterprise Resource Planning) centraliza la información de compras, producción y distribución.

c. Procesamiento y transformación en información

Los datos se procesan para generar reportes, indicadores y métricas relevantes.

Ejemplo: transformar registros de ventas en un informe de demanda semanal.

d. Análisis y generación de conocimiento

La información se interpreta para identificar patrones, tendencias y oportunidades de mejora.

Ejemplo: detectar que la demanda de un producto aumenta en temporadas específicas y ajustar la planificación de producción.

e. Uso en la toma de decisiones

El conocimiento derivado de los datos se aplica en decisiones tácticas y estratégicas: optimización de inventarios, negociación con proveedores, diseño de rutas logísticas, etc.

Ejemplo: decidir aumentar el stock de seguridad para evitar quiebres de inventario en campañas comerciales.

f. Retroalimentación y mejora continua

El ciclo se retroalimenta: las decisiones generan nuevos datos que reinician el proceso.

Ejemplo: la implementación de un nuevo sistema de transporte produce registros que permiten evaluar su eficiencia.

1.3 De las hojas de cálculo (Excel) a las Bases de Datos Relacionales.

Durante décadas, las hojas de cálculo -con Excel como herramienta más difundida- han sido el recurso principal para organizar y analizar datos en las empresas. Su facilidad de uso, flexibilidad y bajo costo las convirtieron en un estándar para tareas como control de inventarios, registro de ventas, planificación de producción, análisis financiero, etc. Sin embargo, a medida que las organizaciones crecieron y los volúmenes de datos se multiplicaron, las hojas de cálculo comenzaron a mostrar limitaciones frente a las necesidades de gestión moderna.

Limitaciones de las hojas de cálculo.

- a. Integridad de Datos: Carece de mecanismos robustos para garantizar la corrección y consistencia de los datos. Es propenso a los errores humanos (sobreescritura de fórmulas, duplicación de registros).
- b. Concurrencia Limitada: No maneja eficientemente el acceso simultáneo por múltiples usuarios, un requisito esencial en un entorno de planta.
- c. Escalabilidad Insuficiente: El rendimiento se degrada severamente con volúmenes de datos masivos (miles o

millones de registros), haciendo lentas e inestables las búsquedas y las operaciones de análisis.

Bases de Datos Relacionales

Las Bases de Datos Relacionales (BDR) surgieron como respuesta a estas limitaciones, ofreciendo un modelo estructurado y robusto para almacenar y gestionar grandes volúmenes de información. En ellas, los datos se organizan en tablas relacionadas entre sí mediante claves primarias y foráneas, lo que permite mantener la coherencia y evitar redundancias.

Ventajas frente a las hojas de cálculo:

- **Integridad y consistencia de los datos**

Las bases de datos garantizan que la información se mantenga coherente a lo largo del tiempo mediante reglas de integridad que evitan duplicidades, inconsistencias y registros inválidos. Esto es fundamental en procesos industriales donde los errores de datos impactan directamente en indicadores de gestión, control de calidad y trazabilidad operativa.

- **Concurrencia y trabajo multiusuario**

Permiten que múltiples usuarios y sistemas accedan simultáneamente a la información sin comprometer su consistencia. Esta característica es esencial en entornos productivos donde diversas áreas —producción,

mantenimiento, logística y gestión— interactúan sobre los mismos datos en tiempo real.

- **Seguridad y control de accesos**

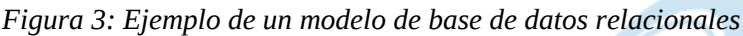
Las bases de datos incorporan mecanismos de autenticación y autorización que permiten definir distintos niveles de acceso a la información. Esto protege los datos críticos de la organización y asegura que cada usuario interactúe únicamente con la información necesaria para su función.

- **Escalabilidad y rendimiento**

Están diseñadas para crecer junto con la organización, soportando el incremento de volumen de datos, usuarios y transacciones sin degradar el desempeño. Esta capacidad es clave en industrias sometidas a procesos de expansión, automatización y transformación digital.

- **Trazabilidad y soporte para el análisis**

Las bases de datos facilitan el registro histórico de los procesos, permitiendo reconstruir eventos, analizar comportamientos y generar indicadores de desempeño. Esta trazabilidad constituye la base para la analítica industrial, el control de procesos y la mejora continua.



Capítulo 2: Conceptos fundamentales de Bases de Datos para sistemas industriales

El ingeniero industrial opera en la intersección entre los procesos tangibles y su representación digital. El Capítulo 2 establece los *conceptos fundamentales de las Bases de Datos* como los pilares para construir esa representación de manera fiel y útil. Su dominio es lo que diferencia un sistema de información que es una carga burocrática de uno que es un motor de mejora continua. Aquí, el lector no aprenderá a programar, sino a ejercer como *arquitecto de información industrial*: una competencia clave para diseñar sistemas que capturen la esencia de las operaciones, garantizando que los datos sean un reflejo confiable, eficiente y explotable de la industria.

2.1 Definición de Base de Datos, SGBD (Sistema Gestor de Bases de Datos).

Una base de datos es un conjunto estructurado de datos relacionados entre sí, diseñado para representar de manera coherente los procesos de una organización y garantizar la integridad, disponibilidad y trazabilidad de la información. En el contexto de la Ingeniería Industrial, la base de datos constituye la memoria operativa del sistema productivo, permitiendo registrar eventos, controlar procesos y analizar su desempeño a lo largo del tiempo.

El Sistema de Gestión de Bases de Datos (SGBD) es el software encargado de administrar dicha estructura, proporcionando los mecanismos necesarios para almacenar, consultar, actualizar y proteger la información. Más allá de su rol tecnológico, el SGBD actúa como un intermediario entre los procesos operativos y los sistemas de información que soportan la toma de decisiones.

Desde una perspectiva industrial, la importancia del SGBD no radica en sus características técnicas aisladas, sino en su capacidad para asegurar la consistencia de los datos, evitar redundancias, facilitar el acceso concurrente a la información y preservar el historial de los procesos. Un SGBD correctamente seleccionado y configurado permite que la base de datos crezca y se adapte a la complejidad de la organización sin perder confiabilidad ni desempeño.

Para el desarrollo de los casos prácticos y ejemplos técnicos en este libro, se ha seleccionado **MySQL** como el Sistema de Gestión de Bases de Datos (DBMS) de referencia.



Figura 4: DBMS líderes de la industria

2.2 Ventajas del uso de bases de datos en la gestión industrial

El uso de bases de datos en entornos industriales aporta beneficios que trascienden el ámbito tecnológico y se reflejan directamente en la gestión de los procesos y en la toma de decisiones. Estas ventajas pueden sintetizarse en los siguientes aspectos clave:

- Eficiencia operativa, al centralizar la información y eliminar la duplicidad de registros dispersos en hojas de cálculo o sistemas aislados.
- Confiabilidad y consistencia de los datos, reduciendo errores de captura y asegurando que los indicadores se basen en una única versión de la información.

- Trazabilidad de los procesos, mediante el registro histórico de eventos, estados y transacciones, lo que facilita la auditoría y el análisis causal.
- Soporte a la toma de decisiones, al permitir la generación de reportes e indicadores oportunos y alineados con los objetivos de la organización.
- Escalabilidad, posibilitando que los sistemas de información acompañen el crecimiento y la complejidad de las operaciones industriales.

Estas ventajas explican por qué las bases de datos no deben concebirse como un repositorio pasivo de información, sino como un componente estratégico de la infraestructura organizacional.

2.3 Arquitectura de bases de datos en entornos industriales

La arquitectura de una base de datos define la forma en que los datos se organizan, almacenan y gestionan para soportar los procesos de la organización. En entornos industriales, una arquitectura adecuada debe responder tanto a los requerimientos operativos diarios como a las necesidades de análisis y control a mediano y largo plazo.

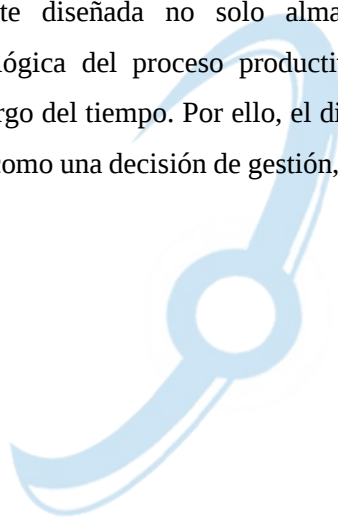
De manera general, la arquitectura de bases de datos puede analizarse en tres niveles: conceptual, lógico y físico. El nivel conceptual representa los procesos y entidades del negocio desde una perspectiva abstracta; el nivel lógico traduce esta representación en estructuras relacionales coherentes; y el nivel físico materializa el modelo en un

sistema concreto, considerando aspectos como tipos de datos, almacenamiento y rendimiento.

Para el ingeniero industrial, comprender esta arquitectura es fundamental, ya que cada decisión de diseño impacta directamente en la calidad de la información, en la eficiencia del sistema y en la capacidad de generar indicadores confiables. Una arquitectura mal concebida puede limitar la trazabilidad, dificultar el análisis histórico y comprometer la toma de decisiones, aun cuando se disponga de tecnología avanzada.

Implicancia para la gestión industrial

Una base de datos correctamente diseñada no solo almacena información, sino que refleja la lógica del proceso productivo y preserva su comportamiento a lo largo del tiempo. Por ello, el diseño de su arquitectura debe entenderse como una decisión de gestión, y no únicamente como una tarea técnica.



Capítulo 3: Levantamiento de Requerimientos y Planificación de la Base de Datos

Aquí es donde el ingeniero industrial despliega sus competencias clave: el análisis de procesos, la observación sistemática y la capacidad de traducir necesidades operativas caóticas en especificaciones técnicas claras. Un error en esta fase —un requerimiento omitido, un proceso malinterpretado— no se corrige con un parche en el código; se multiplica exponencialmente en costos de retrabajo, en datos corruptos y, en última instancia, en la desconfianza de los usuarios hacia el sistema.

La analogía es perfecta: un modelo de datos mal planificado es como una planta con flujos cruzados, cuellos de botella ocultos y almacenamiento inadecuado. Funcionará, pero a un costo exorbitante de ineficiencia, riesgo y frustración. Por el contrario, una base de datos bien planificada es como una planta lean: el "material" (la información) fluye de manera precisa, oportuna y sin desperdicios hacia los "puntos de consumo" (los reportes y los tomadores de decisiones).

3.1 La Importancia Crítica del Levantamiento de Requerimientos

El levantamiento de requerimientos es el proceso de descubrir, analizar, documentar y validar las necesidades de información y funcionalidad que deben cumplirse. Para el ingeniero industrial, este proceso es fundamental porque:

- Alinea el sistema con los procesos de negocio: No se diseña una base de datos para la base de datos. Se diseña para soportar y mejorar procesos específicos: la gestión de inventarios, el control de producción, el mantenimiento preventivo, el análisis de calidad. Sin un entendimiento profundo de estos procesos, el sistema será ajeno a la operación.
- Captura la perspectiva de todos los "usuarios" internos: El supervisor de planta necesita datos en tiempo real de paradas. El jefe de calidad necesita trazabilidad de lotes. El gerente financiero necesita costos estándar versus reales. El ingeniero debe ser el facilitador que integra estas visiones, a veces contradictorias, en un diseño coherente.
- Define el alcance y evita la "deriva": Sin requerimientos claros, el proyecto crece de manera descontrolada ("ya que estamos, agreguemos esto..."). Un alcance bien definido es el mapa que mantiene al proyecto en curso, dentro del presupuesto y el tiempo.
- Establece los criterios de éxito objetivos: ¿Cómo sabremos que el sistema funciona? Los requerimientos deben ser medibles y verificables. Ej.: "El sistema debe reducir el tiempo de generación del reporte de eficiencia de línea de 4 horas a menos de 15 minutos".

3.2 Lo Crítico de la planificación de la Base de Datos

La planificación de un proyecto de base de datos no es una actividad administrativa, sino un ejercicio de ingeniería que define la viabilidad, asigna responsabilidades y establece los hitos para medir el avance. Una planificación deficiente se traduce, invariablemente, en sobrecostos, retrasos y un sistema que no satisface las expectativas.

