Communications of the Association for Information Systems

Volume 56 Paper 10

3-25-2025

Conceptual Data Modeling Use: A Study of Practitioners

Abhishek Sharma
Loyola University Chicago

Nenad Jukić
Loyola University Chicago

Dinko Bačić
Loyola University Chicago

Sippo Rossi
Hanken School of Economics

Svetlozar Nestorov
Loyola University Chicago

Recommended Citation

Sharma, A., Jukić, N., Bačić, D., Rossi, S., & Nestorov, S. (2025). Conceptual Data Modeling Use: A Study of Practitioners. Communications of the Association for Information Systems, 56, 230-254. https://doi.org/10.17705/1CAIS.05610

Conceptual Data Modeling Use: A Study of Practitioners

Cover Page Footnote

This manuscript underwent peer review. It was received 11/05/2024 and was with the authors for one month for one revision. Cecil Chua served as Associate Editor.

Communications of the Association for Information Systems

Research Article DOI: 10.17705/1CAIS.05610 ISSN: 1529-3181

Conceptual Data Modeling Use: A Study of Practitioners

Abhishek Sharma
Information Systems & Supply Chain Management
Loyola University Chicago
Chicago, IL, USA

Nenad Jukić
Information Systems & Supply Chain Management
Loyola University Chicago
Chicago, IL, USA

Dinko Bačić
Information Systems & Supply Chain Management
Loyola University Chicago
Chicago, IL, USA

Sippo Rossi
Information Systems Science
Hanken School of Economics
Helsinki, Finland

Svetlozar Nestorov
Information Systems & Supply Chain Management
Loyola University Chicago
Chicago, IL, USA

Abstract:

Conceptual data modeling is widely viewed in academia as a critical part of relational database development, essential for reducing project failure risks. Although empirical studies have explored various aspects of its use, research about how frequently Conceptual data modeling is applied in practice and the reasons for its adoption or avoidance are lacking. This paper addresses this gap by presenting a study that evaluates the adoption of conceptual data modeling in the industry. The study begins with practitioner discussions to understand real-world project experiences, revealing a potential discrepancy between what is taught in academic settings and what is practiced in industry. Next, a survey of 485 database professionals is conducted and supplemented by follow-up interviews with 34 professionals. Findings indicate that fewer than 40% of practitioners consistently use formal conceptual data modeling, even in cases when they would like to do so. The survey identifies reasons for not using conceptual data modeling, and the follow-up interviews provide the practitioners with clarifications of the identified barriers. This research finds a positive association between conceptual modeling use and overall satisfaction with the outcome of the database development process. Lastly, the findings of this research offer important implications for both database practitioners and educators.

Keywords: Conceptual Data Modeling, Entity Relationship Modeling, Relational Database, Database Design, Database Implementation.

This manuscript underwent peer review. It was received 11/05/2024 and was with the authors for one month for one revision. Cecil Chua served as Associate Editor.

1 Introduction

Conceptual data modeling is widely accepted in academia as an essential and integral part of the database development process, and a key factor for minimizing the risk of project failures. Conceptual data modeling is described in detail in database textbooks and taught in most university-level database courses at the undergraduate and graduate levels. There have been several empirical studies in the past investigating different aspects of the use of Conceptual data modeling by practitioners. However, research examining the extent of actual use of Conceptual data modeling in practice is lacking. Also, we found that none of the previous Conceptual data modeling studies examine whether Conceptual data modeling is used in practice similarly and to the extent suggested by mainstream academic education on the subject. In this paper, we close this gap and present a study that combines both a survey and interviews with database professionals to analyze the adoption of Conceptual data modeling among practitioners. The survey was distributed to practitioners primarily in the United States and Europe, and it resulted in 485 responses. To validate the most significant findings of the survey, we conducted follow-up interviews with selected database professionals who participated in the survey, resulting in a mixed methods approach.

The paper is organized as follows. The next section gives a brief overview of the main steps in the database development process, including the Conceptual data modeling. The following section discusses the most relevant literature on Conceptual data modeling and describes the study motivation. This is followed by the methodology section detailing the survey development, deployment, and participant profile, as well as describing the interview process. The next section presents the survey and interview results, followed by a discussion of the results in the subsequent section. The paper ends with a section that presents the conclusions and proposed future work.

2 Overview of Basic Database Development Steps

Modeling databases is one of the most critical parts in the process of developing an information system, as most databases are created to serve as an integral part of the infrastructure of an information system. Most contemporary database textbooks (such as Elmasri & Navathe, 2015; Hoffer et al., 2021; Jukic et al., 2021; etc.) cover in detail the process of creating and using relational databases and present a standard development lifecycle for creating relational databases, which includes among its basic steps: requirement definition, conceptual data modeling, logical modeling (relational modeling) and implementation.

As described in the textbooks, in Step 1 (Requirements), the requirements for the database are collected and documented. In Step 2 (Conceptual Data Modeling), the requirements created in Step 1 are visualized as a conceptual database model. In Step 3 (Logical Database Modeling), the logical (implementational) database model is created as a relational schema mapped from the conceptual database model created in Step 2. In Step 4 (Implementation), the SQL statements based on the relational schema are executed in a Relational DBMS (RDBMS), such as Oracle, PostgreSQL, MS SQL Server, or similar. Once the database is created, front-end components for data entry and use, containing menus, forms, reports, etc., are created for the database as a part of a broader information system. Then, the information system with its database is deployed and used by its intended end-users, while database administrators (DBAs) provide the user support and maintenance of the database. As a brief overview of the database development steps, in Appendix A, we give an illustrative example of a generic company developing a small database.

This paper focuses on Conceptual data modeling (Step 2). In particular, we are interested in the industry-wide adoption (or lack thereof) of Conceptual data modeling as an integral part of the relational database development cycle.

3 Conceptual Data Modeling Background and the Study Motivation

3.1 Defining Conceptual Data Modeling

Conceptual data modeling is an inherent part of system analysis and design, particularly in its business requirement stage (Jukic et al., 2021, Hoffer et al., 2021). However, its importance extends throughout the entire information systems lifecycle (Fettke, 2009). The primary intent of Conceptual data modeling is not to model the implementational version of the database (this role is carried out by logical database modeling). Instead, the primary role of Conceptual data modeling is to build a model of reality that reflects relevant concepts necessary to capture the real-world domain processes. The need to understand, document, and communicate the domain - regardless of the systems development approach (traditional waterfall, agile, or a hybrid) (Spurrier & Topi, 2021), technology, or context – is often verbalized and visually depicted by clarifying the relevant concepts, the relationships between those concepts, along with identifying the required attributes for each concept (Batra & Marakas, 1995). Conceptual data modeling is not used simply to provide a reflection of database requirements once those requirements are collected, but it is a tool for managing the requirements-gathering process that keeps the focus on asking or seeking answers to the right questions to establish the relevant facts (Jukic, et al., 2020).

3.2 The History of Conceptual Data Modeling

The origins of conceptual data modeling can be traced back to the early and mid-1970s, coinciding with the advent of relational databases. Codd's (1970) seminal work in 1970 introduced the concept of normalization, a method for logically isolating and segmenting data into smaller, manageable tables. Building on this foundation, Abrial (1974) proposed a quasi-semantic model that described data elements as visual concepts, marking an early attempt to graphically depict data without relying on existing datasets. This approach reached a significant milestone with Chen's (1976) introduction of the Entity-Relationship (ER) model, which matured the idea of visualizing data concepts into a widely accepted framework.

Following these foundational developments, research in conceptual data modeling continued to evolve, producing various techniques that are still prevalent today. Early studies primarily focused on enhancing the formalism of these models (Hammer & McLeod, 1981; Hull & King, 1987; Peckham & Maryanski, 1988; Smith & Smith, 1977) and assessing their effectiveness and impact on users (Batra & Davis, 1992; Brosey & Shneiderman, 1978; Jarvenpaa & Machesky, 1989). These advancements allowed users and designers to concentrate more on data properties rather than being preoccupied with file structures and storage details (Elmasri & Navathe, 1988). As a natural progression, theoretical aspects of conceptual modeling began to be investigated (Wand et al., 1995) alongside formalism, techniques, and tools for conceptual modeling (Storey et al., 1997).

As the field matured, various methods emerged for capturing and annotating domain-specific data, leading to a proliferation of conceptual data modeling variants (Siau, 2004; Song et al., 1995). These methods quickly gained traction in practice. For instance, a 2001 survey highlighted the widespread use of ER diagrams, data flow diagrams, and flowcharts among users (Davies et al., 2006). By 2009, about a dozen modeling techniques had become relevant in practice, with the classical ER modeling approach remaining dominant, closely followed by the Unified Modeling Language (UML) (Fettke, 2009; Siau & Cao, 2001).

