

Teaching Exploration of Case-Based Data Modeling Optimization for Database System

Shicheng Hu¹, Yang Liu^{2*}, Song Wang¹

¹School of Economics & Management, Harbin Institute of Technology, Harbin, China

²School of Computer Science & Technology, Harbin Institute of Technology, Harbin, China

Email: *h13799915311@163.com

How to cite this paper: Hu, S. C., Liu, Y., & Wang, S. (2020). Teaching Exploration of Case-Based Data Modeling Optimization for Database System. *Open Journal of Social Sciences*, 8, 514-521.
<https://doi.org/10.4236/jss.2020.83044>

Received: March 15, 2020

Accepted: March 28, 2020

Published: March 31, 2020

Abstract

Database system is a very important information infrastructure of modern society. The consistent sharing of data is the key to the realization of reliable application of database system in different fields of society. With respect to the problems of data inconsistency that are widely existed in database system applications, we focus on the entity-relation modeling, the origin of data generation, combine with practical application cases and propose active and effective prior precautions which are extracted from our many years of teachings and scientific researches. First, we use the objective uniqueness of recognizing entity and relation to overcome the subjective biases in data modeling fundamentally. Next, some empirical rules for optimizing the entity-relation model are employed to further rectify the inappropriateness in the model. The effectiveness and feasibility of the proposed principles of the entity-relation modeling optimization for database system are sufficiently approved by the demonstration of practical application cases.

Keywords

Database System, Data Modeling, Entity-Relation Model

1. Introduction

Database system is an ideal core organization of data processing developed to meet the needs of data processing. The high-speed processing ability and large capacity memory of the computer provide the conditions for realizing the automation of data management (Liu, 2019). The emergence of database system is a milestone in computer application, which changes computer application from scientific computing to data processing, and enables computers to be widely

used in various industries and families (Luo, 2018). Compared with the traditional file management system, the database system provides a unified interface through the database management system DBMS, so the data storage is easier. The database system provides data integrity constraints through DBMS, so it is possible to maintain the consistency of the data. The database system solves the problems of large-capacity storage and high-density access of data in big data era through distributed processing technology (Muzammal, Qu, & Nasrulin, 2019).

Database system is a compulsory basic course for computer related majors (such as Computer Science and Technology, Software Engineering, Internet of Things Engineering, Management Science and Engineering, Data Science, etc.) (Liang, 2019). With the arrival of big data era and the increasing popularity of data-driven decision-making model, database system has become an important information infrastructure in modern society and an important supporting technology to realize the wide application of artificial intelligence in various fields of society.

However, whether it is e-commerce, traffic logistics, social media, or various database systems of artificial intelligence applications, there are more and more data inconsistencies. The specific manifestations are as follows: The redundancy of data, that is, the same kind of data appears many times in the database; The conflict of data, which is represented by the conflict of the same data that appears many times in the database; The noise of data, including data errors, missing data, etc. The problem of data inconsistency seriously affects the data quality, which greatly reduces the reliability of various database application systems (Li, Li, Chen, Yang, & Jiang, 2016). In order to solve the problem of data inconsistency, the industry has put forward a variety of data preprocessing techniques, such as data cleaning, data identification, data conversion, data extraction and data integration and so on (Zhang, 2018). These technologies have achieved certain results in dealing with data inconsistency, but it cannot fundamentally solve the problem of data inconsistency because the existing technology is essentially a post-processing technology. For fundamentally solving the problem of data inconsistency, it is necessary to seek a more positive and effective preventive measure to control it from the source of the data.

The transformation or abstraction of data from the physical world to the information world (that is, database) is realized through data modeling (Xiao, 2016). It is also called Entity-Relationship modeling in the current relational database system environment. However, in the current database system textbooks, only a qualitative description of entities and relationships is given, and a strict definition is not given. The physical world is rich and colorful, and different people may have a completely different understanding of the same object (Zhang, Ma, & Cheng, 2016), so it is difficult to provide a unified standard for Entity-Relationship modeling. For the same range of physical world, the Entity-Relationship model constructed by different people has its subjective bias. The subjective bias of this modeling is the root cause of data inconsistency.