Esta sección proporciona un marco concreto, orientado a la gestión de tiempos, recursos humanos y económicos, estructurado en las fases clásicas del ciclo de vida del desarrollo de sistemas.

a. Fases del proyecto, entregables

Si bien existen diversos enfoques para el desarrollo de una base de datos, en este libro enfatizamos la planificación como una herramienta de control. Una base de datos sin un plan de ejecución carece de trazabilidad y dificulta la evaluación de sus etapas de avance.

Inicio y Definición

- Documento de visión y alcance.
- Acta de constitución del proyecto.

Levantamiento y Análisis

- Matriz de requerimientos (funcionales y no funcionales).

Diseño Conceptual y Lógico

- Modelo Conceptual (DER).

- Modelo Lógico (Esquema de tablas con PK/FK).

Diseño Físico e Implementación

- Tipos de datos.
- Índices.
- Restricciones.

Despliegue y Capacitación

- Manuales de usuario y administrador.
- Registros de capacitación.

Cierre y Soporte Inicial

- Informe Final de Proyecto.
- Plan de Soporte a Largo Plazo.

b. Cronograma estimado

La extensión del cronograma está sujeta a la complejidad de la arquitectura de datos, el volumen de migración de registros históricos y la interoperabilidad con otros sistemas. Para una gestión eficaz, se recomienda el uso de herramientas de control de proyectos como el Diagrama de Gantt, hojas de cálculo avanzadas o software especializado de gestión, permitiendo así visualizar las rutas críticas y el cumplimiento de hitos.

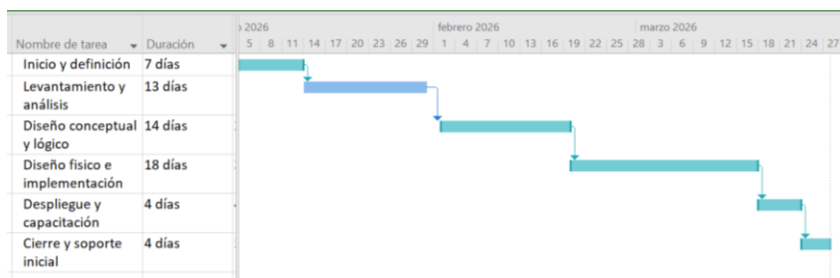


Figura 5: Cronograma de actividades

c. Recursos humanos y responsabilidades

El diseño de una base de datos no es un ejercicio puramente técnico o aislado; es, ante todo, un proyecto de gestión humana. La importancia de definir roles y responsabilidades desde el inicio radica en mitigar el riesgo de desconexión entre la realidad operativa y la solución tecnológica.

d. Recursos financieros

La planificación financiera de un proyecto de base de datos constituye un ejercicio estratégico que trasciende radicalmente la simple elaboración de la base de datos, costos de personal, adquisición de licencias o equipos. Su verdadera importancia radica en determinar la viabilidad técnica, la sostenibilidad operativa y el retorno de inversión a lo largo de todo su ciclo de vida.

Un presupuesto bien estructurado es una herramienta fundamental de gestión y comunicación. Permite realizar dos funciones críticas:

- **Prever el Costo Total de Propiedad:** Ir más allá del costo inicial para incluir todos los gastos recurrentes: mantenimiento del servidor, actualizaciones de software, soporte técnico, capacitación continua y la inevitable evolución del sistema. Subestimar el TCO es la causa principal del fracaso o abandono de proyectos a mediano plazo.
- **Justificar la Inversión con Argumentos Empresariales Sólidos:** Traducir las capacidades técnicas de la base de datos en beneficios tangibles para la alta dirección. Un modelo financiero claro debe contrastar la inversión frente a los ahorros proyectados (reducción de desperdicios, disminución de tiempos de parada, optimización de inventarios) y la creación de valor (mejora en la calidad de decisiones, agilidad en reportes, habilitación de analítica avanzada).

Resumen Parte I

Fundamentos y el rol de los datos en la industria

La Parte I ha establecido los fundamentos conceptuales necesarios para comprender el rol de las bases de datos en la Ingeniería Industrial, abordando los datos no como simples registros técnicos, sino como activos estratégicos que sustentan la gestión operativa, el control de procesos y la toma de decisiones organizacionales.

A lo largo de esta sección, se ha analizado la evolución desde el dato crudo hasta el conocimiento, destacando la importancia de estructurar, contextualizar y relacionar la información para generar valor. La pirámide del conocimiento permite entender que el verdadero impacto de los sistemas de información no reside en la acumulación de datos, sino en su correcta transformación en información confiable y, posteriormente, en conocimiento aplicable a la mejora continua.

Asimismo, se ha diferenciado claramente el uso de hojas de cálculo frente a las bases de datos relacionales, enfatizando las limitaciones operativas de las primeras en escenarios industriales complejos. Las bases de datos se presentan como el soporte estructural que garantiza integridad, trazabilidad, concurrencia y escalabilidad, características indispensables en entornos productivos, logísticos y de servicios.

Esta parte también ha introducido el rol del ingeniero industrial como arquitecto de la información, responsable de diseñar sistemas que representen fielmente los procesos reales de la organización. Desde esta perspectiva, el modelado de datos deja de ser una actividad meramente técnica para convertirse en una extensión del análisis de procesos, alineando la arquitectura de la información con los objetivos estratégicos de la empresa.

En conjunto, la Parte I sienta las bases conceptuales sobre las cuales se desarrollan las siguientes secciones del libro, preparando

al lector para abordar el modelado conceptual, lógico y físico de bases de datos con una visión sistémica, aplicada y orientada a la eficiencia operativa, la analítica industrial y la transformación digital.



Parte II: Diseño Conceptual. Comprendiendo los procesos

Para el ingeniero industrial, el modelado conceptual no es un ejercicio de abstracción teórica, sino un proceso de traducción fiel de la realidad operativa. El error más costoso y común es pretender diseñar una base de datos desde la abstracción de un escritorio, sin comprender el sistema físico que pretende representar. El verdadero Diagrama Entidad-Relación (E-R) comienza a trazarse en el piso de planta, en la observación del flujo de materiales, en el ritmo de las máquinas y en la lógica de las decisiones operativas.

No es posible definir entidades significativas —como `Lote`, `Equipo` o `Ruta de Producción`— ni relaciones críticas —como `se transforma-en`, `consume` o `genera`— sin antes haber comprendido a profundidad el proceso en su contexto real. Un modelo de datos es, en esencia, un espejo lógico del proceso productivo. Si el ingeniero no entiende cómo un material crudo se convierte en un producto terminado, cómo se dispara una alerta de calidad o cómo se planifica un mantenimiento correctivo, el sistema resultante será un conjunto de tablas vacías de significado operativo, incapaz de soportar la toma de decisiones.

Por ello, el modelado efectivo nace de una simbiosis indispensable: la fusión del conocimiento técnico y tácito de los procesos industriales con la disciplina de la abstracción y estructuración de datos. Este capítulo guía al lector en ese viaje de traducción, desde la observación

de la realidad concreta hasta la construcción de su representación conceptual rigurosa.

El levantamiento de requerimientos

La puerta de entrada a esta comprensión es una fase crítica y sistemática: el levantamiento de requerimientos. Esta etapa va mucho más allá de la recolección de deseos; es un proceso de descubrimiento estructurado donde el ingeniero debe actuar no solo como analista, sino como etnógrafo de la operación, un entrevistador hábil y un observador agudo.

Un diseño robusto y útil depende de la aplicación metódica de varias herramientas:

- a. Entrevistas estructuradas y semiestructuradas: Diálogos técnicos con actores clave —desde el operario en la línea y el supervisor de turno hasta el gerente de logística— para descubrir la información vital para cada rol, identificar los puntos de dolor existentes y entender los cuellos de botella que, con frecuencia, son de naturaleza informativa.
- b. Observación directa: El contraste indispensable entre lo dicho en las reuniones y lo hecho en la realidad. Recorrer la línea, seguir un lote, presenciar una parada de máquina. Esta práctica asegura que ninguna entidad, relación o atributo crítico —un gesto, una verificación manual, una señal no documentada— quede fuera del modelo.

- c. Revisión exhaustiva de documentación y formatos: La memoria escrita de la operación. Los formatos en uso (hojas de ruta, checklists, reportes de calidad en papel), los manuales de procedimiento, los estándares de trabajo y los reglamentos internos son minas de oro para identificar datos obligatorios, flujos de aprobación y reglas de negocio que deben estar codificadas en el sistema.
- d. Cuestionarios y encuestas técnicas: Herramientas complementarias para cuantificar y calificar aspectos específicos, como los volúmenes de transacción diarios, la frecuencia de actualización de los datos y la criticidad de cierta información, parámetros esenciales para el diseño posterior del modelo lógico y físico.

En síntesis, esta fase de levantamiento es el cimiento sobre el que se construye todo el edificio de datos. Un cimiento débil o incompleto condena a la estructura entera a la irrelevancia. Las siguientes secciones detallarán cómo transformar estos requerimientos, capturados de la voz viva de la operación, en un modelo conceptual formal, claro y listo para ser implementado.

Capítulo 4: Modelado Diagrama Entidad-Relación (DER)

El modelo entidad-relación constituye la base conceptual para el diseño de bases de datos y representa el primer paso formal para traducir los procesos de una organización en estructuras de información. En el ámbito de la Ingeniería Industrial, su importancia radica en la capacidad de capturar la lógica real de los procesos productivos, logísticos y de gestión, antes de cualquier decisión tecnológica.

El MER permite describir, de manera abstracta y comprensible, los elementos relevantes del sistema (entidades), sus características (atributos) y la forma en que se vinculan entre sí (relaciones). Al trabajar en este nivel, el ingeniero industrial se enfoca en comprender el negocio y sus reglas operativas, evitando aún consideraciones técnicas propias de la implementación.

Entidades

Una entidad representa un objeto, evento o concepto del mundo real sobre el cual la organización necesita almacenar información. En contextos industriales, las entidades suelen corresponder a componentes del proceso, como productos, órdenes de producción, máquinas, operarios, proveedores o registros de mantenimiento.

La correcta identificación de las entidades es una decisión crítica, ya que define el alcance y la coherencia del modelo conceptual. Una

entidad mal definida genera ambigüedades que se trasladan a los niveles lógico y físico del diseño.

Ejemplo

En un sistema de gestión de mantenimiento, pueden identificarse entidades como Equipo, Orden de mantenimiento y Técnico, cada una representando un elemento esencial del proceso operativo.

Atributos

Los atributos describen las características relevantes de una entidad y permiten diferenciar sus ocurrencias. Desde la perspectiva industrial, un atributo debe existir únicamente si aporta valor al control, análisis o gestión del proceso que representa.

Incluir atributos irrelevantes o redundantes incrementa la complejidad del modelo y dificulta su mantenimiento. Por ello, el ingeniero industrial debe evaluar cada atributo en función de su utilidad para la operación y la toma de decisiones.

Ejemplo

Para la entidad Equipo, atributos como código, ubicación, fecha de adquisición o estado operativo resultan relevantes; mientras que otros, sin impacto en el proceso, deben descartarse.

Claves

La clave primaria es el atributo (o conjunto de atributos) que identifica de manera única a cada ocurrencia de una entidad. En el modelo conceptual, su definición permite establecer relaciones claras y evitar ambigüedades en la representación del sistema.

En entornos industriales, se recomienda el uso de identificadores simples y estables, evitando atributos que puedan cambiar con el tiempo o que dependan del contexto operativo, como fechas o descripciones.

Relaciones

Las relaciones representan la forma en que las entidades interactúan entre sí dentro del proceso. Estas interacciones reflejan reglas del negocio, flujos operativos y dependencias funcionales que deben ser correctamente capturadas en el modelo conceptual.

Definir adecuadamente las relaciones y sus cardinalidades permite comprender cómo se comporta el sistema en la realidad y anticipar necesidades de información futuras, como el análisis de desempeño, la trazabilidad o la auditoría de procesos.