3.3 Brief Overview of Existing Literature

In the broader context of evolving technology and development paradigms, current research underscores the ongoing importance and relevance of studying conceptual modeling as information systems practices adapt to the digital age (M. Jabbari et al., 2017; Recker et al., 2021). However, it is also acknowledged that conceptual data modeling requires further contextual adjustments to keep pace with technological shifts. Jabbari and Rosemann (2023) recently observed that the field of conceptual modeling has not yet undergone a comprehensive renewal in light of digitization. The importance of conceptual modeling in modern information systems development has even been questioned (Lukyanenko & Parsons, 2013), while other studies have emphasized the continued significance of conceptual data modeling, suggesting that it can address challenges posed by Big Data through methods such as conceptual-model-based extraction to manage volume and velocity, inter-conceptual-model transformations to handle variety, and conceptualized constraint checking to enhance veracity (Embley & Liddle, 2013). Additionally, Jaakkola and Thalheim (2020) proposed that conceptual data modeling could play a crucial role in overcoming challenges associated with implementing artificial intelligence (AI) and machine learning (ML). Also, Hvalshagen et. al. (2023) proposed that textual narratives can significantly improve the understanding of the semantics of the conceptual database models and thereby broaden the understanding of conceptual database models beyond the technical database professionals.

Over the past two decades, numerous empirical studies have explored various aspects of how practitioners use conceptual data modeling. For instance, some studies have examined current trends in high-level data modeling (Anglim et al., 2009), identified the tools practitioners commonly use—such as UML and ER modeling (Davies et al., 2006; Fettke, 2009)—and investigated how entity-relationship modeling is applied in ternary relationship situations (Hitchman, 2003). Others have explored the human factors of conceptual modeling (Topi & Ramesh, 2002), why and how practitioners use multiple conceptual models (M. Jabbari & Recker, 2017), and how IS domain knowledge and application domain knowledge affect schema understanding (Khatri et al. 2006). However, none of these studies specifically investigated the prevalence of conceptual data modeling or the reasons behind its adoption. Furthermore, while they focused on practitioners who engaged in conceptual data modeling, they did not examine those who chose not to use conceptual data modeling. As a result, no research has explained why some professionals familiar with conceptual data modeling might end up not practicing it.

3.4 Study Motivation

As we stated earlier, conceptual data modeling is a staple of most database textbooks. Consequently, it is also taught worldwide in many database classes at graduate and undergraduate levels at academic institutions. Additionally, statements such as: “The entity-relationship (ER) model and its extensions have been widely used as conceptual models in database design” (G. Chen et al., 2006) are common in research papers.

Given that conceptual data modeling succinctly represents and manages the requirements-gathering process, omitting it can lead to project failure and resource wastage. Baghizadeh et al. (2020) highlight "weak definitions of requirements" as a key reason for the failure of information management projects. Similarly, Yang (2003) and Thalheim (2013) argue that errors in conceptual data modeling result in non-normalized outcomes. Yang (2003) further emphasizes the significance of conceptual data modeling in contexts where practitioners bypass modeling and proceed directly to creating relational tables, noting that the tabular format of data (relational tables) is more complex and prone to errors.

However, there are not many research studies examining the real-world adoption of conceptual data modeling. Even the prominent study involving 312 professionals, titled “How do practitioners use conceptual modeling in practice?” (Davies et al., 2006), did not assess the extent to which practitioners use conceptual data modeling. Instead, it concentrated on preferences for particular tools and techniques within conceptual data modeling. While these findings provided some interesting insights—such as preferences for certain tools—there is still no clear understanding of how widespread conceptual data modeling is or the factors that influence its use. This research aims to fill that gap by determining the actual level of adoption of conceptual data modeling in practice and identifying the factors that contribute to its use.

Our preliminary conversations with a selected group of practitioners and our own experiences with real-world projects indicated that conceptual data modeling may not always be universally practiced in real-world projects. To examine the level of adoption of conceptual data modeling in the real world, we decided to undertake a survey of working professionals that asks practitioners whether they use conceptual data modeling or not and, if not, why. In this survey, we are examining three main questions:

1) To what extent is conceptual data modeling used in practice?
2) What are the reasons for not using conceptual data modeling in cases when conceptual data modeling is not used?
3) To what extent does formal conceptual data modeling use (or non-use) affect the outcome of converting requirements to working databases?

The following section gives details of the survey development, deployment, and participant profile.

4 Methodology

This research follows a mixed method approach employing both quantitative and qualitative components sequentially, with an explanatory strategy (Venkatesh et al., 2013). First, a survey was used to establish the extent of conceptual modeling among practitioners quantitatively. Second, as the qualitative component, a set of interviews were conducted with practitioners. The purpose of this sequential approach was to validate and explain the key findings of the survey with the qualitative interviews, which go deeper into detail than the survey.

Next, we will describe the survey and interview process and outline the participants’ demographic information.

4.1 Survey Development

Survey development started with creating an initial set of questions, followed by a series of discussions with a dozen experienced database developers (a mix of consultants and employees of companies varying in size from small startups to Fortune 500 companies) who provided suggestions and validated the questions. The preliminary version of the survey was administered to the additional twenty industry professionals (of a similar mix to the initial group) who provided their feedback via email. Once we consolidated all the feedback, we created the final working version of the survey which was deployed to data professionals who are either specifically employed as data modelers or who have engaged with data modeling methodologies as part of database development projects.

The final version of the survey starts with several demographic and filtering questions. These questions ask respondents about their age, type, and level of education, size of their workplaces, and length and type of work experience. The survey then proceeds with three main questions about the respondents' practices when they were involved in the development of relational databases.

The first of these questions (Q1) asks how often an explicitly documented conceptual database model reflecting the requirements for the database was created. In addition to finding out how often the conceptual data modeling reflecting the requirements was created, we also wanted to find out the reasons for not creating conceptual database models from respondents who do not create conceptual database models regularly when developing databases.

Therefore, in a follow-up question (Q2) we asked the respondents to indicate the rationale for not creating the explicitly documented conceptual model. This question was accompanied by a list of possible reasons for not engaging in conceptual data modeling while developing a database. The list included eleven reasons which we categorize as a) resources issues (caused by the lack of resources, including the lack of time), b) methodology and perspective issues (reasons by practitioners who do not deem conceptual modeling a necessary part of database development process), c) work environment issues (the constraints imposed by the work environment), d) informal conceptual data modeling (in scenarios when informal whiteboard conceptualizing was done, and there was no formal conceptual data modeling beyond it), and e) other reasons. The creation of this list started during extensive discussions between the study authors and a dozen experienced industry professionals, which produced the initial list of reasons and their categories. The initial list was expanded, modified, and verified during the alpha survey phase that included another twenty industry professionals. The feedback from these processes was consolidated and synthesized into the final list of reasons that were presented as choices for Q2.

The survey concludes with a question on how often the respondents were satisfied with how the process of converting requirements to working databases turned out (Q3). The purpose of this question was to measure whether and how using conceptual data modeling affects the success of the entire process of building a database.

4.2 Survey Deployment

The survey was deployed using Qualtrics starting in April 2022 and completed in May 2024. Using LinkedIn, professional contacts in various consulting and technology companies, and several vendor lists as well as professional data associations and organizations, we approached hundreds of qualified professionals worldwide whose titles and job descriptions included terms such as database designer, database architect, database engineer, database analyst, business analyst, data analyst, and data scientist.

The survey was anonymous, but the participants were given the option to leave their email addresses (in case they were open to being contacted after completing the survey), and they were also given an opportunity to include any written comments that they deemed relevant. The landing page of the survey contained a short write-up and example of the scenario that recaps the textbook process of designing relational databases (including the idea of conceptual data modeling), followed by a button for starting the survey.

The survey started with two screening questions about participants' prior involvement i) in building relational databases, and ii) with relational databases that they did not initially create (by using/maintaining/further developing an existing relational database). These questions were used to filter out anyone who has not personally participated in the relational database development and who has also not personally worked with relational databases created by other developers.

4.3 Survey Demographics

We have received full responses from 485 participants with at least one year of professional experience designing relational databases. Figures 1-6 illustrate the composition of the participants per location, size of workplace, age, levels, and area of study for the degrees earned, and the number of years designing relational databases.

Figure 1 shows the geographical location of the participants. Most respondents were based in North America, with the rest of the participants from Europe and other parts of the world.

Location	# of Respondents	
North America	390	80.4%
Europe	70	14.4%
Asia	16	3.3%
Other	9	1.9%
Grand Total	485	

Figure 1. Location

Figure 2 shows the size of the respondents' companies. Approximately half of the respondents work in larger companies with more than 1000 employees, and a quarter of respondents work in medium size companies with between 100 and 1000 employees. The rest work in smaller companies of varying sizes.

