

Denormalization

In order to increase read performance at the expense of write performance and storage space, a database design technique known as de-normalization adds redundancy to a normalized database.

What It Means: Normalization improves data integrity by reducing redundancy by dividing data into several related tables.

In contrast, de-normalization reduces complex joins and speeds up data retrieval by combining tables, storing precomputed values, or duplicating data.

The Reasons for Using De-normalization:

1. Boost Query Performance: Especially for intricate reports and dashboards, queries run more quickly when there are fewer table joins.

2. Reduce Query Complexity: • Makes SQL easier for developers and end users, which is helpful for applications that use a lot of data.

3. Improve the Efficiency of Reading:

• Quicker access to read-intensive systems, such as data warehouses or analytics platforms.

4. Optimize for Particular Use Cases: • Adjust table layout to accommodate well-known query patterns (such as retrieving client orders and names simultaneously).

When to Use De-normalization:

- When there are a lot of selects and few inserts or updates in the system.
 - When a lot of JOIN operations are causing performance issues.
 - In systems that use data primarily for analysis, such as data warehouses or OLAP (Online Analytical Processing) systems.
 - When utilizing pre-aggregated reports or materialized views.
 - Following the discovery that normalization is causing the application's response time to slow down.
-

Adding redundant columns (like keeping a customer's name in the orders table) is one common de-normalization technique.

Making aggregate/summary tables (such as total sales by region).

- Preserving derived values (e.g., age rather than birthdate).
- Consolidating tables into one flat unit.

DE-NORMALIZATION

- ▶ Normalize to reduce redundancy
- ▶ Denormalize to improve performance

Normalized

CUSTOMERS	
CustomerID	1
Alice Wong	

ORDERS	
OrderID	2024-12-01
2024-12-01	

PRODUCTS	
ProductID	Name
5001	Keyboard
5002	

Denormalized

ORDERS	
OrderID	CustomerID
1	Alice Wong
Email	alice@email.com
	Keyboard