Engaging Databases for Data Systems Education

Taipalus, Toni; Miedema, Daphne; Aivaloglou, Efthimia

DOI
10.1145/3587102.3588804

Publication date
2023

Document Version
Final published version

Published in
ITiCSE 2023 - Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education

Engaging Databases for Data Systems Education

Toni Taipalus
toni.taipalus@jyu.fi
University of Jyväskylä
Finland

Daphne Miedema
d.e.miedema@tue.nl
Eindhoven University of Technology
the Netherlands

Efthimia Aivaloglou
e.aivaloglou@tudelft.nl
Delft University of Technology
the Netherlands

ABSTRACT

Querying a relational database is typically taught in practice by using an exercise database. Such databases may be simple toy examples or elaborate and complex schemas that mimic the real world. Which of these are preferable for students is yet unknown. Research has shown that while more complex exercise databases may hinder learning, they also benefit student engagement, as more complex databases are seen as more realistic. In our mixed-methods study, we explore what aspects of an exercise database contribute to student engagement in database education. To gain insight into what students would deem engaging, we asked 56 students to design, implement, and reflect on engaging databases for database education. The results imply that students are engaged by highly diverse yet easily understood database business domains, relatively simple database structures, and conceivable yet seemingly realistic amounts of data. The results challenge some previous study results while supporting approaches found in some textbooks, and provide guidelines and inspiration for educators designing exercise databases for querying and introducing relational database concepts.

CCS CONCEPTS

• **Applied computing** → **Education**; • **Information systems** → **Data structures**.

KEYWORDS

database, education, complexity, SQL, engagement, motivation

ACM Reference Format:
Toni Taipalus, Daphne Miedema, and Efthimia Aivaloglou. 2023. Engaging Databases for Data Systems Education. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2023), July 8–12, 2023, Turku, Finland*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3587102.3588804

1 INTRODUCTION

Database education is an integral part of several computing curricula [13, 30], as a database is one of the most important components of effectively all information systems. Database related knowledge and querying skills are essential for software professionals. Educational research on database related topics has touched subjects such as query languages [28] and normalization theory [21]. In order to teach topics such as querying or normalization in practice, educators typically utilize databases provided by textbooks, or design and implement exercise databases themselves. Practically, this means that the educator must choose a business domain for the exercise database, design and implement a suitable database structure, and populate the database with a suitable amount of data.

Even though educational research has been concerned with the topic of student engagement (i.e., how emotionally committed one is to a task) in general [17] and in specific contexts [23, 25], the intersection of engagement and database education has remained in the sidelines. That is, scientific research has provided little empirical evidence on how to support database education by providing engaging business domains, structural complexities, and data. Even though authors of database textbooks have presented exercise databases based on their professional opinions, it may be difficult for a professional to see a novice point of view when one has already mastered a particular topic.

In this study, we aim to answer the following research questions:

RQ1 What is an engaging database business domain?
RQ2 Which levels of complexity of database structures are engaging?
RQ3 How much exercise data is engaging and why?

In order to answer the research questions, we asked students to design and implement engaging databases for database education. We used the database designs suggested as engaging by 56 students from three cohorts of an advanced database course. This was complemented by qualitative insight, which was provided via written student reflections. While this study answers *what* is an engaging business domain (RQ1) and *which* levels of complexity are engaging (RQ2), another study [19] complements these findings by answering *why* these aspects are engaging.

2 BACKGROUND

2.1 Student Engagement

On a course-independent level, student engagement has been defined from several, evolving perspectives. For example, a summary of engagement viewpoints describes behavioral, psychological, socio-cultural and holistic perspectives, and concludes that all have received criticism [15]. Additionally, cognitive and affective viewpoints, and how they can be measured, have been described with the conclusion that each of these definitions has its context, and that the viewpoint should be carefully selected for each particular case [17]. Despite the multitude of different perspectives, student engagement has been typically measured by self-reporting, which makes the notion of engagement primarily perceptual [23].

ITiCSE 2023, July 8–12, 2023, Turku, Finland

ACM ISBN 979-8-4007-0138-2/23/07.
https://doi.org/10.1145/3587102.3588804

Based on different perspectives, several proposals on how to further student engagement have been made, e.g., promoting students' self-belief and autonomous work, creating challenging educational experiences, and facilitating the development of social and cultural capital [34]. In the more specific context of software engineering education, some studies have touched the notion of engagement. For example, software visualizations have been shown to positively affect student engagement [3], and students have been observed to perceive database related topics more positively when they have control over how querying is visualized [8]. Additionally, the concept of complexity in exercise databases has been shown to positively affect engagement [14, 32, 33]. To our knowledge, there are no scientific efforts to understand engaging database domains.

In this study, our approach to student engagement is affective (cf. e.g. [2]) as we focus on students striving to understand someone else's perspective, i.e., former novices try to understand what is engaging for a novice. It is worth noting, however, that even though the studies on engagement typically focus on how to define, measure and increase engagement, we aim to do none of those in this study. Rather, we strive to understand what is an engaging database for a database novice.

2.2 Exercise Databases

Prior studies have explored what kind of exercise databases are used in higher education to teach database concepts. According to a summary [26] of five database textbooks [7, 9–11, 16], typical exercise databases are concerned with mundane business domains such as order catalogs and company employee records. The complexities of these textbook databases range from relatively simple and common structures of four tables [11] to relatively complex ones with 15 tables [10].