Workplace EE Size	# of Respondents	
1-10	29	6.0%
10-100	77	15.9%
100-1000	127	26.2%
> 1000	245	50.5%
Don't Know	7	1.4%
Grand Total	485	

Figure 2. Size of Workplace (Employee Count)

Figure 3 shows the age groups of the respondents. About 28% of the respondents are younger than 30. Approximately one-third of respondents are in their 30s. Another third of the respondents are in their 40s and 50s, and the remainder are 60 years old or older.

Age Group	# of Respondents	
< 20	1	0.2%
20-29	136	28.0%
30-39	153	31.5%
40-49	112	23.1%
50-59	68	14.0%
60+	11	2.3%
Not Declared	4	0.8%
Grand Total	485	

Figure 3. Age

Figures 4 and 5 show the educational background of the respondents. As shown in Figure 4, more than 99% of the respondents have at least a bachelor's degree, with more than half of the respondents having postgraduate degrees.

Education Level	# of Respondents	
H School Degree	4	0.8%
Bachelor's Degree	203	41.9%
Master's Degree	242	49.9%
Doctorate Degree	36	7.4%
Grand Total	485	

Figure 4. Highest Degree Earned

Figure 5 shows the variety of areas that the respondents studied during their education. As many respondents obtained degrees in multiple areas, the percentages in this table do not add to 100%. Respondents have degrees in a variety of areas, with most having degrees in data and technology areas such as Information Systems, Computer Science, and Data Analytics.

Area of Study	n	
Information Systems	253	52.2%
Computer Science	150	30.9%
General Business	114	23.5%
Data Analytics	100	20.6%
Engineering	51	10.5%
Humanities & Social Science	23	4.7%
Natural Science	14	2.9%
Other	56	11.5%

Figure 5. Areas of Study

Figure 6 shows the length of the professional database experience for the respondents. Approximately, one-third of respondents have between 1 and 4 years of experience. Another third has between 5 and 10 years of experience. The rest have more than 11 years of experience, and some have more than 20 years of experience. The average for all respondents is over 10.7 years of experience in designing relational databases.

Years of Experience	# of Respondents	
1 - 2	83	17.1%
3 - 4	65	13.4%
5 - 10	154	31.8%
11 - 20	120	24.7%
> 20	63	13.0%
Grand Total	485	

Figure 6. Number of Years Designing Relational Databases

4.4 Interviews

After the survey was completed in May of 2024, in the period between August and October of 2024 we approached and interviewed 34 survey respondents who volunteered their email addresses upon completing the survey. The interviewees reflected the overall survey sample accurately, with 85% of respondents based in North America. Of those interviewed, 56% held baccalaureate degrees, and 44% held master's degrees. Half were employed by companies with over 1,000 employees, and participants had an average of 9.9 years of experience in database design or usage. Additionally, 78% had academic backgrounds in Information Systems, 32% in Data Analytics, and 26% in general business. The interviews provided additional insights and perspectives and insights about the results of the survey.

The next section presents, discusses, and analyzes the results of the survey and the follow-up interviews.

5 Results Analysis and Discussion

5.1 Survey Results

To examine what the survey shows about the frequency of use of conceptual data modeling, reasons for not using conceptual modeling, and the effect of the use of conceptual modeling on the overall database development outcome satisfaction, we present and discuss the results revolving around questions Q1, Q2, and Q3.

First, we will show the findings about the frequency of use of conceptual data modeling. Question Q1 asked participants "How often was an explicitly documented conceptual database model reflecting the requirements for the database (such as an ER Diagram) created in cases when you were involved in building a database?" Figure 7 shows the results of this survey question. Among the 485 participants, only a small fraction (less than 13%) indicated that they consistently used conceptual data modeling during database development. Additionally, more than half (55%) reported that they only sometimes or rarely employed conceptual data modeling, while another 5% report never using conceptual data modeling. This finding supports our initial perception from the pre-survey refinement that conceptual data modeling may not always be explicitly practiced in real-world projects.

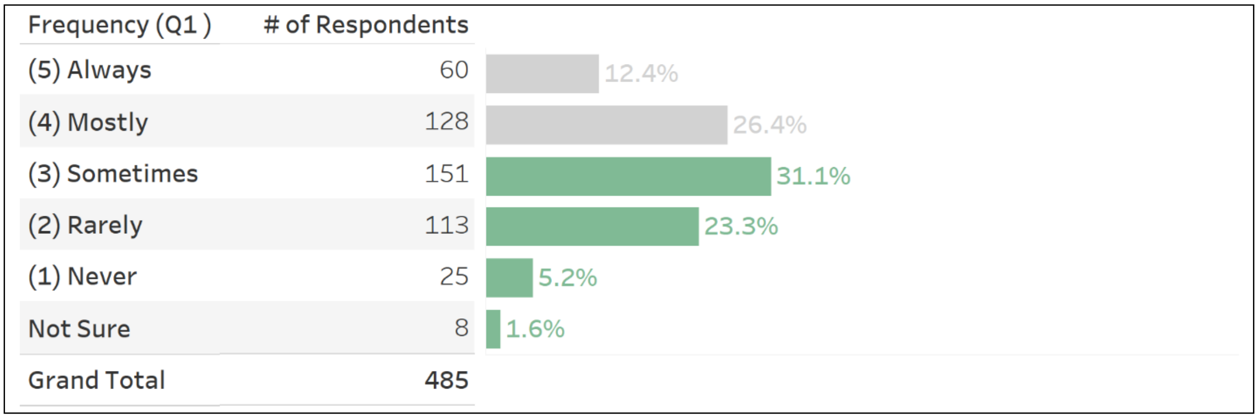

Figure 7. Q1: How often was an explicitly documented conceptual database model reflecting the requirements for the database (such as an ER Diagram) created, in cases when you were involved in building a database?

To gain a deeper understanding of the factors influencing the use of conceptual data modeling, question Q2 was posed: "In your best estimation, choose reasons (choose all that apply) that led to not creating an explicitly documented conceptual database model in cases when you were involved in building a database." In this question, respondents were able to choose from the list of eleven reasons. The responses to this question are summarized in Table 1, with a visual ranking of these results presented in Figure 8. The results are presented with reasons categorized as a) resources issues, b) methodology and perspective issues, c) work environment issues, d) informal conceptual data modeling, and e) other reasons. The responses reveal a variety of causes for the omission of conceptual data modeling in database projects. The most common reason, cited by 45.1% of respondents, was whiteboard conceptualizing without further documentation, indicating that this practice substituted for more formal conceptual data modeling. An almost as frequently cited reason, with 42.1% of respondents reporting it, was the lack of time to engage in conceptual data modeling. A lack of requirements (33.0%) and the absence of established practices either within the work environment (33.0%) or as required by clients (27.9%) were other frequently mentioned reasons. Additionally, some respondents pointed to the perception that conceptual diagrams were unnecessary (22.2%) and to a lack of knowledge about how to create conceptual database models (20.9%). Lesser factors included a lack of available software for creating conceptual database models (13.8%), budgetary constraints (8.8%), and the desire to reduce the number of diagrams (8.8%).

Table 1. Q2: In your best estimation, choose reasons (choose all that apply) that led to not creating an explicitly documented conceptual database model in cases when you were involved in building a database.

Reason	**#**	**%**
Not enough time	125	42.1%
Lack of requirements	98	33.0%
Lack of knowledge about how to create CDMs	62	20.9%
Lack of available software for creating CDMSs	41	13.8%
Budgetary constraints did not allow for CDM	26	8.8%
Desire to have fewer diagrams	26	8.8%
No need for conceptual diagrams	66	22.2%
CDM is not a part of established practice in my work environment	98	33.0%
CDM is not a part of established practice required by the clients	83	27.9%
Informal whiteboard conceptualizing was done and there was no formal CDM beyond it	134	45.1%
Other	24	8.1%
Unknown	14	4.7%