Ejemplo

Una Orden de mantenimiento puede estar asociada a un solo Equipo, pero un Equipo puede tener múltiples Órdenes de mantenimiento a lo largo del tiempo, reflejando una relación uno a muchos.

Implicancia para la gestión industrial

El modelo entidad–relación no es un ejercicio gráfico ni un requisito académico aislado; es una herramienta fundamental para asegurar que la base de datos represente fielmente los procesos reales de la organización. Un MER bien construido facilita el diseño lógico, reduce errores posteriores y garantiza que la información almacenada sea consistente, trazable y útil para la toma de decisiones.

Desde esta perspectiva, el ingeniero industrial actúa como traductor entre el proceso operativo y la arquitectura de datos, sentando las bases para sistemas de información robustos y alineados con los objetivos de la organización.

4.1 Cardinalidades y restricciones en el Modelo Entidad–Relación

Las cardinalidades y restricciones permiten precisar cómo interactúan las entidades dentro de un proceso y bajo qué condiciones dichas interacciones son válidas. En el modelo entidad–relación, estos elementos no son detalles técnicos, sino la formalización de las reglas del negocio que rigen el funcionamiento del sistema productivo.

Definir correctamente las cardinalidades y restricciones es esencial para garantizar que la base de datos represente fielmente la realidad operativa y preserve la integridad de la información a lo largo del tiempo.

Cardinalidades

La cardinalidad expresa el número de ocurrencias de una entidad que pueden asociarse con una ocurrencia de otra entidad. Las relaciones más comunes son uno a uno, uno a muchos y muchos a muchos, cada una con implicancias distintas en el diseño posterior del modelo lógico.

Desde la perspectiva de la Ingeniería Industrial, la correcta identificación de la cardinalidad permite comprender la dinámica del proceso y anticipar el volumen y la frecuencia de los registros.

Ejemplo

En un sistema de gestión de producción, una Orden de producción puede generar múltiples Registros de avance, mientras que cada Registro de avance corresponde a una única Orden de producción. Esta relación uno a muchos refleja el seguimiento progresivo del proceso productivo.

Una definición incorrecta de la cardinalidad puede limitar la capacidad de análisis histórico o generar inconsistencias en los reportes operativos.

Restricciones de participación

Las restricciones de participación indican si la participación de una entidad en una relación es obligatoria u opcional. Este aspecto resulta clave para modelar correctamente los procesos industriales, ya que

muchos eventos no pueden existir sin un contexto operativo específico.

Ejemplo

Una Orden de mantenimiento debe estar asociada obligatoriamente a un Equipo, ya que no tiene sentido operativo registrar una orden sin un activo físico sobre el cual intervenir. En cambio, un Equipo puede no tener órdenes de mantenimiento registradas en un período determinado.

Estas restricciones evitan registros incompletos y refuerzan la coherencia del modelo conceptual.

Restricciones de negocio

Las restricciones de negocio representan reglas específicas del proceso que no siempre pueden expresarse únicamente mediante cardinalidades. Estas reglas derivan de políticas internas, normativas, criterios de control o prácticas operativas propias de la organización.

Ejemplo

En un sistema de control de asistencia, una regla de negocio puede establecer que un operario no puede registrar más de una marcación de ingreso para un mismo turno. Esta restricción no define cuántas entidades se relacionan, sino cómo deben comportarse los datos en el tiempo.

Incorporar estas restricciones desde el modelo conceptual permite anticipar validaciones necesarias y reduce la probabilidad de errores en la implementación.

Implicancia para la gestión industrial

Las cardinalidades y restricciones convierten al modelo entidad–relación en una representación fiel de las reglas que gobiernan los procesos industriales. Al definir las correctamente, el ingeniero industrial asegura que la base de datos no solo almacene información, sino que imponga disciplina operativa y preserve la lógica del negocio.

Un modelo conceptual que omite o simplifica estas reglas genera bases de datos frágiles, dependientes de controles externos y propensas a inconsistencias. Por el contrario, un MER bien definido facilita el diseño lógico, mejora la calidad de la información y fortalece la toma de decisiones basada en datos confiables.

4.2 Errores comunes en el modelado conceptual

En el modelado conceptual de bases de datos aplicadas a entornos industriales, ciertos errores se repiten con frecuencia y suelen pasar inadvertidos en las primeras etapas del diseño. Aunque puedan parecer decisiones menores, estos errores se trasladan a los niveles lógico y físico, afectando la integridad de la información, la trazabilidad de los procesos y la confiabilidad de los indicadores de gestión.

Identificar y evitar estos errores es una competencia fundamental del ingeniero industrial responsable de traducir procesos reales en arquitecturas de datos sólidas.

a. Confundir procesos con entidades

Uno de los errores más frecuentes consiste en modelar procesos o actividades como entidades, en lugar de identificar correctamente los objetos o eventos sobre los cuales se requiere almacenar información.

Ejemplo

- Modelar Producción como una entidad, cuando en realidad corresponde a un proceso. En este caso, las entidades relevantes podrían ser Orden de producción, Producto o Registro de producción.
- Este error genera modelos abstractos poco operativos y dificulta la posterior definición de atributos y relaciones.
- Crear entidades excesivamente generales o ambiguas

Definir entidades con nombres genéricos, como Datos, Registro o Información, conduce a modelos imprecisos que no representan fielmente la realidad industrial.

Una entidad debe tener un significado claro y delimitado dentro del proceso. Cuando una entidad agrupa conceptos distintos, se pierde trazabilidad y se incrementa la probabilidad de inconsistencias.

b. Incluir atributos irrelevantes o sin propósito operativo

Otro error común es incorporar atributos que no aportan valor al control del proceso ni a la toma de decisiones. Esto suele ocurrir cuando no se analiza previamente para qué será utilizada la información.

Ejemplo

Incluir el atributo estado civil de un operario en un sistema cuyo objetivo es analizar la eficiencia de máquinas introduce ruido informativo y complejidad innecesaria en el modelo.

c. Omitir reglas del negocio en el modelo conceptual

El modelo conceptual no debe limitarse a representar entidades y relaciones, sino que debe reflejar las reglas que gobiernan el proceso. Ignorar estas restricciones provoca que la base de datos acepte registros que no tienen sentido operativo.

Ejemplo

Permitir el registro de una Orden de mantenimiento sin asociarla a un Equipo impide la trazabilidad y compromete cualquier análisis posterior.

d. Definir incorrectamente las cardinalidades

Una cardinalidad mal definida distorsiona la representación del proceso y limita el análisis de la información.

Ejemplo

Modelar la relación entre Equipo y Orden de mantenimiento como uno a uno, cuando en la práctica un equipo puede tener múltiples órdenes a lo largo del tiempo, impide registrar el historial completo de mantenimiento.

Implicancia para la gestión industrial

Los errores en el modelado conceptual no son fallas gráficas, sino decisiones que impactan directamente en la calidad de la información y en la capacidad de la organización para aprender de sus propios procesos. Un modelo conceptual sólido permite construir bases de datos coherentes, auditables y orientadas a la mejora continua.

El ingeniero industrial, al comprender la lógica operativa del sistema, desempeña un rol clave en la prevención de estos errores, asegurando que la arquitectura de datos refleje fielmente la realidad y sirva como soporte efectivo para la toma de decisiones.

4.3 Reglas básicas para la construcción de un DER

- Cada entidad debe tener al menos un atributo clave.
- Los atributos deben ser significativos.
- Las relaciones deben representar hechos reales del sistema.
- Las cardinalidades deben definirse con base en reglas del negocio.
- Evitar redundancias conceptuales.

4.4 Buenas prácticas para el modelado conceptual

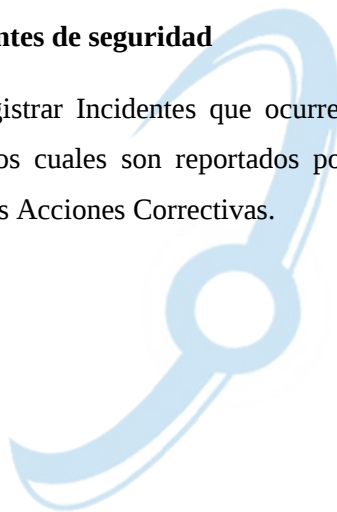
- Analizar primero el proceso antes de modelar datos.
- Utilizar nombres claros y consistentes.
- Documentar supuestos y reglas del negocio.
- Mantener el modelo simple pero completo.

4.5 Ejemplos aplicados para procesos industriales

Los diagramas de ejemplos del modelo conceptual de datos serán creados utilizando draw.io, una herramienta gráfica de diagramación versátil y ampliamente utilizada para la representación visual de diseños a nivel de análisis.

a. Modelo de registro incidentes de seguridad

El objetivo de este modelo es registrar Incidentes que ocurren en diferentes Áreas de la empresa, los cuales son reportados por un Trabajador y derivan en una o varias Acciones Correctivas.



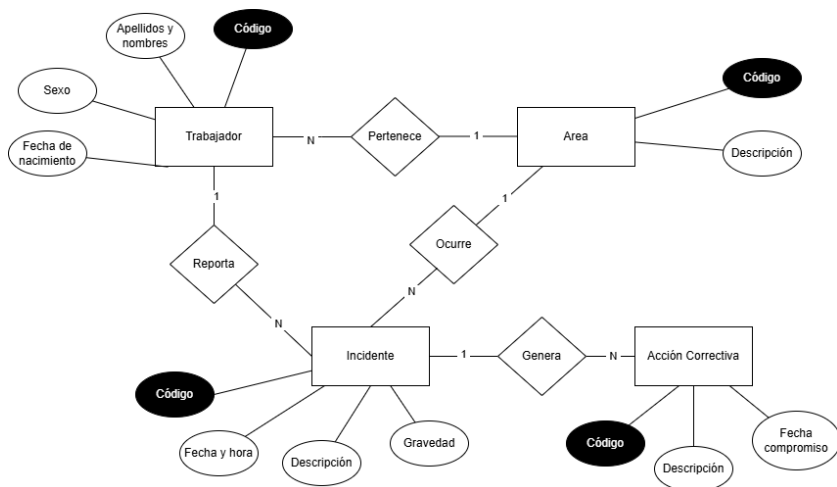


Figura 6: Modelo conceptual registro de incidentes de seguridad

El modelo se compone de cuatro entidades principales:

- **Trabajador:** Representa al personal de la empresa. Identificado por un Código, almacena sus nombres, apellidos, sexo y fecha de nacimiento.
- **Área:** Representa los departamentos o secciones de la empresa (Ej. Almacén, Producción). Se identifica por un Código y una Descripción.
- **Incidente:** Es el evento central del modelo. Registra Cuándo (Fecha y hora), Qué pasó (Descripción) y Qué tan grave fue (Gravedad).
- **Acción Correctiva:** Son las tareas diseñadas para solucionar o prevenir la repetición del incidente. Incluye una Descripción y una Fecha de compromiso.

b. Modelo de control de producción por órdenes de trabajo

Este modelo conceptual está diseñado para el control de la producción industrial. Se centra en cómo una Orden de Trabajo coordina los recursos (empleados, máquinas, materiales) para transformar insumos en productos terminados.

La entidad central es la Orden de Trabajo, funciona como el eje que conecta todas las demás operaciones de la planta. Actúa como el registro que vincula quién, con qué y cómo se produce algo.

Este modelo de producción tiene relaciones y atributos específicos para el seguimiento temporal y operativo de las Órdenes de Trabajo.

- Empleado: El capital humano. Se registra su Código, Apellidos y Nombres, Sexo, Fecha de nacimiento y su Especialidad (clave para saber qué tareas puede realizar).
- Material: Los insumos necesarios. Incluye Código, Descripción, Unidad de medida y el Stock disponible.
- Máquina: La maquinaria utilizada. Se identifica por su Código, Nombre, Tipo y su Estado (ej. operativa, en mantenimiento).
- Proceso: Las etapas técnicas de fabricación. Define el Código, la Descripción de la tarea y un Tiempo estimado.
- Producto: El resultado final de la producción. Posee Código, Nombre, Descripción y un Tiempo estimado de fabricación total.

