



# Módulo 4: Material Complementario

---

## Denormalización en Bases de Datos

### 1. Introducción

La denormalización es una técnica utilizada en el diseño de bases de datos relacionales que consiste en introducir deliberadamente redundancia en los datos, con el objetivo de mejorar el rendimiento del sistema, particularmente en entornos de consulta intensiva. A diferencia de la normalización, que busca eliminar la redundancia, la denormalización prioriza la eficiencia en ciertas operaciones, principalmente las de lectura.

### 2. ¿Por qué denormalizar?

En entornos reales, especialmente en sistemas de análisis de datos, reportes o consultas masivas, el acceso a datos altamente normalizados puede generar múltiples operaciones JOIN que afectan negativamente el rendimiento. En estos casos, la denormalización puede simplificar el modelo lógico y reducir el tiempo de respuesta.

### 3. Cuando es conveniente denormalizar

- Cuando existen muchos JOINS que impactan el rendimiento.
- Si el sistema es de lectura intensiva y escritura poco frecuente.
- En sistemas orientados a reportes (OLAP, data warehouses).
- Cuando se accede repetidamente a datos agregados o combinados.
- En sincronización con sistemas externos que requieren estructuras simplificadas.

### 4. Riesgos y desventajas

La principal desventaja es la introducción de redundancia y, por ende, la posibilidad de inconsistencias si los datos no se actualizan correctamente en todas sus copias. También se incrementa la complejidad de mantenimiento y se corre el riesgo de generar anomalías de modificación o eliminación.

### 5. Comparación con la normalización

La normalización prioriza la consistencia y la integridad de los datos mediante la separación en múltiples tablas relacionadas. La denormalización, por el contrario, reduce la complejidad de acceso a costa de mayor redundancia. Ambas son herramientas válidas que deben utilizarse según el contexto y los objetivos del sistema.

## 6. Ejemplo práctico

Supongamos el siguiente esquema normalizado:

**CLIENTES(id, nombre, id\_localidad)**

**LOCALIDADES(id, nombre, provincia)**

Una consulta típica requerirá un JOIN para obtener el nombre de la provincia.

En un modelo denormalizado, el campo provincia podría incorporarse directamente en la tabla CLIENTES, reduciendo la necesidad de realizar operaciones JOIN para cada consulta.

## 7. Buenas prácticas al denormalizar

- Justificar siempre con métricas (rendimiento, consumo de recursos).
- Documentar los cambios y las dependencias lógicas que se generan.
- Implementar mecanismos de actualización automáticos para evitar inconsistencias.
- Evaluar el impacto a largo plazo en la mantenibilidad del sistema.

## 8. Conclusión

La denormalización no es una mala práctica en sí misma, sino una decisión estratégica que debe tomarse con fundamentos claros. El equilibrio entre normalización y denormalización es lo que permite construir bases de datos eficientes, consistentes y adaptadas a las necesidades reales de cada sistema.