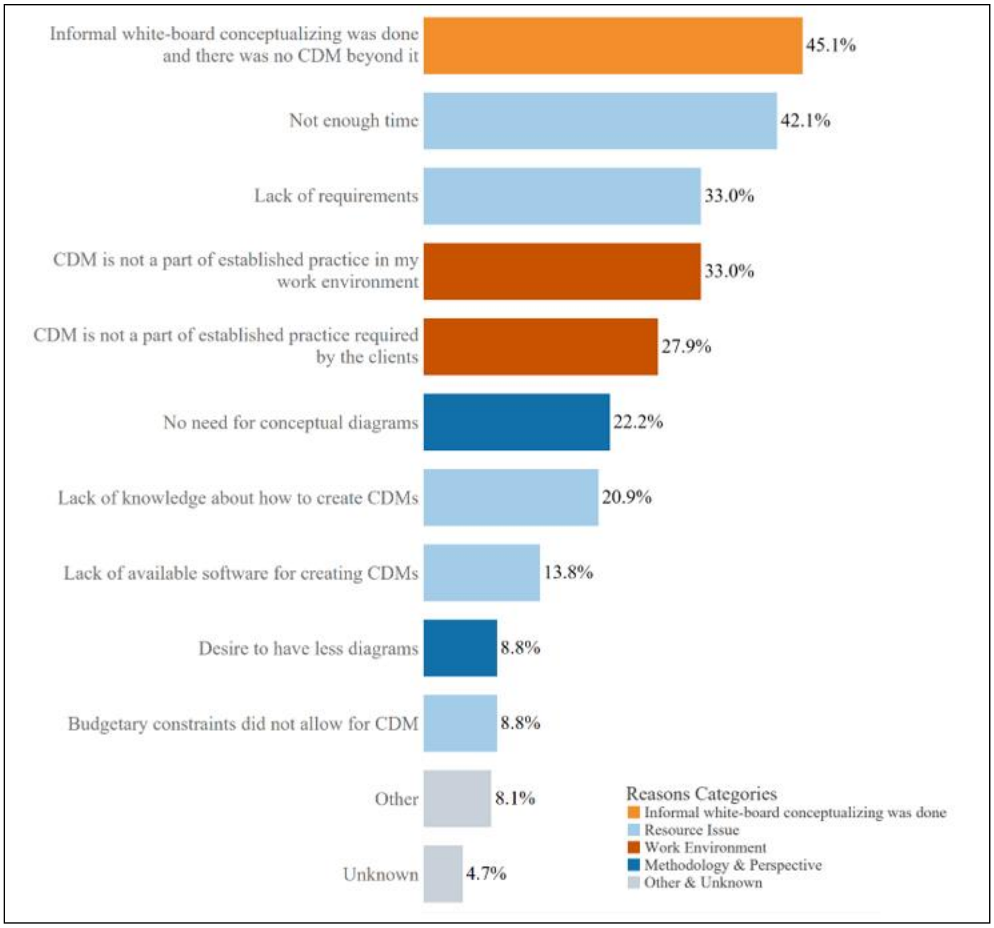

Figure 8. Ranking Q2

These results provided the foundational context for developing the interview protocols used in the subsequent qualitative phase of the study. Interviews were conducted to delve deeper into the underlying reasons and experiences that shaped the respondents' practices, offering further insights into the factors influencing the use of conceptual data modeling in database development. We present the analysis of interviews later in subsection 5.2.

Next, we discuss the results demonstrating the effect of the use of conceptual data modeling on the overall database development outcome satisfaction. The question Q3 asks: “How often were you satisfied with how the process of converting requirements to working databases turned out, in cases when you were involved in building a database?” Figure 9 shows the results for question Q3.

Frequency (Q3)	# of Respondents	
(5) Always	55	11.3%
(4) Mostly	262	54.0%
(3) Sometimes	114	23.5%
(2) Rarely	40	8.2%
(1) Never	3	0.6%
Not Sure	11	2.3%
Grand Total	485	

Figure 9. Q3: How often were you satisfied with how the process of converting requirements to working databases turned out, in cases when you were involved in building a database?

Less than 12% of respondents report being always satisfied with the outcome of converting requirements to working databases, and another 54% report being mostly satisfied. Close to 24% report being sometimes satisfied, while approximately 9% report being rarely or never satisfied with the outcome.

To examine the potential link of conceptual modeling with the outcome, we created a chart shown in Figure 10. This chart provides the average satisfaction scores grouped into five groups based on the respondents' Q1 answers. In other words, we show respondents' satisfaction scores based on how often they use conceptual modeling. The X-axis refers to Q1 (five groups based on the frequency of conceptual modeling), and the Y-axis refers to Q3 (database satisfaction score). The data points shown as circles represent average satisfaction scores for each of the five groups. The size of each circle represents the group size.

As shown in the chart in Figure 10, the average outcome satisfaction score of the respondents who always use database conceptual modeling is 4.25. This score drops down by over 16% to 3.77 (respondents who mostly use conceptual modeling), further 2.7% to 3.68 (respondents who sometimes use conceptual modeling), another 6.9% to 3.50 (respondents who rarely use conceptual modeling), and finally another 8.5% to 3.28 (respondents who never use conceptual modeling).

The trend in the chart shown in Figure 10 suggests that the use of conceptual database modeling may be linked to the outcome of the database design process, with more frequent use of conceptual database modeling appearing to be associated with higher satisfaction scores.

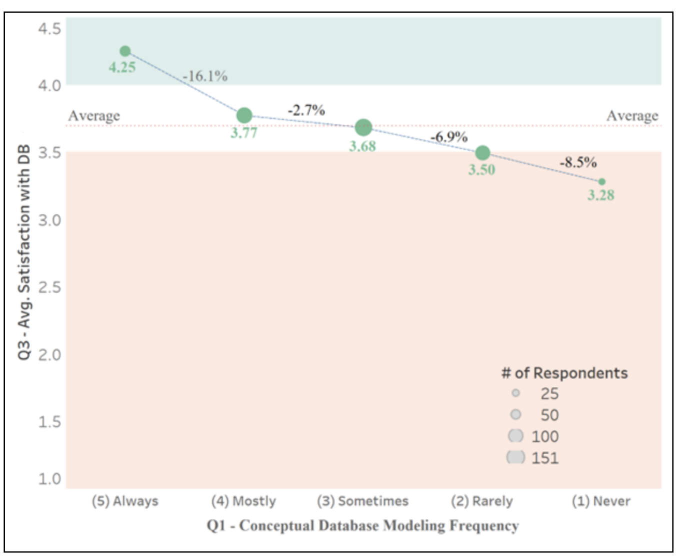

Figure 10. Outcome Satisfaction by Conceptual Data Modeling Frequency

We tested this association using regression analysis. The results of our exploratory regression analysis are shown in Table 2.

Table 2. Regression Analysis

Model:	**Q3=Constant + Q1**			
R-sq: 0.0839	Sig. < 0.0001	Mod DF:2	Res. DF: 470	
SSE: 283.287	MSE:0.6027	St. Error: 0.7764	N=472	
Variables	**Coeff.**	**St. Err.**	**t-stat**	**p-value**
Constant***	3.00405	0.11031	27.234	< .0001
Q1 ***	0.21525	0.03282	6.559	< .0001
Note: *** indicates significance at 99% level				

The N value in Table 2 is 470, as we excluded (from the total of 485) 15 observations associated with participants who selected 'Not sure' for either Q1 or Q3 in the chart shown in Figure 4.

The fitted regression model Q1 = 3.004 + 0.215* (Q1) is statistically significant (R-squared = 0.084, F (2, 470) = 0.776, p < .0001), suggesting that over 8.4% variance in satisfaction with how the process of converting requirements to working databases (Q3) is explained by the model (variance in conceptual modeling practice regularity/frequency (Q1)). Given the ubiquity and importance of database solutions, this potential association should be investigated in future research.

5.2 Qualitative Responses

The survey provided data about the incidence of various reasons that result in database practitioners omitting formal conceptual modeling. This quantitative approach allowed us to verify that surprisingly many practitioners indeed do not use formal conceptual modeling during the development of database systems. However, some of the given reasons opened new questions. For example, why do the practitioners claim that budgetary reasons and lack of available software are common reasons for not doing conceptual modeling when free and widely used open-source tools exist? Following an explanatory mixed methods approach, to further validate the findings of the survey, as well as to enable discovering more intricate findings, we conducted semi-structured follow-up interviews with selected survey respondents.

We initiated this process by having a pool of 240 survey respondents who had provided their email addresses for potential follow-up inquiries. A random sample of these participants was then selected for interviews. This sample included a mix of respondents who reported consistently using conceptual data modeling and respondents who stated in the survey that they do not consistently use conceptual data modeling. Interviews continued until thematic saturation was reached, meaning no new insights were emerging. In total, 34 individuals were interviewed. Within this group, we were able to address questions about each of the reported reasons during the survey and develop themes for each of the reasons.

The overwhelming sentiment of the interviewees was that conceptual data modeling is a prudent and useful part of the database system development cycle. This sentiment was shared equally by interviewees who consistently engage in conceptual data modeling and the interviewees who, for a variety of reasons, do not consistently engage in conceptual data modeling. One of the main goals of the interviews was to shed more light on the reasons that prevent database developers from consistently using conceptual data modeling. In the subset of survey respondents that we were able to interview, we were able to identify multiple interviewees for every one of the reasons listed in Table 1 and Figure 8. This gave us an opportunity to ask those respondents to elaborate more on why they selected a particular reason. We asked them to describe their reasons in their own words, and we recorded the explanations they gave us.