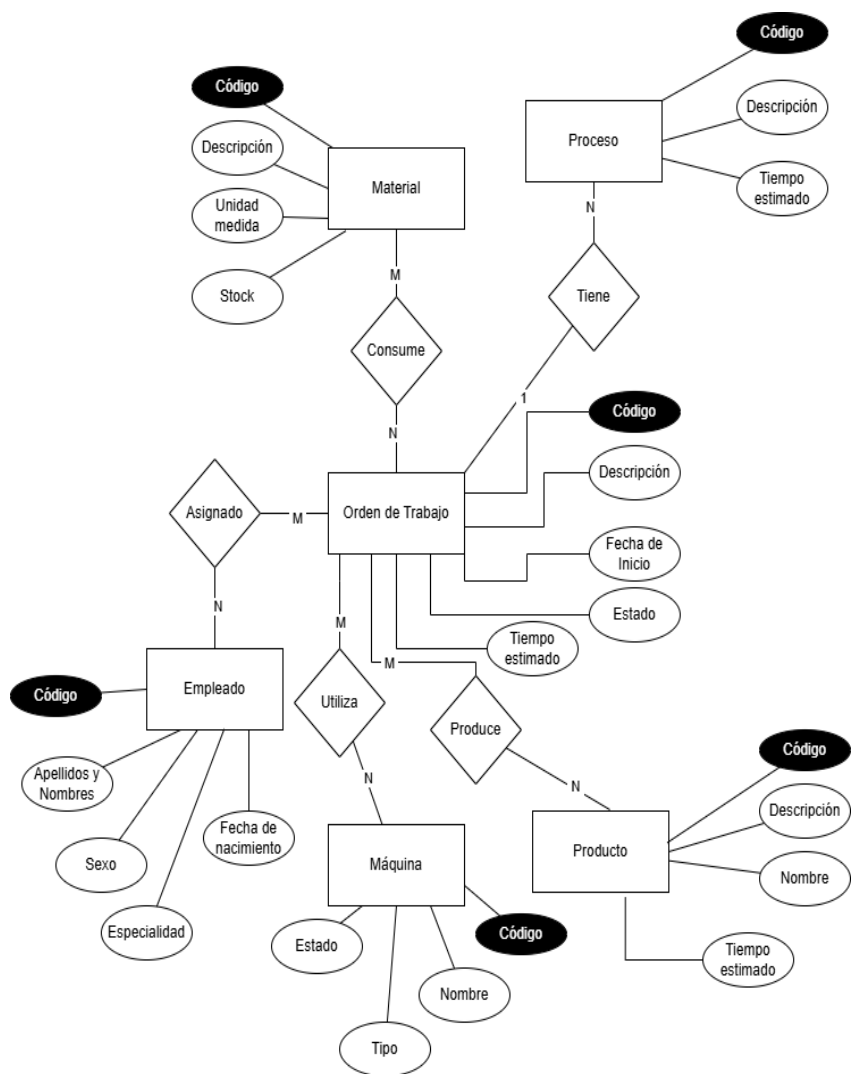


Figura 7: Modelo conceptual control de producción por órdenes de trabajo.

c. Modelo de un sistema de registro de fallos y mantenimiento

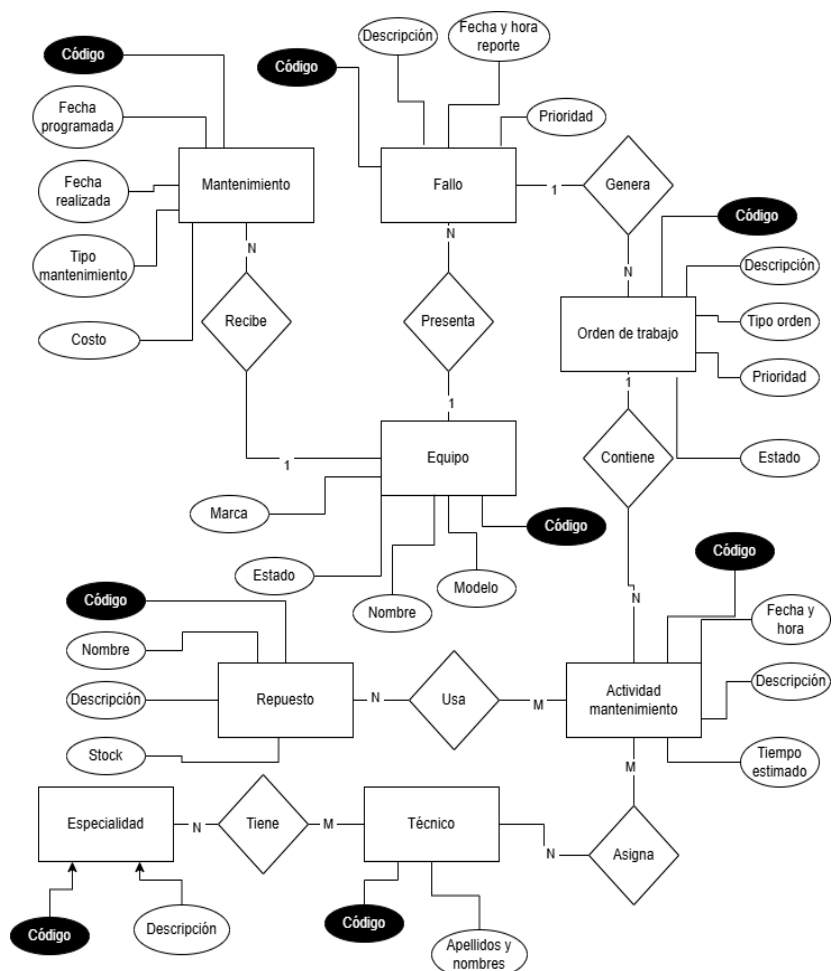


Figura 8: Modelo conceptual registro de fallos y mantenimiento

Este modelo conceptual está diseñado para la Gestión de Mantenimiento de Equipos y el control de activos en un entorno

industrial. El diagrama integra el ciclo completo desde que se detecta un problema hasta que se ejecuta la solución técnica.

El activo central es el equipo, que es el nexo entre las fallas detectadas y las tareas de mantenimiento.

Dentro de sus atributos se considera el estado, que puede ser operativo, fuera de servicio, con fallos, etc.

El modelo describe un flujo de respuesta ante incidentes:

- Fallo: Un Equipo puede presentar muchos (N) fallos. Cada fallo registra su Descripción, Fecha y hora de reporte y su Prioridad (Urgente, Alta, Media, Baja).
- Generación de Orden: Un Fallo genera 1 Orden de trabajo específica para su resolución.
- Estructura de la Orden: Una Orden de trabajo contiene muchas (N) Actividades de mantenimiento. Esto permite desglosar una reparación compleja en pasos más sencillos, cada uno con su propio Tiempo estimado y Descripción.

Para que el mantenimiento ocurra, el modelo gestiona los recursos humanos y materiales:

- Personal Técnico: Los Técnicos (identificados por Código y Nombres) se asignan a las actividades. La relación es N:M (muchos a muchos) entre Técnicos y Actividades, lo que

significa que varios técnicos pueden colaborar en una tarea y un técnico puede tener varias tareas asignadas.

- Especialización: Los técnicos tienen una Especialidad (ej. Mecánica, Electricidad).
- Repuestos: Las actividades de mantenimiento usan (M: N) Repuestos. Se controla el Stock, el Nombre y la Descripción de cada pieza utilizada para asegurar la trazabilidad del inventario.

Finalmente, el modelo permite el seguimiento administrativo:

- Mantenimiento: La entidad Mantenimiento registra el resultado final de la intervención.
- Trazabilidad temporal: Almacena la Fecha programada vs. la Fecha realizada.
- Análisis Financiero: Incluye un atributo de Costo, fundamental para calcular el gasto por equipo y determinar si es más rentable reparar o sustituir un activo.

4.6 Ejercicios propuestos

A continuación, se presentan diversos escenarios del entorno industrial y corporativo diseñados para poner a prueba su capacidad de abstracción y modelado conceptual. El objetivo de estos ejercicios es identificar con precisión las entidades, sus atributos clave y las reglas de negocio (cardinalidades) que rigen cada proceso.

a. Gestión de inventarios y almacenes

Caso de estudio: Una planta industrial necesita un sistema para controlar su inventario de materias primas, productos en proceso y productos terminados. Se requiere registrar las transacciones de entrada y salida, ubicaciones físicas (estanterías, pasillos) y los niveles de stock mínimo y máximo.

Se pide:

- Modelar productos (con identificación, nombre, categoría, y si es perecedero o no).
- Controlar lotes (número de lote, fecha de fabricación, fecha de caducidad).
- Registrar movimientos (entrada/salida, cantidad, fecha, responsable).
- Gestionar ubicaciones (código de almacén, zona, estante).

b. Control de calidad y pruebas de producto

Caso de estudio: Un laboratorio de control de calidad debe registrar los resultados de las pruebas realizadas a los productos terminados y a las materias primas recibidas de proveedores.

Se pide:

- Modelar proveedores, materiales y productos.
- Registrar tipos de prueba (ej.: resistencia, durabilidad) con especificaciones (límite inferior, límite superior).
- Capturar resultados de pruebas (valor, fecha, inspector, estado: aprobado/rechazado).
- Relacionar pruebas con lotes específicos.

c. Planificación de la producción y asignación de recursos

Caso de estudio: Una fábrica con múltiples órdenes de producción requiere asignar recursos (máquinas, personal) a tareas específicas, considerando disponibilidad y habilidades.

Se pide:

- Modelar órdenes de producción (prioridad, fecha inicio, fecha fin).
- Gestionar recursos (máquinas: tipo, capacidad; personal: habilidades, turnos).

- Registrar tareas (duración estimada, estado, secuencia dentro de la orden).
- Controlar la asignación de recursos a tareas (fecha, horas utilizadas).

d. Seguimiento de la cadena de suministro y logística

Caso de estudio: Una empresa de distribución necesita rastrear el flujo de productos desde los proveedores hasta los clientes, incluyendo transporte y estados de entrega.

Se pide:

- Modelar proveedores, clientes, pedidos y envíos.
- Registrar vehículos (placa, capacidad, conductor asignado).
- Controlar rutas (origen, destino, distancia estimada).
- Seguir el estado de cada envío (en preparación, en tránsito, entregado).

e. Gestión energética y sostenibilidad

Caso de estudio: Una planta industrial quiere monitorizar el consumo de recursos (energía, agua, gas) y sus emisiones, para mejorar la eficiencia y cumplir normas ambientales.

Se pide:

- Modelar equipos consumidores (máquinas, líneas de producción).
- Registrar mediciones (tipo de recurso, valor, fecha-hora, medidor utilizado).
- Gestionar objetivos de reducción (meta, período, responsable).
- Relacionar consumo con producción (ej.: kWh por unidad fabricada).

Resumen Parte II

Modelado conceptual de datos para procesos industriales

La Parte II ha desarrollado el modelado conceptual como una etapa crítica en el diseño de bases de datos aplicadas a la Ingeniería Industrial. En esta fase, el énfasis no se sitúa en la tecnología ni en la implementación, sino en la correcta representación de la realidad operativa de la organización, capturando sus procesos, reglas de negocio y relaciones fundamentales.

A través del modelo entidad-relación, se ha establecido un lenguaje formal que permite abstraer los procesos industriales y traducirlos en estructuras lógicas comprensibles, coherentes y verificables. Las entidades, atributos y relaciones se presentan como la expresión estructurada de objetos, eventos y actores que intervienen en sistemas productivos, logísticos, administrativos y de soporte.

Esta parte ha profundizado en la correcta identificación de entidades relevantes, evitando la confusión entre procesos y objetos persistentes, y destacando la importancia de definir atributos con propósito operativo. Asimismo, se ha enfatizado el rol de las cardinalidades y restricciones como mecanismos para incorporar las reglas del negocio directamente en el modelo de datos, garantizando consistencia y trazabilidad desde las primeras etapas del diseño.