2. The Objective Uniqueness Criterion for Identifying the Relationship between Entities

For the cause of clearly showing the rules and implementation process of database system data modeling and optimization, this paper first gives a specific case, which is described as follows.

The bank has many branches. Each branch is located in a specific city and is identified by a unique name. The bank monitors the property of each branch.

Bank customers are identified by their customer-id. The bank stores the name of each customer and the street and city where he lives. The customer can have an account and can make a loan. The customer may contact a bank employee who may act as the customer's loan officer or private bank assistant.

Bank employees are identified by their employee-id. The management body of the bank stores the name, telephone number, relative name of each employee and the employee-id of the manager. The bank also needs to know the date on which the employee starts work, from which the employee's period of employment can be calculated.

Accounts can be shared by two or more customers, or a customer can have two or more accounts. Each account is assigned a unique account number. The bank records the balance of each account and the most recent date on which each account owner accessed the account.

Each loan is issued by a branch and can be shared by one or more customers. A loan is marked with a unique loan number. Banks need to know the amount of each loan and the payment status. Although the payment number of a loan cannot be uniquely identified among all payments made by the bank for the loan, it can be uniquely identified as the payment made for a loan. It is essential to record the date and amount of each payment.

The above is a concise requirements analysis description of the banking application system. Clearly this includes both some entities and some links between entities. However, it is not easy to identify and connect. Some seeming entities are often mistaken for relationships, while relationships are treated as entities. Without a set of objective and unique identification principles, the two will often be confused. After years of teaching and application practice of the database, two guidelines for identifying the objective uniqueness of the entity relationship are given.

Rule 1: identifying an entity is fundamental, and an entity is generally a noun object.

Rule 2: identifying the relationship which is a verb object as a rule is the key.

There is no doubt that the identification of entities is fundamental, because the relation occurs between entities. If there is an entity identification error, the relation is bound to have an error. Since an entity is an "independent" static object, the only objective principle for identifying an entity is that an entity is a noun object. Of course, its state will change as there is a relationship with other entities, and its state is changed by other entities through relation. For example, a library book is an entity whose state include in the library, out of the library,

etc., and these states are changed by another entity reader by contacting the reader-the book.

The identification of relationship is based on entities. After all the entities are correctly identified, identifying the relationship will greatly reduce the identification error of the relation. Because at this time, the error of relationship identification only occurs between entities, not between entities and relationship. From the perspective of influence, the error of identification confusion between entities and relationships has bullwhip effect. The influence of errors will become greater and greater in the later stage of database construction, which cannot be corrected or suppressed. Errors between entities are relatively small, and certain measures can be taken to correct or suppress them after discovery.

According to the above two criteria of objective uniqueness in identifying entity relationship, an Entity-Relationship model of the banking application system can be preliminarily established. As shown in **Figure 1**.

3. Empirical Rules for Optimizing Entity-Relationship Model

The construction of Entity-Relationship model involves the subjective initiative or cognitive ability of human. Since the cognition ability between people is subjective in nature and varies according to different people, the cognition of the same object is bound to be different. Of course, although rules emphasize the objective uniqueness, different people will have different understandings of the two criteria for identifying the objective uniqueness of entity relationship. From the perspective of scientific understanding, the understanding of anything has a cycle of process. It's hard to do it overnight.