As we stated earlier, the responses summarized in Table 1, with a visual ranking of these results presented in Figure 8, were categorized as a) resources issues, b) methodology and perspective issues, c) work environment issues, and d) informal Conceptual data modeling. In this section, we will summarize the findings of the interviewing process for each of these categories. We will also summarize the qualitative responses from the respondents who reported their reasons as e) other.

Resources Issues

Practitioners who consistently engage in conceptual data modeling report that appropriating sufficient time and expertise for conceptual data modeling is one of their main reasons for the success of the entire database development process. They also report that motivated and knowledgeable practitioners can use any tools, even the most basic ones, for conceptual data modeling. On the other hand, among respondents who do not consistently engage in conceptual data modeling, problems were reported with each of the reasons in the resources issues category: lack of time, lack of requirements, lack of knowledge about how to create conceptual database models, lack of available software for creating conceptual database models, and budgetary constraints. Table 3 presents a summary of the interview findings related to the reported resource issues.

Table 3. Interviews: Elaboration on Reported Resources Issues from Table 1/Figure 8

Not enough time
Themes: In some cases, in deadline intensive situations, CDM overlooked. Under time pressure, features prioritized over CDM.
Selected Quotes: *"We find ourselves in situations when we have to get the database out in a hurry and there is just not enough time to do full documentation including the CDM."* *"In consulting there is often a firm timeline. Scope of the work gets underestimated, and in such cases the documentation, including the CDM, ends up not being a priority."* "*We had some time to develop the initial CDM and we needed more time to complete it as we became aware of the scope that exceeds initial expectations. However, the deadline never changed so we had to proceed with an incomplete CDM."* *"Often businesses want to see something that provides value very quickly. CDM doesn't represent a return that, for example, a dashboard does."*
Lack of requirements
Themes: Some projects experience difficulties engaging with all necessary constituents during the requirements process. Relying on intermittent feedback and intuition to fill in the gaps in requirements.
Selected Quotes: *"We would love to have clear requirements every time we build a database. Who wouldn't? However, that is simply not what happens in most database development projects with us. We try to define the requirements and have detailed conversations with all constituents, but organizing proper appointments for that purpose is almost impossible, due to scheduling issues, deadlines, availability of people, other priorities, etc. Therefore, we proceed in the 'best of my knowledge' or 'this would be a useful thing' mode. This of course, doesn't result in a proper CDM process prior to development".* *"Things are ever changing in our database projects. You build a little bit from some initial guidelines. Then you show it to the client, and they tell us what to change and add, and that goes on. We end up with a product without having formal requirements or CDM."* *"Sometimes stakeholders often don't know what they need and want, but at the same time they want you to develop something. You end up developing databases without requirements or a CDM."* *"All the 'requirements' were high-level without nearly enough detail. It was a guessing game from the get-go. We managed to get some feedback from the future users here and there as we were moving along with the development without a CDM."*
Lack of knowledge about how to create CDMs
Themes: Some projects have participants in the database development process who do not have database modeling educational background or meaningful CDM experience.

Selected Quotes: *"Some people who work in database development just fall into this job without studying it in school. We have developers who never studied it in school, coming in with other backgrounds such as marketing, and yet we have to use them in our database development projects."* *"Ideally, we would love to have all our database developers with a proper CDM educational background. However, that is not always the case. Also, even the people who were trained in CDM often are not versed with due to the low emphasis on it in their careers."* *"At one company I worked for, a data company was commissioned where 'database consultants' with supposed excellent domain knowledge had no clue how to do CDM, and yet we had to defer to them when it came to the database design. The project was a total failure."* *"People with various educational backgrounds (math, business, even MIS) that were a part of the database development team simply did not know how to do CDM."*
Lack of available software for creating CDMSs
Themes: In some cases, available tools are perceived as not having all the necessary functionalities. Open-source and free tools are not easily approved for use.
Selected Quotes: *"It is not easy to bring in free and open source software into an enterprise, due to security reviews, legal issues, etc. Free tools can have limited functionality when you are getting into real complex projects. Commercial tools often do not accommodate pure CDM."* *"Most tools for ER Modeling are cumbersome for making formal handoff CDM documents that look professional enough."* *"In my publicly traded company we can't use any tool that we want. Using using-open source tools is especially problematic due to all kinds of legal reasons."* *"We are highly restricted with what tools we are allowed to use, even if they are available on the market, and the process for approval is not very quick or straightforward."*
Budgetary constraints did not allow for CDM
Themes: In some cases, budgets mostly focused on implementation, overlooking documentation and CDM.
Selected Quotes: *"Some projects do not pay enough to put people's time into anything but implementation. In those cases, we try to redirect a portion of the budget towards the CDM, but we are not always successful"* *"Core work of implementing databases is what we are paid for. Sometimes documentation work is not in the budget."* *"What helps users is higher budgetary priority than what helps developers."* *"Both budget and time make us sacrifice the CDM process. We have to work too fast, and we are not given enough paid time to model."*

Methodology and Perspective Issues

Practitioners who consistently engage in conceptual data modeling report that creating conceptual database models is part of their documentation methodology and practice. Among the respondents who do not consistently engage in conceptual data modeling, the desire to have fewer diagrams and the lack of need for conceptual modeling was reported. Table 4 presents a summary of the interview findings related to the methodology and perspectives issues.

Table 4. Interviews: Elaboration on Reported Methodology and Perspective Issues from Table1/Figure8

Desire to have fewer diagrams
Themes: In circumstances when CDM is perceived as not feasible, it is deliberately omitted.
Selected Quotes: *"When I am doing a quick project with a tight deadline, I will intentionally skip everything other than the development".* *"There is very little focus on documentation in general in the database projects I am involved with. We end up with less diagrams, whether we actually decided to not have them or whether it is just a matter of other factors, such as time pressure."* *"It is a battle just to have an updated detailed documented relational schema in my environment. There is no way we could also create and maintain a real CDM in our circumstances, so we skip it."* *"When there is not good way to store and manage created CDMs we deliberately create fewer diagrams."*
No need for conceptual diagrams
Themes: In some cases, in quick, simple, and well-understood projects, CDM is not perceived as necessary. Also, in some cases, developers are comfortable skipping CDM and proceeding with logical models.
Selected Quotes: *"In cases when requirements are brief and simple we are able go straight into the development without creating other documentation, including the CDM."* *"In small scope projects that are intuitive and based on well understood domain knowledge, we can get away without CDM."* *"With strong requirements and good architects, I've always been able to skip the CDM and go right to logical model"* *"For us the speed is everything and we are able to do quick projects without CDM."*

Work Environment Issues

Practitioners who consistently engage in conceptual data modeling report that their work environments foster the creating of conceptual database models. Among the respondents who do not consistently engage in Conceptual data modeling, both CDM not being a part of established practice in their work environments, and CDM not being a part of established practice required by the clients, was reported. Table 5 presents a summary of the interview findings related to the reported work environment issues.

Table 5. Interviews: Elaboration on Work Environment Issues from Table 1/Figure 8

CDM is not a part of established practice in my work environment
Themes: In some work environments, not engaging in CDM is a default option.
Selected Quotes: *"We have a development group that simply will not do CDM. Their approach is 'tell me what tables to build and where to put them'".* *"In my last work environments, CDM was just not seriously practiced. That is one of the reasons why I left for another job."* *"In my company, we do rough drafts but no formal CDMs. It is just not part of the culture."* *"I have been part of the environment where every developer went straight to the relational schemas. I was a young developer, and I went along."*
CDM is not a part of established practice required by the clients
Themes: In some cases, clients of database projects are either unaware of CDM or do not see a value in it.

Selected Quotes: *"CDM of the database often doesn't sound valuable to the client, because once you show it to them, they say 'this is my data, I already knew this' and they are not willing to pay for it."* *"Most clients I worked with on database development projects flat out do not know about CDM, and if my organization doesn't bring it up with them (which it never does) CDM is not done."* *"We often come to clients where CDM is not a part of their database development lifecycle and then we try to introduce it to them, so we can proceed in a professional manner."* *"Our client developed things on the fly without CDM and they ended up with a database nobody understood. When they brought us in, they were not willing to pay for the CDM, so we didn't make it for them. We made one for ourselves (just so we would know what to do) but my management didn't let me share it with the client."*

Informal conceptual data modeling

Practitioners who consistently engage in conceptual data modeling report that a documented formal conceptual database model is part of their practice, even when it is preceded by informal whiteboarding. Among the respondents who do not consistently engage in conceptual data modeling, creating informal whiteboard conceptualizations that are not documented and do not lead to a formal conceptual database model was a common theme. Table 5 presents a summary of the interview findings related to the reported conceptual database model.