Un aporte central de esta sección es la identificación de errores comunes en el modelado conceptual, los cuales suelen tener su origen en una comprensión incompleta del proceso o en decisiones de diseño apresuradas. Reconocer estos errores permite prevenir inconsistencias que, de trasladarse a etapas posteriores, afectarían la calidad de la información y la confiabilidad de los sistemas.

En conjunto, la Parte II consolida el modelado conceptual como una actividad estratégica para el ingeniero industrial, en la que el conocimiento del proceso es tan importante como la técnica de modelado. Los conceptos desarrollados en esta sección constituyen la base indispensable para la transformación posterior del modelo entidad-relación en estructuras relacionales robustas, orientadas al análisis y a la mejora continua.

Parte III: Diseño lógico y normalización. Del modelo de negocio al modelo de datos

Capítulo 5. Transformación del modelo entidad–relación al modelo relacional.

La transformación del modelo entidad–relación al modelo relacional constituye el puente entre la comprensión conceptual del proceso y la implementación estructurada de la base de datos. En esta etapa, las entidades, atributos y relaciones definidos a nivel conceptual se traducen en tablas, campos y claves, preservando la lógica del negocio y garantizando la integridad de la información.

Para el ingeniero industrial, este proceso no es una conversión mecánica, sino una decisión de diseño que impacta directamente en la trazabilidad de los procesos, la calidad de los datos y la capacidad de análisis posterior.

5.1 Conversión de entidades en tablas

Cada entidad identificada en el modelo conceptual se transforma en una tabla del modelo relacional. Los atributos de la entidad pasan a ser columnas de la tabla, y la clave primaria definida en el modelo conceptual se mantiene como identificador único de cada registro.

Ejemplo

La entidad Equipo se convierte en la tabla equipo, con atributos como id_equipo, descripcion, ubicacion y estado_operativo.

Esta transformación debe preservar el significado de la entidad y evitar la incorporación de atributos que no aporten valor al control del proceso.

5.2 Definición de claves primarias

La clave primaria garantiza la unicidad de cada registro y permite establecer relaciones coherentes entre tablas. En entornos industriales, se recomienda el uso de identificadores simples y estables, independientes de atributos operativos que puedan cambiar con el tiempo.

Una clave primaria mal definida compromete la integridad referencial y dificulta la gestión de históricos y estados del proceso.

5.3 Transformación de relaciones uno a muchos

Las relaciones uno a muchos se implementa incorporando la clave primaria de la entidad del lado “uno” como clave foránea en la tabla correspondiente al lado “muchos”.

Ejemplo

Una relación uno a muchos entre Equipo y Orden de mantenimiento se implementa añadiendo el campo `id_equipo` como clave foránea en la tabla `Orden_mantenimiento`.

Este enfoque permite registrar múltiples eventos asociados a una misma entidad y conservar el historial completo del proceso.

5.4 Transformación de relaciones muchos a muchos

Las relaciones muchos a muchos requieren la creación de una tabla intermedia, conocida como tabla asociativa, que contenga las claves primarias de las entidades relacionadas. Esta tabla puede incluir atributos propios que describan la interacción entre las entidades.

Ejemplo

La relación entre Técnico y Orden de mantenimiento se implementa mediante una tabla `Tecnico_orden`, que contiene los campos `id_tecnico`, `id_orden` y, eventualmente, atributos como `horas_trabajadas` o `fecha_intervencion`.

Este diseño permite modelar correctamente la participación de múltiples actores en un mismo proceso.

5.5 Implementación de restricciones e integridad referencial

Las restricciones definidas en el modelo conceptual se trasladan al modelo relacional mediante claves foráneas, restricciones de unicidad y reglas de obligatoriedad. Estas restricciones aseguran que la base de datos no permita registros inconsistentes o carentes de sentido operativo.

Incorporar estas reglas en la estructura de la base de datos reduce la dependencia de controles externos y fortalece la confiabilidad de la información.

Implicancia para la gestión industrial

La correcta transformación del modelo entidad–relación al modelo relacional garantiza que la base de datos refleje fielmente la lógica del proceso industrial y preserve su comportamiento a lo largo del tiempo. Un diseño relacional coherente facilita la generación de reportes, el análisis histórico y la toma de decisiones basada en datos confiables.

En esta etapa, el ingeniero industrial consolida su rol como arquitecto de la información, asegurando que las decisiones conceptuales se materialicen en estructuras de datos robustas, escalables y alineadas con los objetivos de la organización.



Capítulo 5: El Modelo Relacional

En las partes anteriores de este libro, usted ha dominado el arte de capturar la esencia de los problemas de la ingeniería industrial mediante modelos conceptuales. Ha aprendido a identificar entidades como Órdenes de Trabajo, Incidentes de Seguridad o Equipos en Mantenimiento, y a definir las relaciones que existen entre ellas en el lenguaje claro y visual de los diagramas entidad-relación.

Sin embargo, un modelo conceptual es como los planos arquitectónicos de una planta industrial: muestran espacios, flujos y relaciones, pero no especifican el tipo de concreto, el calibre de los cables eléctricos o el diámetro exacto de las tuberías. El diseño lógico relacional constituye precisamente esa especificación técnica para nuestra base de datos. En este capítulo, aprenderemos las reglas de transformación sistemática que convierten nuestro modelo ER en un esquema de tablas listo para ser implementado en cualquier sistema gestor de Bases de Datos (SGBD) relacional como MySQL, PostgreSQL u Oracle.

Para el ingeniero industrial, esta transición no es meramente técnica; es el momento en el que los conceptos de proceso, control y gestión se materializan en una estructura de datos que permitirá la automatización, el análisis y la trazabilidad de la información operativa.

Para el diseño y creación de los modelos lógicos de datos se utilizará MySQL Workbench. Esta herramienta permite materializar el esquema conceptual en un diagrama visual estandarizado, generar el script SQL de creación de tablas de forma automática y realizar ingeniería inversa, lo que la hace idónea para el ciclo de desarrollo propuesto.

5.1 Reglas fundamentales de transformación.

a. Transformación de entidades fuertes

Cada entidad fuerte en el modelo conceptual se convierte en una tabla en el esquema relacional.

- Los atributos de la entidad se convierten en columnas de la tabla.
- El identificador principal (clave primaria) se convierte en la clave primaria (PK) de la tabla.

Ejemplo: Tomando como referencia el ejemplo del modelo conceptual de los incidentes de trabajo.

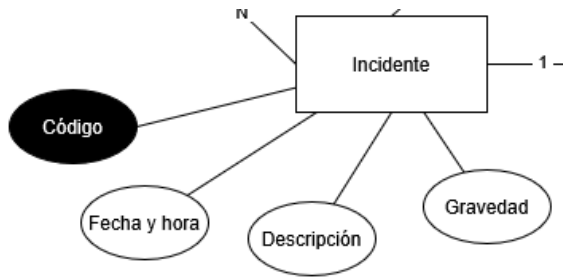


Fig. 9. Modelado conceptual de la entidad “Incidente”

Para alinear el diseño con las mejores prácticas en ingeniería de Bases de Datos, se aplicarán las siguientes convenciones de nomenclatura:

- Normalización de nombres: Las entidades y atributos no contendrán espacios, tildes ni caracteres especiales.
- Prefijos estandarizados: Se utilizará el prefijo `id_` para todas las claves primarias y foráneas.
- Consistencia gramatical: Los nombres de tablas y campos se expresarán en minúsculas y en singular.

Estas reglas garantizan la portabilidad del esquema, evitan conflictos con sistemas gestores de Bases de Datos sensibles a mayúsculas / minúsculas y mejoran la legibilidad del código SQL.

Aplicando estas convenciones, la entidad descrita se materializa en el siguiente esquema de tabla:

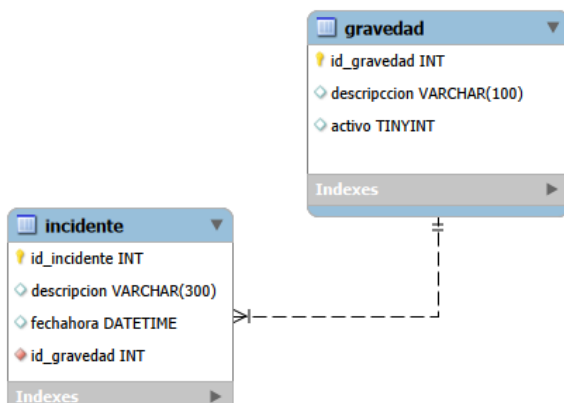


Fig. 10. Modelado lógico de la tabla “incidente” y “gravedad”

En el diagrama presentado, las claves primarias se representan mediante el icono de una llave, mientras que las claves foráneas se identifican con un rombo de color naranja.

Respecto al modelado de datos, durante la fase conceptual se definió el atributo "gravedad". Sin embargo, tras un análisis más detallado para la implementación del modelo lógico, se determinó que los valores de este atributo son limitados y predefinidos. Por ello, se optó por normalizarlo en una tabla independiente, mejorando así la integridad y eficiencia de la base de datos. Este mismo criterio aplica para otros atributos con características similares, como el tipo de sangre, el país, la especialidad de un empleado, entre otros, que también pueden ser modelados como entidades propias.

b. Transformación de atributos compuestos

Los atributos compuestos se descomponen en atributos simples (columnas atómicas).

- En el modelo relacional, todos los atributos deben ser atómicos (indivisibles).
- Esto cumple con el requisito fundamental de la Primera Forma Normal (1FN).

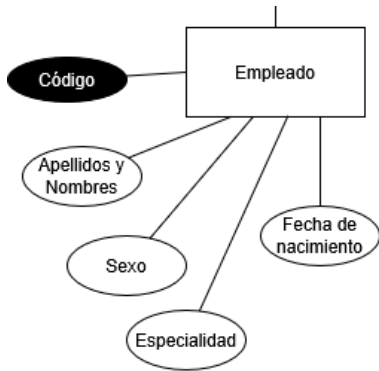


Fig. 11. Modelado conceptual de la entidad "Empleado".

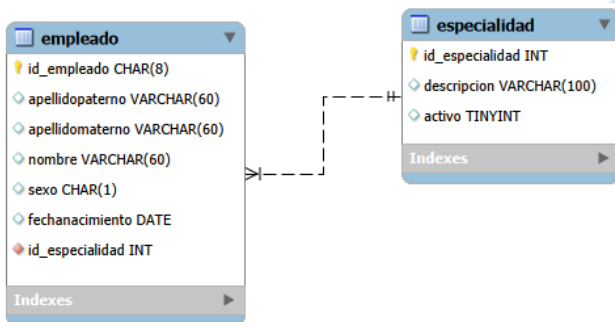


Fig. 12. Modelado lógico de las tablas "empleado" y "especialidad".

En el modelo se implementó una normalización de los atributos para garantizar la precisión y consistencia de los datos. En primer lugar, el atributo compuesto "Apellidos y Nombres" fue desglosado en tres campos independientes: "apellidopaterno", "apellidomaterno" y "nombre". Esta descomposición permite una identificación unívoca de cada trabajador. Asimismo, al atributo "especialidad" se le aplicó el mismo tratamiento de normalización previamente utilizado para el atributo "gravedad", definiéndolo como un catálogo cerrado de valores predeterminados para asegurar la integridad referencial y la estandarización de la información.

La clave primaria `id_empleado` utiliza el formato CHAR(8), ya que está destinada a almacenar el DNI del empleado. Este es un caso de uso específico en el diseño de Bases de Datos: emplear un identificador natural (uno que ya existe en el mundo real) en lugar de un identificador artificial (como un número auto incremental). Esta estrategia es aplicable cuando el identificador natural cumple con los requisitos de una clave primaria: es único, inmutable y de longitud conocida. Otros ejemplos comunes son el RUC de una empresa, la matrícula de un vehículo o el código de un producto estándar.

c. Transformación de Relaciones 1: N (Uno a Muchos)

La relación se representa mediante una clave foránea en la tabla correspondiente al lado "muchos" de la relación.