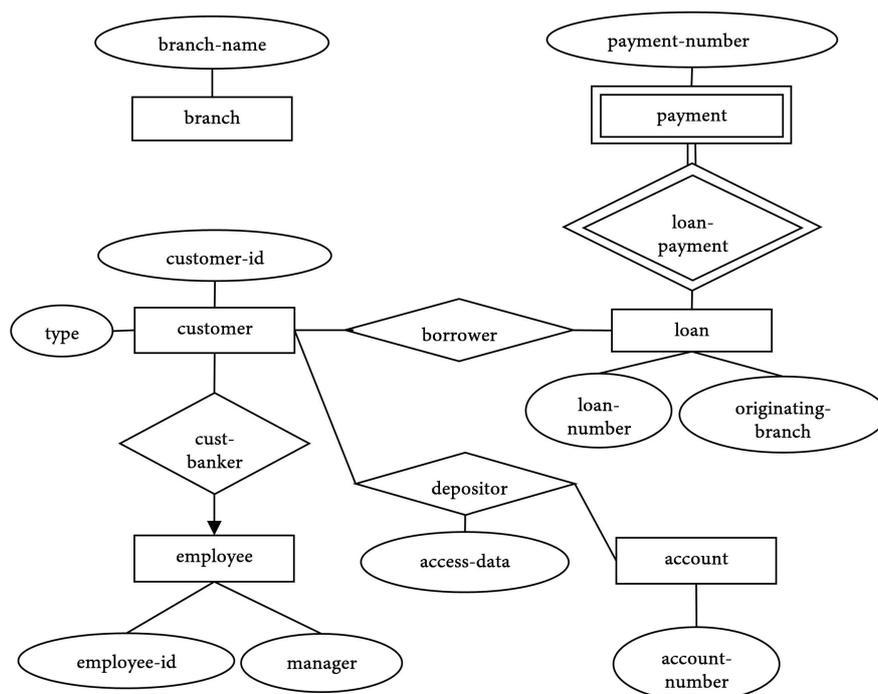


Figure 1. Initial entity-relationship for a banking application.

The same is true for database data modeling. Many years of teaching and application practice have proved that the objective uniqueness criterion for identifying entity associations can greatly avoid some common mistakes in database data modeling, but it cannot overcome some extreme anomalies. In view of this, we further explored the empirical rules for optimizing the Entity-Relationship model on the basis of the objective uniqueness criterion that is very effective for common mistakes:

Rule 3: There must be no isolated entities in Entity-Relationship.

Rule 4: The primary code of one entity set should not be an attribute of another entity set. The attribute implies a relationship between two entity sets.

Database is defined as a collection of interrelated data stored in a computer. Relational database is the most widely used data at present. The basic objects stored in a relational database are relationships. Each entity will be converted into a relation in the later stage of database construction. So it makes sense that the data in the database definition is a relationship. Obviously, according to the definition of the database, the basic requirement is to emphasize “interconnection”. This means that no isolated data objects are allowed. Data must be related to each other, and in the entity relationship model, entities must be related to each other. From the definition of the system, the internal objects of the system must form a system through some relationship since it is a system.

According to the definition of the entity: Things or events that can be distinguished from other objects in the real world. Each entity is marked by a unique code, and the difference between the entities is represented by a code value. If the primary code of an entity is an attribute of another entity set, then a primary code value must also be an attribute value of another entity. This objective fact indicates that there is a relationship between the two entities. Therefore, in this case, it needs to be expressed by relation.

Based on Rule 3, we find an isolated entity “Branch” in the entity relationship model which is shown in **Figure 1**. The entity cannot exist in isolation, and it must be examined to see if there is a relationship between the entity “Branch” and some other entity. It is also found that there is an attribute “originating-branch” for the entity “loan” in **Figure 1**. So how do we represent the specific value of this property? Evidently, you can only use the primary code value of the entity “branch”, which means that the primary code of the entity “branch” is also the property of “Loan”. Thus, according to Rules 3 and 4, isolated entities and implicit code attributes do not make sense. Using a combination of Rules 3 and 4, the link between entities “Loan” and “Branch” is reestablished, removing both isolated entities and unreasonable implicit code attributes. The specific process is shown in **Figure 2**.

In **Figure 1** we further discover that the entity “Employee” has a property “Manager”. So how do we represent the value of this “manager” property? It can only be represented by the primary key value of another entity “Manager”. By Rule 4, the implicit primary key attribute must be transformed into a relationship between two entities which are “Employee” and “Manager” respectively.