Table 5. Interviews: Elaboration on Work Environment Issues from Table 1/Figure 8

Informal whiteboard conceptualizing was done and there was no formal CDM beyond it
Themes: In some cases, whiteboard conceptualizing is not followed by formal CDM, and the result of whiteboarding is the only attempt at conceptualization.
Selected Quotes: *"When you are whiteboarding a database model during the discovery stage, it often gets lost during the development phase."* *"In our biggest project, for two weeks we did whiteboarding of main concepts with the senior leadership. Once we got the database project the results were not used to create a master CDM. Instead, various teams took different bits straight into the implementation phase".* *"It is important to conceptualize in the beginning of the database development process, so we know the direction of the database project. Once we get into implementation, we make too many changes in order to go back and document them in a formal CDM."* *"In many of my projects there was some rigor in the beginning of the project, including some detailed whiteboarding of the CDM, but as the time went by it was not followed by formal conceptual modeling, mostly because delivering actionable items became main priority".*

Other Reasons

Among the respondents who do not consistently engage in conceptual data modeling, a small group of respondents (less than 9%) listed other reasons. The analysis of the reported reasons revealed that most reasons reported as "Other" actually fit into the categories of reasons listed in Figure 8/Table 1 and analyzed above. For example, *"Lack of employees with CDM training"* was reported as "Other", but it actually fits the reason "Lack of knowledge about how to create CDMs" analyzed in Table 3. In another example, "*Clients were told how long it will take to build a CDM and then basically said it was not in the budget*" was also reported as "Other", but it actually fits the reason "Budgetary constraints", also analyzed in Table 3.

One other issue that was discussed by several respondents was reverse engineering the conceptual database model. A few respondents reported using reverse engineering during the development as a method for building, supplementing, and/or verifying conceptual database models. Some respondents also reported using reverse engineering to understand and conceptualize what was already previously implemented but left unaccompanied by documentation.

6 Discussion

The goal of this research was to add to the understanding of the conceptual data modeling use that goes beyond intuitions and assumptions about whether, how, and why it is used or not used.

First, we wanted to establish to what extent conceptual data modeling is used in practice. Even though conceptual data modeling has been taught for decades in database courses, our survey results show that a minority of practitioners create documented conceptual models regularly when developing databases. While this finding may be surprising to some members of the database community, and others may view it as obvious, with this result, we have unbiased information that was not previously available.

Next, we wanted to establish the reasons for not using conceptual data modeling among the majority of the practitioners who do not use it consistently. Our pre-survey study identified various reasons for not creating documented conceptual database models and classified them into different categories. The survey results revealed that in most cases, the developers are not opposed to using conceptual data modeling. In fact, as our analysis of survey data demonstrated, most of the respondents who do not use conceptual data modeling regularly do not think that conceptual modeling is unnecessary. Even when not producing documented conceptual database models, many of them engage in less formal conceptual modeling. Also, our survey findings show that database designers often face issues related to resources and workplace practices that simply prevent them from engaging in conceptual data modeling. By identifying, classifying, and quantifying the reasons for non-use, we provide additional perspective and information about conceptual data modeling practices.

However, the survey results themselves do not completely answer the question: why don’t practitioners engage in formal conceptual data modeling, even in cases when they would like to do so? Therefore, following the survey and the analysis of its results, we conducted interviews with selected survey respondents in order to get more detailed and descriptive answers to this question.

These interviews provided themes and narratives behind the reasons quantified in the survey. When it comes to the resources’ issues of lack of time, lack of requirements, lack of knowledge, and lack of available software, the interviews revealed several themes. According to the interviewees’ testimonies, in deadline-intensive situations, time pressures can result in conceptual data modeling being deprioritized or overlooked. Furthermore, interviewees elaborated on how scarce or incomplete requirements present an impediment to the conceptual data modeling process. They reported on experiencing difficulties engaging with all necessary constituents during the requirements process and having to rely on intermittent feedback and intuition to fill in the gaps in requirements. Such lack of mutual detailed consensus on requirements between the users/clients and the developers during the requirements’ engineering process, results in inconsistencies, omissions, and ambiguities (Dawson & Swatman, 1999), and it has unfortunately been a constant and well-documented problem in software and information systems projects, as evidenced by publications dating from decades ago (DeMarco 1982; Jackson 1998).

Interviewees also reported having to work on database development projects with participants lacking the necessary conceptual data modeling knowledge and skills due to not having a database modeling educational background or meaningful conceptual data modeling experience. They also reported difficulties in using the conceptual data modeling software. The most cited software-related problems were the inability to get approval for the use of open-source and free tools due to legal and security reasons, and the perception of available tools not having all the necessary functionalities. And finally, when it comes to the resources-related issues, interviewees testified about the occurrences of budgets focused mainly on implementation, overlooking financial resources needed for paid documentation and conceptual data modeling work.

Themes revolving around the conceptual data modeling methodology and perspective issues also emerged during the interviews. Under certain circumstances, such as short projects with tight deadlines or lack of an organized system for storing and maintaining diagrams, some interviewees reported deliberately omitting conceptual data modeling altogether. Also, some interviewees stated that in certain quick, simple, and well-understood projects, conceptual data modeling may not be necessary. In addition, some developers reported being comfortable skipping conceptual data modeling and proceeding with logical models when strong requirements and good architects are available.

The interviewees also discussed how work environment issues may affect the use of conceptual data modeling. Some interviewees report being in work environments where not engaging in conceptual data modeling is a default option during database development projects. In such environments, creating relational schemas without a conceptual model or even writing SQL DDL statements without any model is a standard practice. They also reported that, in some cases, clients of database projects are either unaware of conceptual data modeling or do not see a value in it. In such cases, clients may be reluctant to approve the time and financial resources needed to engage in conceptual data modeling.

Also discussed by the interviewees were cases and scenarios when informal whiteboard conceptualizing is not followed by formal conceptual data modeling. This happens in projects that do not have an enforced established system or prescribed practice for following up initial whiteboarding with documented conceptual data modeling. In such cases, the result of whiteboarding ends up as the sole form of conceptualization in the database development project.

Finally, we also examined to what extent the use (or non-use) of conceptual data modeling usage affects the outcome of converting requirements to working databases. We found out that the more regular practice of creating documented conceptual database models is correlated with higher satisfaction with the process of creating working databases. This correlation between proper conceptual modeling and the positive outcomes of database development projects in practice may be expected (especially by database educators and academics who consider it an essential part of the database development lifecycle), but it was still important to establish it and support it by evidence, especially considering the widespread non-use of conceptual modeling.

7 Conclusions, Implications, Limitations, and Future Work

This paper focused on establishing and researching what practitioners think are the main reasons for not doing conceptual data modeling. The survey was created based on the practitioners' input and it was answered by the practitioners. It was followed up by interviews with the practitioners. We wanted to hear, report, and analyze their input, not influence it. This is not a paper that purports to cover every possible aspect and variation of conceptual data modeling. This is a paper that reports on what the practitioners collectively say that they do and don't do when it comes to conceptual data modeling, and what are their reasons for it.

The findings of this research offer important implications for both database practitioners and educators, such as the following:

1. **Academic Database Education**: Despite the long-standing emphasis on conceptual data modeling in education and research, this research reveals its limited use in practice and highlights a gap between academic database education and real-world practices, where barriers, such as time pressure, incomplete requirements, lack of knowledge, and software limitations, often inhibit the use of conceptual data modeling. This suggests a need for academics to supplement their database classes with readings, discussions, and scenarios that reveal these issues that impede the use of conceptual data modeling in practice. It is paramount that students in database courses are taught in detail about conceptual data modeling. However, they should also be told about the obstacles that may await them when they try to apply what they learned in their classes in their workplaces. Forewarned is forearmed.
2. **Organizational and Corporate Database Training:** Organizations and corporations whose employees are engaged in database design and development should engage in continuous assessment and training of those employees. Employees who are thrust into database design roles for which they are not qualified should be identified and then supported by proper training that includes detailed lessons about methods and practices of conceptual data modeling (including the overview of obstacles mentioned in the previous paragraph). This should be followed up by practical exercises and mentored real-project experiences, where this knowledge can be applied and brought up to the professional level. The same opportunities should be given to employees who are theoretically qualified but whose skills deteriorated or atrophied due to the lack of use.
3. **Database Project Documentation:** Consistent database project documentation should be used during the requirements stage, even in cases when database developers experience difficulties engaging with all necessary constituents and rely on intermittent feedback and intuition to fill in the gaps in requirements. Every database project that results in a developed database has requirements, whether these requirements are properly documented and reflected in a formal conceptual data model or just implied. In cases when the database is developed based on incomplete feedback and intuition, without documented requirements and conceptual data model

(i.e., implied requirements), requirements and conceptual data model can be reverse-engineered and presented to the clients to inform them on what was actually developed. The clients can even be asked to give their approval at that stage or suggest changes, which would also have to be documented. This could, in a circuitous way, result in proper database development practices, including the creation of a formal conceptual data model.