- Esta es la transformación más común y directa en los modelos industriales.

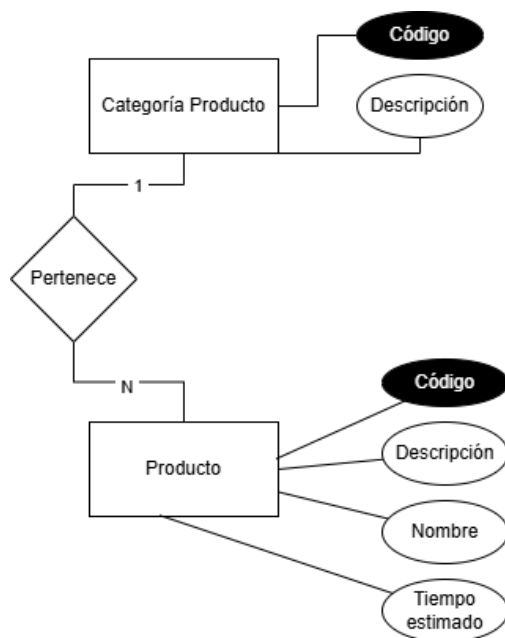


Fig. 13. Modelo conceptual de las entidades "Producto" y "Categoría Producto"

En el modelo conceptual presentado, se establece una relación uno a muchos (1: N) entre las entidades "Categoría Producto" y "Producto". Esto significa que cada producto se asocia a una única categoría, mientras que una categoría puede agrupar múltiples productos.

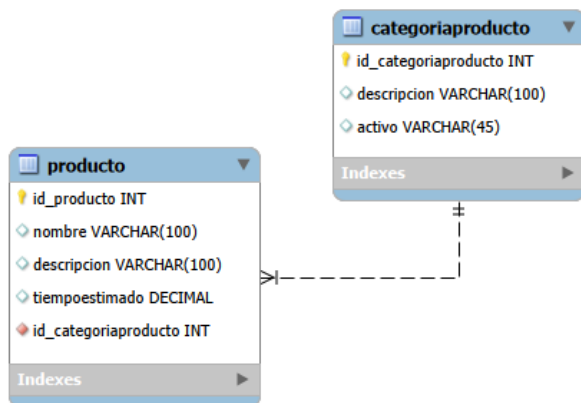


Fig. 14. Modelo lógico de las tablas “producto” y “categoría producto”

d. Transformación de Relaciones N:M (Muchos a Muchos)

Se crea una nueva tabla (tabla de asociación o tabla puente) que contiene las claves primarias de ambas entidades.

- Esta tabla representa la relación y puede incluir atributos propios de la relación.

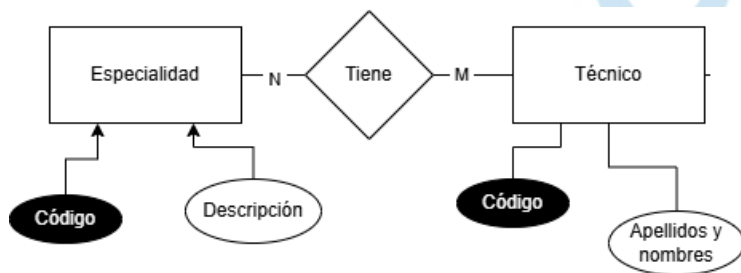


Fig. 15. Modelado conceptual entidades “Técnico” y “Especialidad”.

En el modelo conceptual de registro de fallos y mantenimiento, podemos ver que existe una relación entre técnico y especialidad de N: M, es decir un técnico puede tener muchas especialidades y una especialidad puede estar asignada a muchos técnicos.

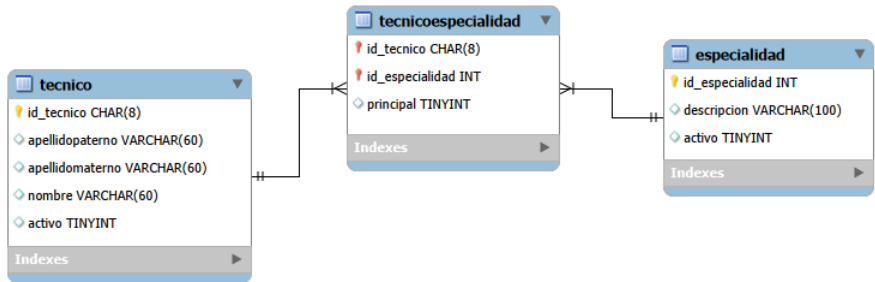


Fig. 16. Modelado lógico de las entidades “tecnico”, “especialidad” y la tabla de asociación “tecnicospecialidad”

Hemos creado una tabla de asociación, que contiene las claves de especialidad y del técnico, es decir se puede registrar varias especialidades para un técnico, además hemos agregado el campo (principal) que guardará para su valor (1) cual es la especialidad principal de un técnico.

5.2 Normalización de Datos

La normalización es el proceso de organizar los datos en una base de datos para reducir la redundancia y mejorar la integridad de los mismos. En el contexto industrial, donde los sensores y los registros de producción generan miles de líneas de datos por segundo, una base de datos mal normalizada puede causar cuellos de botella operativos y errores en los reportes críticos.

Las Formas Normales (FN). El proceso se divide en etapas llamadas Formas Normales. Cada etapa "limpia" el modelo de anomalías específicas.

a. Primera Forma Normal (1FN): Eliminación de grupos repetidos

- Cada celda debe contener un único valor (atómico).
- Cada registro debe tener una Clave Primaria (PK) única.
- No debe haber grupos de valores repetidos en diferentes columnas.

Ej. (`especialidad1`, `especialidad2`, `especialidad3`).

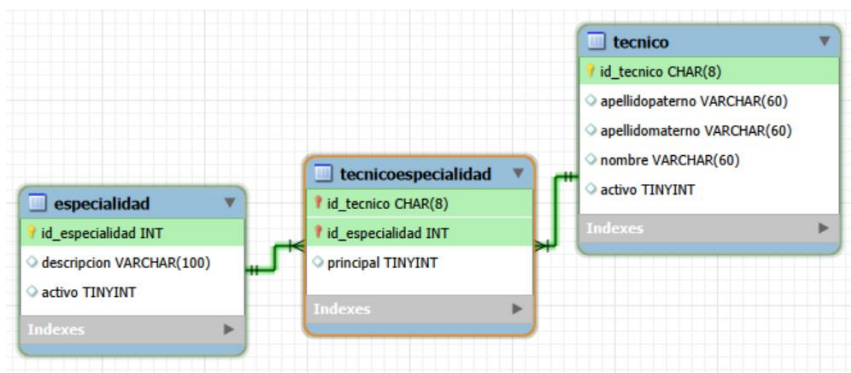


Fig. 17. Primera forma normal aplicada a las tablas “técnico” y “especialidad”

Como se observa en el modelo lógico, ante una relación de muchos a muchos (N:M) entre el técnico y sus especialidades, la solución óptima consiste en implementar una tabla de asociación (como tecnicospecialidad). Esta estructura no solo permite vincular a un técnico con múltiples especialidades de forma eficiente, sino que además admite la inclusión de atributos propios —por ejemplo, un indicador de 'especialidad principal' o fecha de certificación—, proporcionando la flexibilidad necesaria para escalar el sistema sin generar redundancia.

b. Segunda Forma Normal (2FN): Dependencia Funcional Completa

- Una tabla está en 2FN si ya cumple la 1FN y, además:

- Todos los atributos que no forman parte de la clave primaria deben depender de toda la clave primaria, no solo de una parte de ella.

Ejemplo:

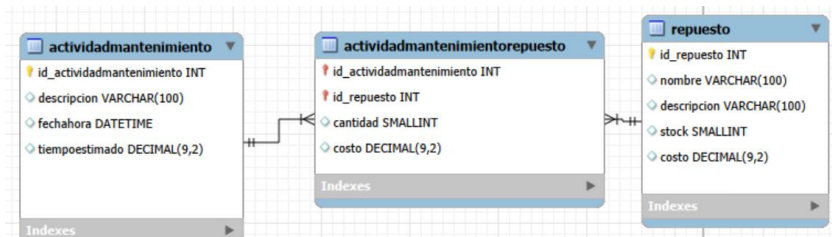


Fig. 18. Segunda forma normal aplicada a las tablas “repuesto”, “actividadmantenimiento” y “actividadmantenimientorepuesto”.

Como se observa en el modelo lógico, las tablas de asociación (como actividadmantenimientorepuesto) son ejemplos perfectos de la aplicación de la Segunda Forma Normal. En este caso, los campos cantidad y costo presentan una dependencia funcional completa respecto a la clave primaria compuesta (id_actividadmantenimiento + id_repuesto), ya que ambos valores son específicos de esa combinación única de intervención y material. Resultaría un error de diseño incluir la descripcion_repuesto en esta estructura, dado que dicho atributo depende exclusivamente de la tabla repuesto. Al mantener esta separación, garantizamos que cualquier cambio en la descripción del insumo se realice en un solo lugar, evitando inconsistencias en el historial de mantenimiento."

c. Tercera Forma Normal (3FN): Eliminación de Dependencias Transitivas

- Una tabla está en 3FN si cumple la 2FN y:
- No existen dependencias transitivas entre atributos. Esto significa que un campo no clave no puede depender de otro campo no clave.

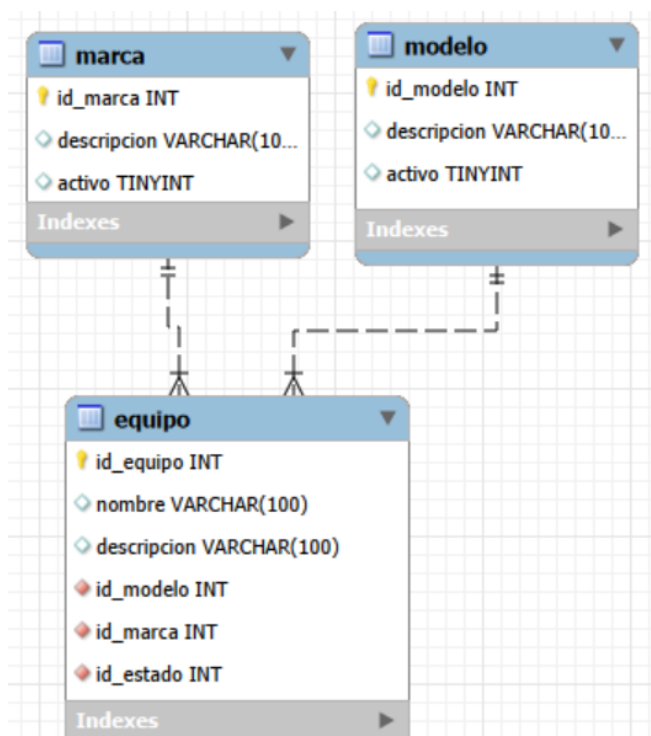


Fig. 19. Tercera forma normal aplicada a las tablas “equipo”, “marca” y “modelo”

Ejemplo: En un diseño ineficiente (no normalizado), podrías tener una sola tabla llamada equipo con campos como nombre_marca o nombre_modelo. Sin embargo, en el modelo propuesto se aplica la 3FN. La tabla equipo solo guarda los punteros id_marca e id_modelo.

Si hubiésemos incluido la descripción de la marca directamente en la tabla equipo, estarías creando una dependencia transitiva. El nombre de la marca depende del id_marca, y el id_marca depende del id_equipo. Al separar las marcas en su propia tabla (marca), aseguras que cada dato dependa únicamente de su propia clave primaria.

Beneficios de la normalización en la Ingeniería

- **Integridad de los Datos:** Evita inconsistencias. Si el nombre de un proveedor cambia, solo se actualiza en un lugar.
- **Optimización del Almacenamiento:** Al eliminar la redundancia, el tamaño de la base de datos se reduce significativamente.
- **Flexibilidad:** Facilita la expansión del modelo. Agregar un nuevo tipo de mantenimiento no requiere alterar toda la estructura de la orden de trabajo.