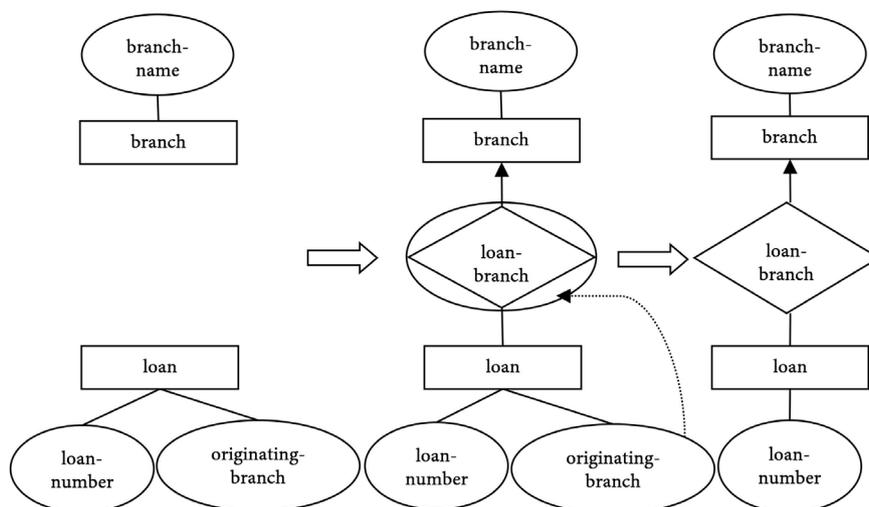


Figure 2. Modification of isolated entities and unreasonable implied primary codes.

However, the entity “Manager” does not exist in **Figure 1** at all. Do we need to add an entity “Manager”? If we add an entity “Manager”, how do we represent the relationship between it and the entity “Employee”? This is a problem that must be considered in the process of transformation.

The above problems cannot be solved completely by mechanical logical thinking. Field research is also needed to find out whether the two entities “Employee” and “Manager” are separate entities in the banking system. At the same time, we need to use Rule 1 to investigate. In fact, through further investigation of the banking system, it is found that “Employee” is indeed a real entity, and fully conforms to Rule 1. But “manager” is not a separate entity, but a kind of “employee”. “Manager” cannot exist alone and depends on “employee”. In other words, “Manager” is a relationship of “Employee”, that is, the relationship between superior and subordinate, which can be expressed as a unary relation among the members within the entity. The specific process is shown in **Figure 3**.

After the modeling and optimization of rules 1 - 4, we can get a relatively complete Entity-Relationship model, as shown in **Figure 4**.

4. Conclusion

Entity-Relationship data modeling is the transformation of data from the physical world to the information world, the first step to build database application system, and the fundamental guarantee of database data quality. Data modeling is a process of abstracting data from the perspective of data, which is highly subjective. In order to avoid the personality difference of the same physical world expressed the information world caused by cognitive subjectivity, we summarize the objective uniqueness criterion for identifying entity relationship and the empirical rule for optimizing Entity-Relationship model based on many years of teaching and application practice. We verify it through specific application examples. The practical application cases fully demonstrate the effectiveness and

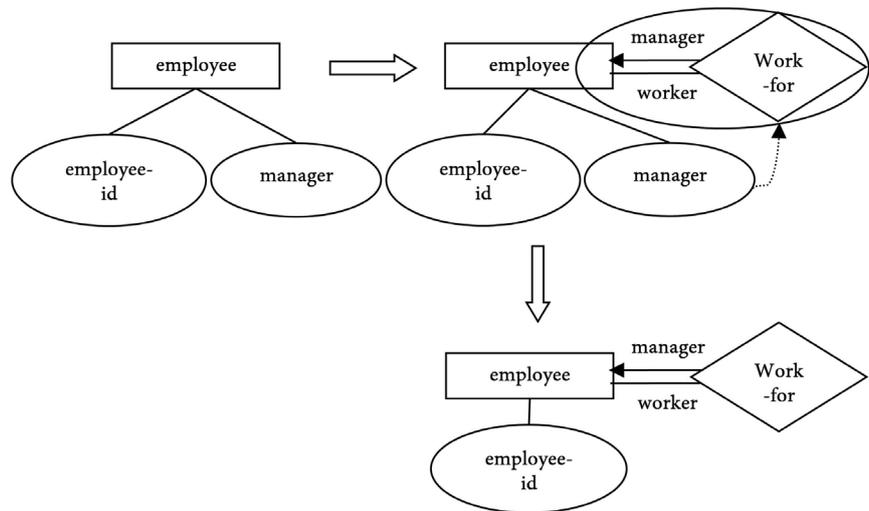


Figure 3. Representation of dependencies between members within an entity.