4. **Database Project Budgeting**: Budgeting of database projects should include, in advance, an itemized part dedicated to database requirements and conceptual data modeling, which should account for a percentage of the entire project budget. This would prevent the situations described in this paper when budgetary constraints and the lack of valuation of conceptual data modeling by clients result in the absence of conceptual data modeling. When clients are not aware or convinced about the benefits of dedicating some portion of the overall budget towards conceptual data modeling, efforts should be made to demonstrate the long-term benefits of proper conceptual modeling, particularly in improving satisfaction and project outcomes. Once conceptual modeling is properly budgeted for, it will also be more likely to be given enough time to accomplish.
5. **Database Modeling Software:** In addition to providing ample time and personnel for conceptual data modeling, budgeting should also cover the necessary software. Not every project will be able to afford professional database design software tools. Still, if there is a serious commitment to engaging in conceptual data modeling, the task can also be undertaken using generic drawing tools (such as Visio, Lucidchart, Draw.io) or free educational tools (such as ERDPlus). Whichever tools are selected, any challenges regarding internal approval for the use of such tools must be dealt with prior to engaging in projects.

In summary, while conceptual database modeling is often deprioritized due to practical constraints, its documented benefits suggest that organizations should consider strategies to address these barriers, whether through better training, resource allocation, or adjustments in project timelines. Educators, too, should be aware of these trends and adjust their teaching to reflect the realities of industry practice. They should continue to emphasize the use of conceptual modeling as an important and essential part of the database documentation and development process and resist pressures to minimize it or skip it when teaching the database development process. At the same time, they should be aware of the reasons that may prevent practitioners from engaging in formal conceptual data modeling even when they would like to do so and share them with their students.

We hope that the research presented here will serve as an additional clear and data-supported resource in situations and cases when database development projects are planned, discussed, and managed. In addition, this research can also serve to supplement database educational efforts. And finally, we also hope that this paper helps with and motivates further research on database development and design practices.

One limitation of this research is that the survey sample is mostly composed of respondents from North America. In our future research, we would like to create a follow-up survey of other regions and examine the possibility of regional biases. Another direction for our future research is to expand the survey to examine the practices and user perspectives within other parts of the database system development cycle, beyond conceptual data modeling.

References

Abrial, J. (1974). Data Semantics. In J. W. Klimbie & K. L. Koffemen (Eds.), *Data base management* (pp. 1-59). North-Holland.

Anglim, B., Milton, S., Rajapakse, J., & Weber, R. (2009). Current trends and future directions in the practice of high-level data modeling: An empirical study. In *ECIS 2009, 22* (pp. 122-133). https://aisel.aisnet.org/ecis2009/22

Baghizadeh, Z., Cecez-Kecmanovic, D., & Schlagwein, D. (2020). Review and critique of the information systems development project failure literature: An argument for exploring information systems development project distress. *Journal of Information Technology, 35*(2), 123-142.

Batra, D., & Davis, J. G. (1992). Conceptual data modelling in database design: Similarities and differences between expert and novice designers. *International Journal of Man-Machine Studies, 37*(1), 83–101.

Batra, D., & Marakas, G. M. (1995). Conceptual data modelling in theory and practice. *European Journal of Information Systems, 4*(3), 185–193.

Brosey, M., & Shneiderman, B. (1978). Two experimental comparisons of relational and hierarchical database models. *International Journal of Man-Machine Studies, 10*(6), 625–637.

Chen, G., Lin, L., & Guo, X. (2006). Introducing λ-Specialization into the Fuzzy EER Model. In *Modern Information Processing* (pp. 333–343). Elsevier.

Chen, P. P.-S. (1976). The entity-relationship model—Toward a unified view of data. *ACM Transactions on Database Systems* (TODS), *1*(1), 9–36.

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, *13*(6), 377–387.

Davies, I., Green, P., Rosemann, M., Indulska, M., & Gallo, S. (2006). How do practitioners use conceptual modeling in practice? *Data & Knowledge Engineering, 58*(3), 358–380.

Dawson, L. and Swatman, P. (1999). The use of object-oriented models in requirements engineering: A field study. In *ICIS 1999 Proceedings, 23*. https://aisel.aisnet.org/icis1999/23

DeMarco, T. (1982). *Controlling software projects: Management, measurement and estimation*. Yourdon Press.

Elmasri, R., & Navathe, S. B. (1988). *Fundamentals of database systems*. Benjamin Cummings Publishing Co.

Elmasri, R., & Navathe, S. B. (2015). *Fundamentals of database systems* (7th ed.). Pearson.

Embley, D. W., & Liddle, S. W. (2013). Big data–Conceptual modeling to the rescue. In *Proceedings of the 32nd International Conference on Conceptual Modeling*, *8217*, 1–8.

Fettke, P. (2009). How conceptual modeling is used. *Communications of the Association for Information Systems, 25*(1), 43.

Hammer, M., & Mc Leod, D. (1981). Database description with SDM: A semantic database model. *ACM Transactions on Database Systems* (TODS), *6*(3), 351–386.

Hitchman, S. (2003). An interpretive study of how practitioners use entity-relationship modelling in a ternary relationship situation. *Communications of the Association for Information Systems, 11*(1), 26.

Hoffer, J., Venkataraman, R., & Toppi, H. (2021). *Modern database management* (13th ed.). Pearson Education Limited.

Hull, R., & King, R. (1987). Semantic database modeling: Survey, applications, and research issues. *ACM Computing Surveys* (CSUR), *19*(3), 201–260.

Hvalshagen, M., Lukyanenko, R., & Samuel, B. M. (2023). Empowering users with narratives: Examining the efficacy of narratives for understanding data-oriented conceptual models. *Information Systems Research, 34*(3), 890-909.

Jaakkola, H., & Thalheim, B. (2020). Sixty years – and more – of data modelling. In M. Tropmann-Frick, B. Thalheim, H. Jaakkola, Y. Kiyoki, & N. Yoshida (Eds.), *Frontiers in artificial intelligence and applications*. IOS Press.

Jabbari, A., & Rosemann, M. (2023). Modeling 4.0: Conceptual modeling in a digital era. *Communications of the Association for Information Systems, 53*(1), 42.

Jabbari, M., & Recker, J. (2017). Combined use of conceptual models in practice: An exploratory study. *Journal of Database Management* (JDM), *28*(2), 56–88.

Jabbari, M., Lukyanenko, R., Recker, J., Samuel, B., & Castellanos, A. (2017). Conceptual modeling research in information systems: What we now know and what we still do not know. In *Proceedings of the 16th AIS SIGSAND Symposium*.

Jackson, M. C. (1998). Critical systems thinking and information systems development. In *Proceedings of Eighth Australasian Conference on Information Systems, University of South Australia, Adelaide, South Australia* (pp. 1-20).

Jarvenpaa, S. L., & Machesky, J. J. (1989). Data analysis and learning: An experimental study of data modeling tools. *International Journal of Man-Machine Studies, 31*(4), 367–391.

Jukic, N., Vrbsky, S., Nestorov, S., & Sharma, A. (2021). *Database systems—Introduction to databases and data warehouses* (2nd ed.). Prospect Press.

Khatri, V., Vessey, I., Ramesh, V., Clay, P., & Park, S. J. (2006). Understanding conceptual schemas: Exploring the role of application and IS domain knowledge. *Information Systems Research*, *17*(1), 81-99.

Peckham, J., & Maryanski, F. J. (1988). Semantic data models. *ACM Computing Surveys* (CSUR), *20*, 153–189.

Recker, J. C., Lukyanenko, R., Jabbari Sabegh, M., Samuel, B., & Castellanos, A. (2021). From representation to mediation: A new agenda for conceptual modeling research in a digital world. *MIS Quarterly, 45*(1), 269–300.

Siau, K. (2004). Informational and computational equivalence in comparing information modeling methods. *Journal of Database Management* (JDM), *15*(1), 73–86.

Siau, K., & Cao, Q. (2001). Unified modeling language: A complexity analysis. *Journal of Database Management* (JDM), *12*(1), 26–34.

Smith, J. M., & Smith, D. C. P. (1977). Database abstractions: Aggregation and generalization. *ACM Transactions on Database Systems* (TODS), *2*(2), 105–133.

Song, I.-Y., Evans, M., & Park, E. K. (1995). A comparative analysis of entity-relationship diagrams. *Journal of Computer and Software Engineering, 3*, 39.

Spurrier, G., & Topi, H. (2021). *Systems analysis and design in an age of options*. Prospect Press.