5.3 Errores comunes en modelos lógicos industriales

En el diseño de modelos lógicos de Bases de Datos aplicados a entornos industriales, ciertos errores conceptuales se repiten con frecuencia y suelen pasar desapercibidos en las primeras etapas del proyecto. Sin embargo, estas decisiones de diseño, aparentemente

menores, generan problemas graves de integridad, escalabilidad y confiabilidad de la información a mediano y largo plazo. Identificar y evitar estos errores es una competencia clave del ingeniero industrial responsable de la arquitectura de datos.

a. Uso de fechas u otros atributos temporales como clave primaria

Uno de los errores más comunes consiste en utilizar campos de tipo fecha u hora (fecha, fecha_hora, timestamp) como clave primaria de una tabla, bajo el supuesto de que cada evento ocurre en un momento único.

Por qué es un error

- Las fechas no garantizan unicidad absoluta: pueden existir múltiples eventos en el mismo segundo.
- Introducen dependencia directa del sistema de captura (zona horaria, latencia, formato).
- Dificultan la modificación o corrección de registros históricos.
- Limitan la escalabilidad cuando el volumen de eventos aumenta (sensores, IoT, registros operativos).

b. Repetición de datos de catálogo en múltiples tablas

Otro error frecuente es almacenar valores descriptivos repetitivos — como estados, tipos, categorías o prioridades— directamente en múltiples tablas, en lugar de modelarlos como catálogos independientes.

Por qué es un error

- Genera inconsistencias semánticas (ej. “Activo”, “activo”, “ACT”).
- Obliga a realizar actualizaciones masivas ante cambios de nomenclatura.
- Incrementa la redundancia y el tamaño de la base de datos.
- Dificulta la estandarización de reportes e indicadores.

c. No modelar históricos ni estados temporales

En muchos modelos lógicos iniciales se diseñan tablas que reflejan únicamente el estado actual de los procesos, ignorando la evolución histórica de la información. Este error es especialmente crítico en contextos industriales, donde la trazabilidad y el análisis temporal son fundamentales.

Por qué es un error

- Impide auditar eventos pasados.

- Limita el análisis de tendencias, fallos recurrentes o desempeño histórico.
- Obliga a soluciones improvisadas (copias en Excel, respaldos manuales).
- Reduce el valor analítico de la base de datos a largo plazo.



5.4 Caso Integrado: Del modelo conceptual al modelo relacional

Para el caso integrado, utilizaremos el ejemplo del modelado de un sistema de registro de fallos y mantenimiento.

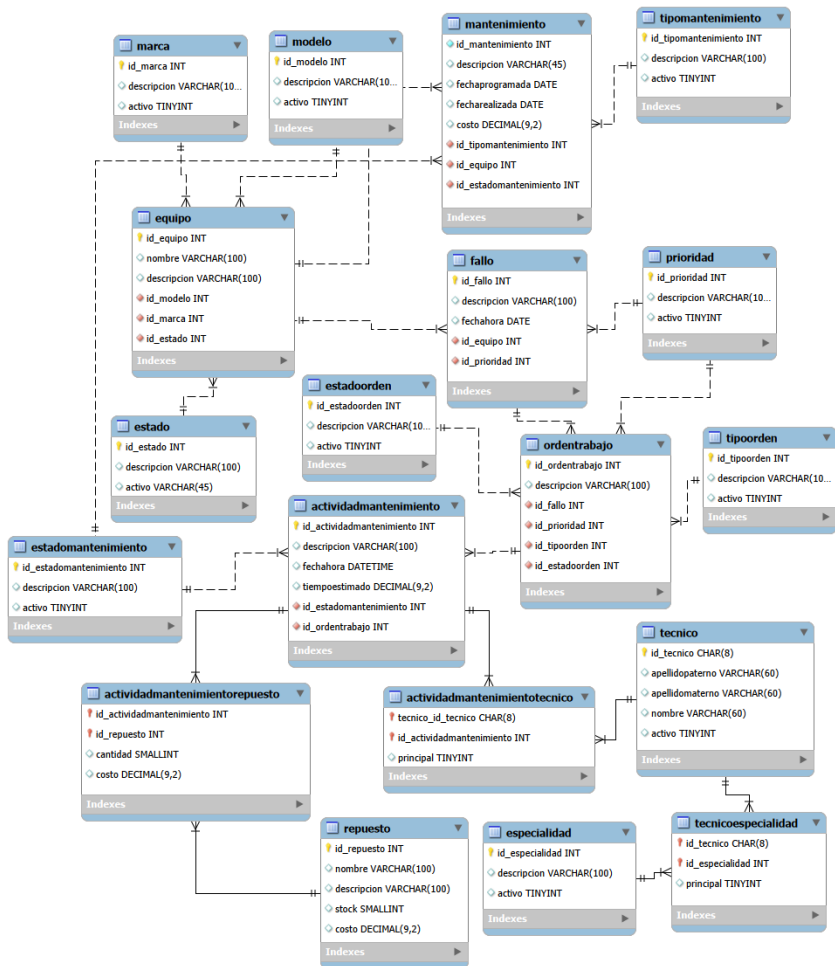


Fig. 20. Modelo lógico de un sistema de registro de fallos y mantenimiento.

Para asegurar una transición coherente y eficiente entre el modelo conceptual y su implementación lógica, se han aplicado las siguientes consideraciones arquitectónicas:

- **Normalización de catálogos y tablas maestras.** Se ha realizado extracción de atributos como marca, modelo, prioridad y estado hacia tablas independientes. En el modelo conceptual eran simples óvalos, pero en el lógico se han convertido en tablas maestras. Esto facilita la integridad y permite añadir metadatos (como el campo activo) sin redundancia.
- **Resolución de relaciones N: M.** las relaciones "Muchos a Muchos" identificadas en el modelo conceptual (Técnico-Actividad y Repuesto-Actividad) se han resuelto correctamente mediante tablas de asociación.
- **Implementación de la jerarquía de mantenimiento.** La cadena de dependencia:

equipo -> fallo -> ordentrabajo -> actividadmantenimiento

Esta estructura jerárquica es fundamental para mantener la trazabilidad exigida por los estándares industriales, permitiendo auditar desde el activo afectado hasta la tarea técnica ejecutada.

5.5 Ejercicios propuestos

Se propone el desarrollo del modelo lógico de datos para los siguientes escenarios industriales. El objetivo es transformar las abstracciones del modelo conceptual (ejercicios propuestos) en esquemas relacionales eficientes, garantizando la integridad y la velocidad de consulta en cada caso:

- Gestión de inventarios y almacenes
- Control de calidad y pruebas de producto
- Planificación de la producción y asignación de recursos
- Seguimiento de la cadena de suministro y logística
- Gestión energética y sostenibilidad

Resumen Parte III

Del modelo conceptual a la estructura relacional

La Parte III ha abordado el modelo lógico como la etapa en la que las abstracciones del modelado conceptual se transforman en estructuras formales capaces de ser implementadas en un sistema gestor de bases de datos. Este proceso representa un punto de inflexión en el diseño, donde las decisiones conceptuales se materializan en tablas, claves y relaciones que sostendrán la operación diaria y el análisis de la información.

A lo largo de esta sección se ha desarrollado la transformación del modelo entidad-relación al modelo relacional, destacando que dicha conversión no es automática ni meramente técnica. Por el contrario, implica decisiones de diseño orientadas a preservar la semántica del proceso, garantizar la integridad referencial y facilitar la trazabilidad de los datos a lo largo del tiempo.

Se ha profundizado en la definición de claves primarias y foráneas como mecanismos fundamentales para asegurar unicidad, consistencia y coherencia entre tablas, así como en el tratamiento adecuado de las relaciones uno a muchos y muchos a muchos. Estos elementos permiten modelar eventos recurrentes, históricos operativos y la interacción de múltiples actores dentro de los procesos industriales.

Asimismo, la Parte III ha incorporado el proceso de normalización como una herramienta para reducir redundancias, evitar anomalías de actualización y mejorar la calidad estructural de la base de datos. Desde una perspectiva industrial, la normalización se presenta no como un ejercicio teórico, sino como un medio para construir sistemas confiables, escalables y orientados a la eficiencia operativa.

En conjunto, esta parte consolida el modelo relacional como la base estructural sobre la cual se construyen los sistemas de información industriales. Al finalizar esta sección, el lector cuenta con los criterios necesarios para diseñar bases de datos coherentes con la lógica del proceso, preparadas para soportar análisis, indicadores de gestión y procesos de mejora continua.

Parte IV: Modelo Físico de Datos. De la teoría a la infraestructura operativa

Si el modelo lógico representaba el plano de ingeniería detallado de una planta industrial, el modelo físico constituye la fábrica en sí misma: construida, cableada y lista para operar. En esta etapa decisiva, los conceptos abstractos se materializan en estructuras ejecutables, donde la eficiencia, la seguridad y el rendimiento dejan de ser metas para convertirse en imperativos operativos diarios. La pregunta central ya no es solo "qué" información almacenar, sino "cómo" hacerlo de manera óptima: cómo estructurarla para un acceso ultrarrápido, cómo garantizar su integridad bajo condiciones de alta concurrencia y cómo mantener su disponibilidad las 24 horas del día. Este es el momento de la verdad, donde el diseño se traduce al lenguaje concreto de un motor de Bases de Datos (SQL) y se adapta meticulosamente a sus características técnicas únicas.

Capítulo 6: Especificación técnica y diccionario de datos

La transición del esquema lógico al modelo físico exige un salto de gran precisión. Cada entidad, atributo y relación debe ser revisado bajo una lupa técnica que defina su implementación exacta en el sistema. Este proceso de especificación se consolida en el Diccionario de Datos, el manual de referencia maestro que actúa como contrato técnico entre el diseñador, el administrador de Bases de Datos (DBA) y los desarrolladores de aplicaciones.

Es crucial realizar una revisión exhaustiva para definir:

- El tipo de dato preciso: No basta con "número"; se debe elegir entre INT, SMALLINT, DECIMAL(10,2) o FLOAT, según el rango y la exactitud requerida por el proceso industrial.
- La longitud o dominio óptimo: Determinar la longitud máxima de un VARCHAR (¿50, 100, 255 caracteres?) basándose en el análisis de los datos reales, evitando tanto la truncación como el desperdicio de espacio.
- Las restricciones de integridad a nivel de columna: Especificar qué campos son obligatorios (NOT NULL), cuáles tienen valores por defecto (DEFAULT) y qué reglas de validación (CHECK) deben aplicarse para garantizar la calidad del dato desde su ingreso.

A continuación, aplicaremos este proceso al “Caso Integrado”, analizando la implementación física de la tabla mantenimiento, un componente crítico para la confiabilidad de los activos productivos.

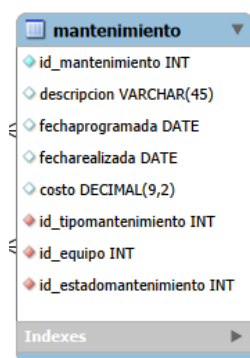


Fig. 21. Tabla mantenimiento.

a. Tipos de datos numéricos y claves primarias

Como estándar de diseño, la clave primaria (`id\_mantenimiento`) se ha definido como INT. Este es el tipo de dato más eficiente para la indexación y realización de cálculos vinculados a llaves primarias en motores relacionales. En MySQL, la elección del entero debe basarse en la escala del proceso industrial:

- TINYINT: Ideal para estados lógicos o categorías pequeñas (rango de -128 a 127).
- SMALLINT: Útil para contadores medianos.
- INT: El estándar para identificadores únicos y transacciones de gran volumen.

b. Cadenas de texto de longitud variable

Para el campo `descripcion`, se ha implementado el tipo VARCHAR(n). A diferencia del tipo `CHAR`, este optimiza el almacenamiento al ocupar solo el espacio del texto realmente ingresado. Se ha establecido una longitud de 45 caracteres, dimensión analizada como suficiente para detallar la naturaleza del mantenimiento sin desperdiciar recursos del servidor.

c. Atributos temporales y funciones del Sistema

Los campos `fechaprogramada` y `fecharealizada` utilizan el tipo DATE, diseñado para almacenar valores cronológicos de manera estructurada. El uso de este tipo de dato permite aplicar funciones de

motor como ``DATEDIFF()``, esencial para calcular indicadores de gestión como el cumplimiento de plazos o días de retraso entre la programación y la ejecución real.