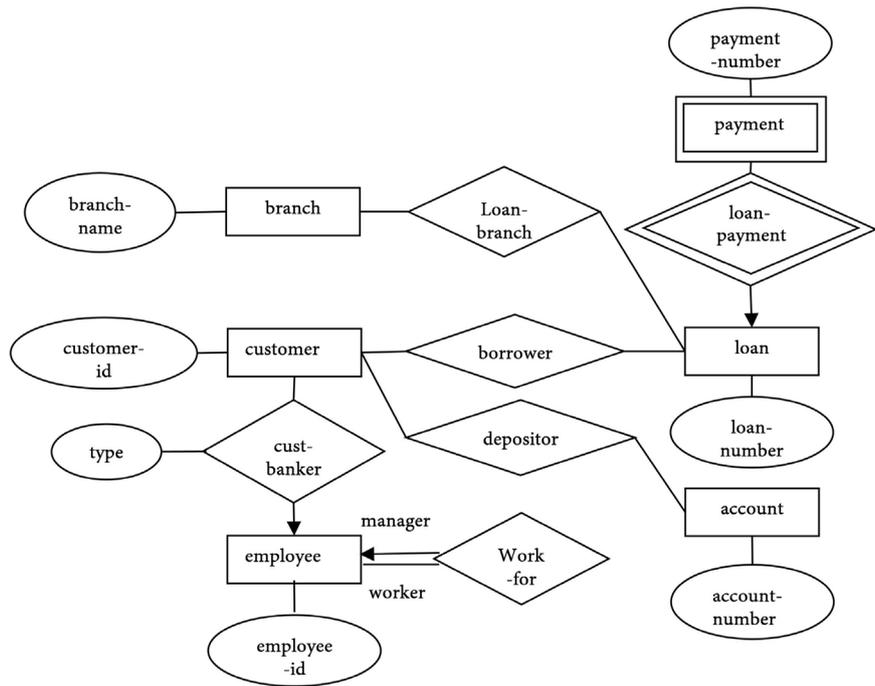


Figure 4. Entity relationship model after modeling optimization of banking system.

feasibility of the Entity-Relationship modeling optimization rules proposed in this paper.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

Li, W. B., Li, Z. H., Chen, Q., Yang, J. Y., & Jiang, T. (2016). Distributed Big Data Incon-

- sistency Detection. *Journal of Software*, 8, 2068-2085.
- Liang, Y. L. (2019). Teaching Research on the Course of "Database System". *Wireless Internet Technology*, 14, 34.
- Liu, X. (2019). Application Research of Computer Database Technology in Information Management. *Management Informatization in China*, 22, 165-166.
<https://doi.org/10.1177/0920203X19825589>
- Luo, W. W. (2018). Exploring the Effective Application of Computer Database System in Information Management Based on the Background of Big Data. *Informatization of China*, 4, 60-61.
- Muzammal, M., Qu, Q., & Nasrulin, B. (2019). Renovating Blockchain with Distributed Databases: An Open Source System. *Future Generation Computer Systems*, 90, 105-117. <https://doi.org/10.1016/j.future.2018.07.042>
- Xiao, H. R. (2016). *Principle and Application of Database*. Tsinghua University Press.
- Zhang, F., Ma, Z. M., & Cheng, J. (2016). Enhanced Entity-Relationship Modeling with Description Logic. *Knowledge-Based Systems*, 93, 12-32.
<https://doi.org/10.1016/j.knosys.2015.10.029>
- Zhang, Y. C. (2018). Inconsistent Repair and Preprocessing of Temporal RDF Data. *Computing Technology and Automation*, 4, 23.