# Module Five

# Short Paper

# Database Normalization vs. Denormalization

Andrew Robinson

IT 650: Principles of Database Design

Southern New Hampshire University

August 29, 2024

Daniel Foster

## Introduction

Database normalization is the process of organizing a database in a way that reduces redundancy and avoids anomalies such as insertion, update, and deletion issues. Normalization is a crucial practice in database design to ensure data integrity, consistency, and efficient data management. However, in some production environments, denormalization—a process where some of the normalization rules are intentionally violated—can be used to improve performance under specific circumstances. This paper will explore the concepts of normalization and denormalization, their benefits and drawbacks, and the considerations for deploying each approach.

## Normalized Database

A normalized database is structured to minimize redundancy and prevent anomalies by adhering to a series of normal forms, typically the first three: First Normal Form (1NF), Second Normal Form (2NF), and Third Normal Form (3NF).

#### First Normal Form (1NF)

Ensures that each column contains atomic (indivisible) values and each record is unique.

#### Second Normal Form (2NF)

Builds on 1NF by ensuring that all non-key attributes are fully functionally dependent on the primary key.

#### Third Normal Form (3NF)

Further refines the structure by ensuring that all attributes are not only fully functionally dependent on the primary key but also non-transitively dependent.

## Advantages of a Normalized Database

#### Data Integrity

By reducing redundancy, normalized databases prevent update anomalies and ensure that data remains consistent across the database.

#### Efficient Storage

Normalization minimizes the amount of duplicate data stored, reducing the overall size of the database.

#### Simplified Maintenance

A well-structured, normalized database is easier to maintain and modify because relationships and dependencies are clear.

## Denormalized Database

A denormalized database, on the other hand, intentionally introduces redundancy by merging tables that would typically be separate in a normalized structure. This process can improve query performance by reducing the need for complex joins but at the cost of data integrity and increased storage requirements.

## Advantages of a Denormalized Database

#### Performance Improvement

Denormalization can reduce the number of joins required in queries, which can significantly improve performance in read-heavy environments.

#### Simplified Query Structure

With fewer joins, queries become simpler and faster to write and execute, which can be beneficial in applications with heavy reporting or analytical requirements.

## Drawbacks of Denormalization

#### Data Anomalies

Denormalization can reintroduce the risk of anomalies, such as update anomalies, where data must be updated in multiple places.

#### Increased Storage

Redundancy in data leads to increased storage requirements, which can be a concern in large databases.

## Circumstances for Deploying a Denormalized Database

Deploying a denormalized database can be reasonable in specific scenarios, such as:

#### Performance Optimization

In cases where read performance is critical, and the database serves high-volume, read-heavy workloads, denormalization can reduce the number of joins and improve query speed.

#### Complex Reporting Needs

When complex analytical queries require data from multiple tables, denormalization can simplify these queries and improve their performance.

#### Real-Time Systems

In real-time systems where response time is critical, denormalization can help reduce latency by minimizing the need for multiple table lookups.

However, denormalization should be approached with caution, as it reintroduces the potential for data anomalies and increased maintenance complexity. It is essential to document any denormalization decisions thoroughly and ensure that the database design still meets the business requirements and maintains data integrity.

## Normalization in the Grandfield College Project

In the Grandfield College project, the database will be primarily normalized to ensure data integrity, consistency, and ease of maintenance. The project's focus is on creating a robust software tracking system that can withstand audits and maintain accurate records of software installations, licenses, and requests. Given the importance of accurate data in this context, normalization will help prevent anomalies and ensure that the system operates reliably. However, should performance concerns arise during implementation, specific tables may be denormalized in a controlled manner to optimize query performance, particularly in reporting modules.

## Conclusion

Normalization and denormalization are both valuable techniques in database design, each with its own set of advantages and trade-offs. While normalization is crucial for maintaining data integrity and reducing redundancy, denormalization can be beneficial in scenarios where performance is paramount. The decision to denormalize should be made carefully, considering the specific needs of the application and the potential risks involved. For the Grandfield College project, a primarily normalized database structure will be implemented to ensure accuracy and reliability, with denormalization considered only if necessary for performance optimization.

## References

Conger, S. (2014). Hands-On Database (2nd ed.). Pearson Education (US). https://mbsdirect.vitalsource.com/books/9780133927078