Para fortalecer la auditoría del sistema, se puede incorporar el campo ``fecharegistro`` utilizando el valor por defecto `CURDATE()`. Esta función automatiza la captura de la fecha actual del servidor al momento de la inserción, garantizando la inmutabilidad del registro cronológico.

d. Reglas de validación

Para el campo ``costo``, se ha seleccionado `DECIMAL(9,2)`. Esta especificación permite almacenar hasta siete dígitos enteros y dos decimales (Ej. 9,999,999.99), asegurando precisión matemática en cálculos contables que los tipos de punto flotante podrían comprometer.

Además, a nivel físico se establece una regla de integridad (CHECK): el costo debe ser mayor o igual a cero. Esta restricción actúa como un filtro de calidad directo en la base de datos, impidiendo el ingreso de valores inconsistentes que afectarían los reportes financieros del área de mantenimiento.

Capítulo 7: Índices y estrategias de optimización

Una estructura bien construida debe diseñarse para responder con velocidad. Los índices son el sistema de transporte interno: como un índice alfabético en una enciclopedia, permiten al motor localizar un registro entre millones en milisegundos, sin tener que revisar cada fila.

Este pilar es crítico y estratégico. Se analiza cuándo indexar (columnas usadas frecuentemente en búsquedas, filtros o joins, como un `id_employado` o una `fecha_orden`) y, de igual importancia, cuándo evitarlo, ya que cada índice añade sobrecarga en las operaciones de escritura (inserciones, actualizaciones). El equilibrio define el rendimiento final del sistema.

a. Fundamentos de la indexación

En el modelo físico, la eficiencia no es una opción, sino una exigencia operativa. Imagine una tabla de mantenimiento con millones de registros de mantenimiento; buscar una orden específica sería un proceso inaceptablemente lento. Aquí es donde los índices actúan como un sistema de transporte de alta velocidad para localizar información entre millones de filas en cuestión de milisegundos.

Un índice es una estructura de datos persistente que almacena una selección de columnas de una tabla, organizadas de tal manera que permiten búsquedas extremadamente rápidas. Técnicamente, funcionan bajo los siguientes principios:

- Punteros y referencias: El índice no contiene todos los datos de la fila, sino una copia de la columna indexada (como el `id_equipo`) y un puntero físico hacia la ubicación exacta del registro completo en el disco duro.
- Búsqueda binaria y estructuras de árbol (B-Tree): En lugar de leer la tabla de forma lineal (del 1 al 1,000,000), el motor de la base de datos utiliza el índice para descartar masivamente los datos que no coinciden, reduciendo drásticamente el número de operaciones de lectura.
- Localización instantánea: Al igual que el índice alfabético de una enciclopedia permite saltar directamente a la página exacta de un concepto, un índice en SQL permite al motor saltar directamente a la fila deseada sin tener que inspeccionar el resto de la tabla.

El valor estratégico de la indexación

Este pilar es crítico para el desempeño de cualquier sistema industrial. Una indexación bien diseñada garantiza que, a medida que la empresa crece y acumula años de historial de producción, las consultas para generar indicadores de gestión (KPIs) o auditorías de calidad sigan respondiendo con la misma agilidad que el primer día.

Sin embargo, como todo sistema de ingeniería, requiere equilibrio: cada índice añadido mejora la velocidad de lectura (búsquedas), pero añade una ligera sobrecarga en las operaciones de escritura (inserciones y actualizaciones), ya que el índice también debe

actualizarse cada vez que el dato cambia. Por ello, se debe analizar estratégicamente qué columnas indexar —como identificadores de empleados, códigos de equipos o fechas de órdenes— para maximizar el rendimiento global del sistema.

b. Diseño de índices estratégicos

Saber cómo funciona un índice es solo la mitad del desafío; la verdadera pericia del diseñador de Bases de Datos reside en saber dónde y cuándo aplicarlos. Un exceso de índices puede ralentizar las operaciones de escritura (como el registro de una nueva orden de producción), mientras que la falta de ellos puede paralizar la generación de reportes estratégicos.

Para acelerar reportes críticos sin comprometer la agilidad del sistema, debemos aplicar los siguientes criterios de selección:

- Columnas de búsqueda frecuente (claves de filtrado)

Cualquier columna que aparezca habitualmente en la cláusula WHERE de una consulta SQL es una candidata primaria para la indexación.

Ejemplo: Si el departamento de planeación consulta constantemente la producción por fecha (`fecha_orden`), indexar este campo permitirá obtener resultados instantáneos de un día específico entre años de historial.

- Columnas utilizadas en vinculaciones (Joins)

En un modelo normalizado, la información está dispersa en varias tablas. Los índices en las claves foráneas (FK) son vitales para que el motor de la base de datos unas tablas rápidamente.

Ejemplo: Al generar un reporte que combine los datos del técnico con sus actividades realizadas, el campo `id_tecnico` en la tabla de asociación debe estar indexado para que el cruce de datos no degrade el rendimiento.

- Columnas con alta selectividad (cardinalidad)

La selectividad se refiere a cuántos valores distintos existen en una columna en relación con el número total de registros.

Criterio de éxito: Es mucho más eficiente indexar un código de empleado (`id_empleado`), donde cada valor apunta a una persona específica, que indexar un campo como "género" o "estado de actividad" (activo/inactivo), donde el índice tendría que devolver la mitad de la tabla de todas formas.

- Columnas de ordenamiento y agrupación

Si los reportes de producción requieren visualizar los datos ordenados por prioridad o agrupados por línea de producción (`GROUP BY`, `ORDER BY`), la creación de índices en estos campos evita que el servidor tenga que realizar procesos de

ordenamiento pesados en la memoria RAM cada vez que se solicita el informe.

- El principio de economía de índices

Finalmente, el diseño estratégico exige recordar que cada índice es un objeto físico que ocupa espacio en disco y tiempo de procesamiento durante las inserciones (INSERT) y actualizaciones (UPDATE).

Regla de oro: Indexe solo aquello que sea estrictamente necesario para el cumplimiento de los indicadores de gestión (KPIs) y la operatividad diaria. Un modelo físico con pocos índices bien elegidos siempre superará en eficiencia a uno con índices en cada columna.

c. El equilibrio del rendimiento

Como en todo sistema de ingeniería, la optimización de Bases de Datos es un juego de equilibrios. Aunque hemos establecido que los índices son "vías rápidas" para el acceso a la información, su implementación excesiva o descuidada puede transformar una base de datos ágil en un sistema pesado y costoso de mantener.

El equilibrio del rendimiento no se logra siguiendo una fórmula fija, sino mediante el análisis constante de la carga de trabajo del sistema. Un buen modelo físico es aquel que tiene

los índices mínimos necesarios para el máximo beneficio operativo, garantizando que la información estratégica fluya con la velocidad que la industria del futuro demanda.

- El costo oculto de la escritura

Cada vez que se ejecuta una sentencia INSERT, UPDATE o DELETE en una tabla con índices, el motor de la base de datos debe realizar doble trabajo:

- Modificar el dato real en la tabla física.
- Actualizar la estructura de cada uno de los índices asociados para reflejar el cambio.

En una planta industrial con sensores que registran datos cada segundo, o en un almacén con cientos de movimientos de stock por hora, un exceso de índices puede generar "cuellos de botella", retrasando el registro de la producción mientras el sistema reorganiza sus índices internos.

- El consumo de recursos físicos

Los índices no son invisibles; ocupan espacio físico en el disco duro y consumen memoria RAM. En entornos de grandes volúmenes de datos históricos, un conjunto de índices mal planificado puede llegar a ocupar tanto o más espacio que los datos reales de la tabla, incrementando los costos de

infraestructura y dificultando las tareas de respaldo y recuperación.

- Criterios para mantener el equilibrio

Para alcanzar el punto óptimo de rendimiento, el diseñador debe aplicar los siguientes principios:

Indexación selectiva: No indexar tablas pequeñas (donde el motor tarda menos tiempo leyendo toda la tabla que buscando en un índice) ni columnas con baja variabilidad de datos (como campos tipo "Sí/No").

Monitoreo de consultas: Identificar mediante herramientas de diagnóstico cuáles son las consultas que realmente tardan más tiempo y crear índices específicos para ellas, en lugar de intentar predecir cada búsqueda posible.

Mantenimiento periódico: Los índices se fragmentan con el uso intensivo. Es necesario programar tareas de "reorganización" o "reconstrucción" de índices para asegurar que el sistema de transporte siga siendo rápido a lo largo del tiempo.

Resumen Parte IV

Implementación física y gestión de bases de datos en entornos industriales

La Parte IV ha desarrollado el modelo físico de datos como la etapa final del diseño de bases de datos, en la que las estructuras lógicas se concretan en configuraciones técnicas orientadas al rendimiento, la seguridad y la operación continua de los sistemas de información industriales. En este nivel, el diseño deja de ser exclusivamente conceptual para convertirse en un componente crítico de la infraestructura organizacional.

A lo largo de esta sección se han abordado decisiones relacionadas con la definición de tipos de datos, la creación de índices, la organización del almacenamiento y la implementación de restricciones físicas, todas ellas orientadas a garantizar eficiencia operativa, integridad de la información y disponibilidad del sistema. Estas decisiones tienen un impacto directo en el tiempo de respuesta de las consultas, la concurrencia de usuarios y la estabilidad de la base de datos en escenarios de alta demanda.

Asimismo, se ha destacado la importancia del diccionario de datos como instrumento de documentación técnica y de gobierno de la información, permitiendo mantener coherencia semántica, facilitar el mantenimiento del sistema y asegurar la continuidad operativa ante cambios organizacionales o tecnológicos.

La Parte IV también ha puesto énfasis en aspectos fundamentales de la gestión de bases de datos, como el control de accesos, la seguridad de la información, la administración de usuarios y la planificación de respaldos. Desde una perspectiva industrial, estos elementos son esenciales para garantizar la confiabilidad del sistema y proteger uno de los activos más críticos de la organización: sus datos.

En conjunto, esta parte consolida la visión de la base de datos como un componente estratégico de la infraestructura industrial, cuyo diseño físico y correcta administración permiten sostener la operación, habilitar la analítica avanzada y acompañar los procesos de transformación digital. Con ello, el ingeniero industrial completa su rol como arquitecto de la información, integrando conocimiento del proceso, rigor técnico y visión estratégica en la gestión de los datos organizacionales.



Referencias bibliográficas

- Díaz, D. J. (2024). Un momentito con la Ingeniería Industrial (1ª ed.). Fondo Editorial.
- Coronel, C., & Morris, S. (2020). Sistemas de bases de datos: Diseño, implementación y administración (13ª ed.). Cengage Learning.
- Elmasri, R., & Navathe, S. B. (2016). Fundamentos de sistemas de bases de datos (7ª ed.). Pearson Education.
- Hernandez, M. J. (2020). Database design for mere mortals: A hands-on guide to relational database design (4th ed.). Addison-Wesley Professional.
- Kopp, S., & Taylor, A. G. (2023). SQL for dummies (10th ed.). John Wiley & Sons.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). Fundamentos de bases de datos (7ª ed.). McGraw-Hill.
- García, F. J., & Martínez, L. (2022). Sistemas de información para la gestión industrial: De los datos a la decisión (2ª ed.). Ediciones Paraninfo.
- Krajewski, L. J., Ritzman, L. P., & Malhotra, M. K. (2019). Administración de operaciones: Procesos y cadena de suministro (12ª ed.). Pearson Educación.
- Müller, M. (2021). Data Science en la industria 4.0: Conceptos y aplicaciones prácticas. Marcombo.
- Provost, F., & Fawcett, T. (2020). Data Science para negocios: Lo que necesitan saber sobre minería de datos y pensamiento analítico. Anaya Multimedia.