Storey, V., Chiang, R. L., D, D., Goldstein, R.C., & Sundaresan, S. (1997). Database design with common sense business reasoning and learning. *ACM Transactions on Database Systems, 22*(4), 471-512.

Thalheim, B. (2013). *Entity-relationship modeling: Foundations of database technology*. Springer Science & Business Media.

Topi, H., & Ramesh, V. (2002). Human factors research on data modeling: A review of prior research, an extended framework and future research directions. *Journal of Database Management* (JDM), *13*(2), 3-19.

Venkatesh, V., Brown, S. A., & Bala, H. (2013). Bridging the qualitative-quantitative divide: Guidelines for conducting mixed methods research in information systems. *MIS Quarterly*, *37*(1), 21-54.

Yang, H. L. (2003). Comparing relational database designing approaches: Some managerial implications for database training. *Industrial Management & Data Systems*, *103*(3), 150-166.

Appendix A: Example of a Generic Company Developing a Small Database

Step 1 - Requirements

The database is based on the requirements which are collected and documented, as follows:

- *The database will keep track of employees, facilities, and devices.*
- *For each employee, the database will keep track of a unique employee identifier as well as their name.*
- *For each facility, the database will keep track of a unique facility identifier as well as the facility type.*
- *For each device, the database will keep track of a unique device identifier as well as the device type.*
- *Each employee can access one or more facilities. Each facility can be accessed by one or more employees.*
- *Each device is issued to exactly one employee. Each employee can have between zero and many devices issued to them.*

Step 2 – Conceptual Database Model

The requirements shown in Step 1 are visualized as a conceptual database model, shown in Figure A1.

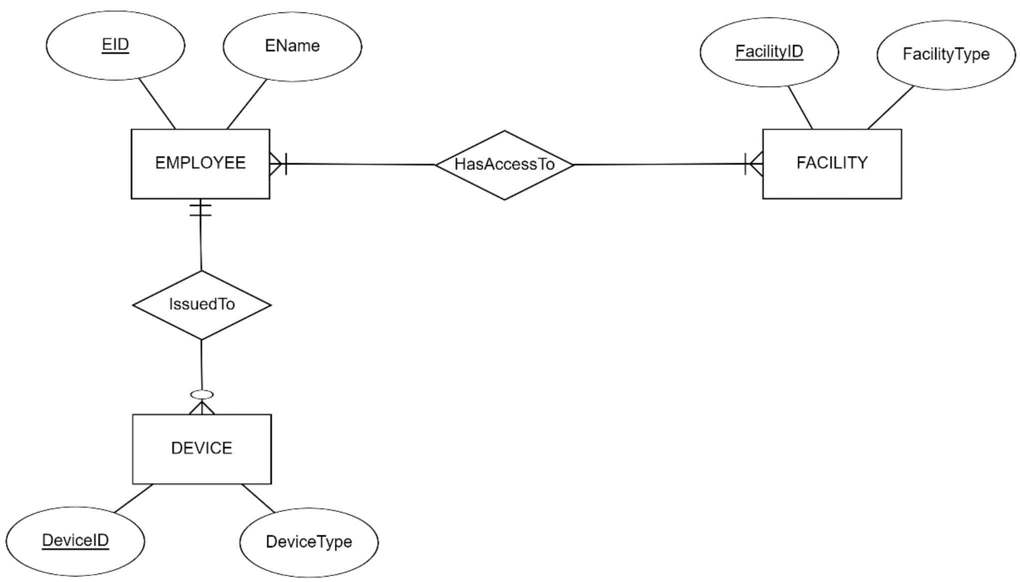

Figure A1. Example Conceptual Database Model Implemented as an ER Diagram

ER Modeling (Chen, 1976) is a common conceptual data modeling technique. The diagram shown in Figure 1 is an ER Diagram, a result of conceptual data modeling based on the requirements shown in Step 1.

Step 3 – Logical Database Model

The logical (implementational) database model, shown in Figure A2, is created as a relational schema mapped from the conceptual database model created in Step 2. Primary keys are bold and underlined.

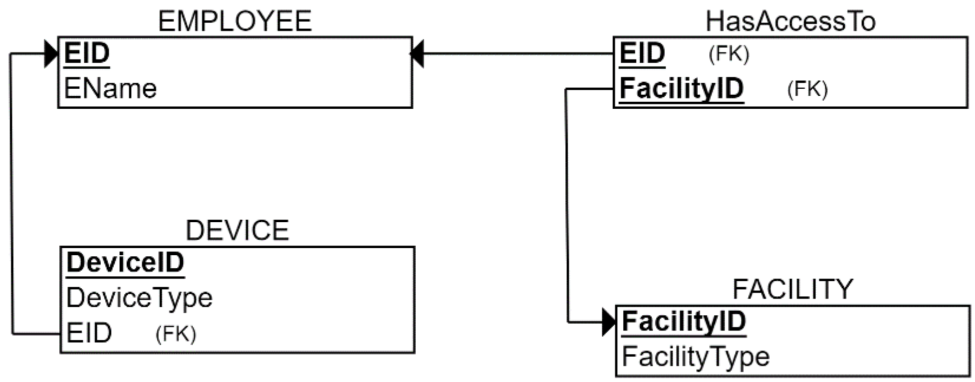

Figure A2. Example Logical Database Model Implemented as a Relational Schema

Step 4 – Implementation

The database, whose relational schema is shown in Step 3, is implemented using the following SQL statements that are executed in a Relational DBMS (RDBMS), such as Oracle, PostgreSQL, SQL Server, or similar.

```
CREATE TABLE EMPLOYEE
(
  EID INT NOT NULL,
  EName CHAR(20) NOT NULL,
  PRIMARY KEY (EID)
);

CREATE TABLE FACILITY
(
  FacilityID INT NOT NULL,
  FacilityType CHAR(15) NOT NULL,
  PRIMARY KEY (FacilityID)
);

CREATE TABLE DEVICE
(
  DeviceID INT NOT NULL,
  DeviceType CHAR(20) NOT NULL,
  EID INT NOT NULL,
  PRIMARY KEY (DeviceID),
  FOREIGN KEY (EID) REFERENCES EMPLOYEE(EID)
);

CREATE TABLE HASACCESSTO
```

```
(
  EID INT NOT NULL,
  FacilityID INT NOT NULL,
  PRIMARY KEY (EID, FacilityID),
  FOREIGN KEY (EID) REFERENCES EMPLOYEE(EID),
  FOREIGN KEY (FacilityID) REFERENCES FACILITY(FacilityID)
);
```

About the Authors

Nenad Jukić is a Professor of Information Systems and a Director of Master of Science in Information System and Analytics at the Quinlan School of Business at Loyola University Chicago. Dr. Jukić conducts research in various information management–related areas, including Database Systems, Data Warehousing, Data Analytics, Big Data, Data Mining, Information Systems Strategy, and Information Systems Education. His work has been published in numerous management information systems and computer science academic journals, conference publications, and books. In addition to his academic work, Dr. Jukic provides expertise to database, data warehousing, business intelligence, and Big Data projects for corporations and organizations that vary from startups to Fortune 500 companies to U.S. government agencies. He is a co-founder of ERDPlus.com, a free web-based database modeling software suite used by universities and corporations in more than 100 countries.

Abhishek Sharma is a Clinical Professor of Information Systems at the Quinlan School of Business at Loyola University Chicago. Parallel with his consulting work and teaching, he researches statistics, artificial intelligence, forecasting, data architecture, and machine learning.

Dinko Bačić is an Assistant Professor of Information Systems at Loyola University Chicago's Quinlan School of Business and the founder of the UX & Biometrics (UXB) lab. His research interests include data visualization, human-computer interaction, biometrics, cognition, neuro IS, business intelligence & analytics, and pedagogy. He has papers published in premier journals such as Decision Support Systems, Behaviour & Information Technology, Communications of the Association for Information Systems, AIS Transactions on Human-Computer Interaction, Leonardo, Journal of Information Systems Education, and International Journal of Information Technology & Decision Making, among others. He has over fifteen years of corporate and consulting experience in business intelligence, finance, project management, and human resources.

Sippo Rossi is an Assistant Professor at Hanken School of Economics in Helsinki, Finland. His research interests include databases, social media, and AI in healthcare. His work has been published in journals such as Decision Support Systems and the International Journal of Information Management.

Svetlozar Nestorov is an Associate Professor of Information Systems in Loyola University Chicago's Quinlan School of Business. His research and teaching interests are in databases, data mining and analytics, data visualization, education, and e-commerce. He has published articles in international journals and conferences including Decision Support Systems, Information Systems Management, Journal of Database Systems, ACM SIGMOD Record, PLoS Computational Biology, and Computing in Science and Engineering.