

# Chapter 13

## Designing an Efficient and Scalable Relational Database Schema: Principles of Design for Data Modeling

**Rajesh Kanna Rajendran**

*CHRIST University (Deemed), India*

**T. Mohana Priya**

*CHRIST University (Deemed), India*

### **ABSTRACT**

*Relational databases are a critical component of modern software applications, providing a reliable and scalable method for storing and managing data. A well-designed database schema can enhance the performance and flexibility of applications, making them more efficient and easier to maintain. Data modeling is an essential process in designing a database schema, and it involves identifying and organizing data entities, attributes, and relationships. In this chapter, the authors discuss the principles of designing an efficient and scalable relational database schema, with a focus on data modeling techniques. They explore the critical aspects of normalization, data types, relationships, indexes, and denormalization, as well as techniques for optimizing database queries and managing scalability challenges. The principles discussed in this chapter can be applied to various database management systems and can be useful for designing a schema that meets the demands of modern data-intensive applications.*

### **INTRODUCTION**

This chapter is a comprehensive guide to designing an efficient and scalable relational database schema. The paper begins by discussing the importance of data modeling in the database design process and how it can impact the performance and flexibility of applications. We then delve into the critical aspects of data modeling, including normalization, data types, relationships, indexes, and denormalization. The

DOI: 10.4018/978-1-6684-9809-5.ch013

## Designing an Efficient and Scalable Relational Database Schema

normalization process helps in avoiding data redundancy, which can improve data consistency and reduce storage requirements. We discuss various normal forms, including first normal form (1NF), second normal form (2NF), and third normal form (3NF), and provide practical examples of how to apply them. We also discuss data types and how they can impact database performance and storage requirements. We cover common data types such as strings, integers, dates, and decimals, and discuss their advantages and disadvantages. Relationships are another critical aspect of data modeling, and we explore different types of relationships, including one-to-one, one-to-many, and many-to-many relationships. We also discuss how to define and enforce referential integrity constraints to maintain data consistency (Priyanka, et al.,2016).

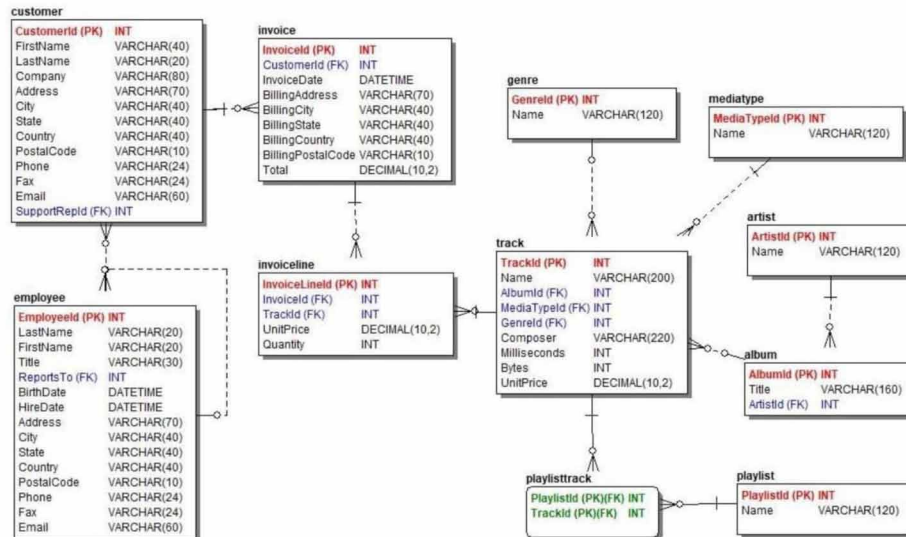
Indexes are essential for optimizing database queries, and we discuss how to create indexes and choose the appropriate data structures for specific scenarios. We also cover denormalization, a technique used to improve query performance by intentionally introducing redundant data into the database. Finally, we explore techniques for managing scalability challenges, including partitioning, sharing, and replication. We provide practical examples of how to apply these techniques in different scenarios and database management systems (Diego Sevilla et al.,2015).

Throughout the article, we provide practical examples and best practices for designing an efficient and scalable relational database schema, making it a valuable resource for database designers, developers, and administrators (David A. Maluf et al.) (Zain Aftab et al.,2020).

## RELATIONAL DATABASE

In order to demonstrate the power of a relational database, we'll create a database for a business that wishes to handle its merchandise, clients, orders, and staff (Alberto Hernández Chillón et al.,2019). Figure 1 shows Simple Entity Relationship Data Model.

Figure 1. Simple entity relationship data model



7 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/designing-an-efficient-and-scalable-relational-database-schema/330495](http://www.igi-global.com/chapter/designing-an-efficient-and-scalable-relational-database-schema/330495)

## Related Content

---

### Evolution of Security Engineering Artifacts: A State of the Art Survey

Michael Felderer, Basel Katt, Philipp Kalb, Jan Jürjens, Martín Ochoa, Federica Paci, Le Minh Sang Tran, Thein Than Tun, Koen Yskout, Riccardo Scandariato, Frank Piessens, Dries Vanoverberghe, Elizabeta Fourneret, Matthias Gander, Bjørnar Solhaugand Ruth Breu (2014). *International Journal of Secure Software Engineering* (pp. 48-98).

[www.irma-international.org/article/evolution-of-security-engineering-artifacts/121682](http://www.irma-international.org/article/evolution-of-security-engineering-artifacts/121682)

### Security Requirements Engineering for Evolving Software Systems: A Survey

Armstrong Nhlabatsi, Bashar Nuseibehand Yijun Yu (2010). *International Journal of Secure Software Engineering* (pp. 54-73).

[www.irma-international.org/article/security-requirements-engineering-evolving-software/39009](http://www.irma-international.org/article/security-requirements-engineering-evolving-software/39009)

### Change Impact Analysis for UML Model Maintenance

Anne Kellerand Serge Demeyer (2012). *Emerging Technologies for the Evolution and Maintenance of Software Models* (pp. 32-56).

[www.irma-international.org/chapter/change-impact-analysis-uml-model/60716](http://www.irma-international.org/chapter/change-impact-analysis-uml-model/60716)

### A UML-Compliant Approach for Intelligent Reconfiguration of Embedded Control Systems

Amen Ben Hadj Ali, Mohamed Khalgui, Samir Ben Ahmedand Antonio Valentini (2013). *Embedded Computing Systems: Applications, Optimization, and Advanced Design* (pp. 108-124).

[www.irma-international.org/chapter/uml-compliant-approach-intelligent-reconfiguration/76953](http://www.irma-international.org/chapter/uml-compliant-approach-intelligent-reconfiguration/76953)

### UML-Driven Software Performance Engineering: A Systematic Mapping and Trend Analysis

Vahid Garousi, Shawn Shahnewazand Diwakar Krishnamurthy (2013). *Progressions and Innovations in Model-Driven Software Engineering* (pp. 18-64).

[www.irma-international.org/chapter/uml-driven-software-performance-engineering/78208](http://www.irma-international.org/chapter/uml-driven-software-performance-engineering/78208)