What is the most suitable exercise database for database education remains under debate, and only a few studies have touched the subject [26, 33]. Furthermore, *suitability* itself is a multifaceted concept in this context. For example, it has been proposed that using realistic (i.e., complex) data in exercise databases better prepares students for their future work [14, 32]. It has also been shown that students find more complex database structures more interesting than typical "toy examples" often found in textbooks [33]. On the other hand, students have voiced that more complex data makes querying more difficult [32]. It has also been shown that for databases with higher logical complexity, students make more querying mistakes that they are unable to fix [26]. Additionally, it has been speculated that students are interested in realistic and timely topics in database domains [26], but we are not aware of any scientific evidence on the subject.

2.3 Database Complexity

There are at least three scientific proposals for calculating relational database complexity. First, the multidimensional model complexity metric (MMCM) [22] is concerned with data warehouses, a specific type of database. According to MMCM, the complexity of a database is defined by the number of tables, attributes, and foreign keys in the database. Second, the Database Complexity (DC) metric [24] dictates database complexity by counting the sum of the number of all attributes, primary, secondary and foreign keys, and indices. Third, an unnamed metric [5] uses five numbers to determine database complexity: number of tables (NT), number of attributes (NA), number of foreign keys (NFK), cohesion of the schema (COS), and depth referential tree (DRT). As MCMM is targeted for data warehouses rather than all relational databases, and as DC mixes metrics measuring both logical and physical complexity and merges the results into a single number, we chose the third complexity metric to measure complexity in this study.

The calculation of NT, NA and NFK in the chosen metric [5] are rather self-explanatory. If the relational database is presented as a directed graph where tables represent vertices and foreign keys represent edges, COS is calculated by the sum of the square of the number of vertices in each component of the graph, and DRT is the number of edges in the longest path, not counting loops.

3 RESEARCH METHOD

3.1 Data Collection

We recruited study participants from three cohorts of an advanced-level database course given at the university of the first author. The advanced-level course consists of topics such as database programming and applying NoSQL data models in practice, and it is taken after a basic course in databases, which follows the topics introduced in, e.g., curriculum guidelines for information systems [31]. Out of 68 students, 56 (82%) chose to participate. Most of the participants majored in information systems (52%) and software engineering (36%), other singular students majoring in cyber security, cognitive science, statistics, physics, educational technology, and communication.

As part of a course assignment, we asked students to design and populate a database. The learning objective of the assignment was to build a database from start to finish. The students had to deliver the database schema (ER-diagram or similar), a description of the target area, and the SQL commands to create the tables and data. Then they were provided with three reflection questions, which also inspired our Research Questions:

(1) Why did you choose this target area and why is the target area interesting for a novice?
(2) Why did you choose this number of tables and columns and why is the structure interesting for a novice?
(3) Why did you choose this amount of data and why is this particular amount interesting for a novice?

We requested that the students submit their work to participate in this study, but there were no incentives or deterrents for participating. The students were shown a full privacy statement prior to choosing whether to participate. Participation was based on informed consent.

3.2 Data Analysis

For RQ1, we interpreted the participants' database structures in order to shortly describe the database business domain. We also categorized the business domains into four types, in order to develop understanding on a higher level of abstraction on what types of database domains are engaging.

For RQ2, we analyzed the participants' databases according to the chosen database complexity metrics detailed in Section 2.3, that

Table 1: Database domains and purposes categorized into four themes; note that some domains pertain to more than one category, e.g., there were five databases for digital music platforms, and three of these five databases also contained data structures for social interaction

Database type (# of occurrences)	Database domains (# of occurrences, if more than one)
Support for a physical service (24)	dog contest and health (3); books (2); car sales (2); hotel reservation system (2); university course enrollment (2); bank; board game details and interrelationships; business trip invoicing; car rental service; car wash; employee management; gym memberships; hospital access control; library; multidisciplinary primary school courses; pet health; statistics on board game matches; vaccinations
Delivery of physical or digital goods (21)	online shop (7); digital music platform (5); digital video game distribution platform (4); mobile application store; food delivery platform; marketplace for internet domains; online multiplayer game; operating system update service
Information propagation or collection (14)	statistics on soccer matches (3); dog contest and health (3); academic publications (2); concerts; digital game speedruns; digital mobile gamers; music and movie streaming; pet health; trekking locations
Social interaction (8)	digital video game distribution platform (4); digital music platform (3); car sales

is, for each database, we calculated the number of tables, number of attributes, number of foreign keys, depth referential tree and the cohesion of the database schema.

For RQ3, we report descriptive statistics on the total number of rows and the number of rows per table, based on how much data the participants inserted in their databases. Recognizing that those quantitative results could have been affected by whether the participants used a tool for populating the database, for answering RQ3 we complemented them with qualitative insight on the reasoning the participants provided for the amount of data deemed engaging, which was collected via the participants' reflections. We applied conventional content analysis [12] on the participants answers. First, three authors individually coded participant answers using the same subset of 20% of the data. Next, the authors convened to discuss inter-coder agreements and disagreements by discussing each individual answer. Codes with similar arguments were merged and the explanations of the codes adjusted. Next, two authors split the remaining data and coded the reflections using the new codes discussed in the previous step. Finally, all three authors convened to discuss and reach a consensus on the codings.

4 RESULTS

4.1 Database Domains (RQ1)

The participants chose a wide range of business domains for their databases. These are categorized and detailed in Table 1. Each database could be categorized into more than one type. The most common type (i.e., deemed most engaging) was a database which supports a physical service, e.g., a database of a bank. The second most popular type was a database that enabled the delivery of physical or digital goods, e.g., a database of an online shop. Third came databases which were intended for information collection or propagation, e.g., a database for academic publications. Finally, some databases were concerned with social interaction, e.g., a database for a vehicle marketplace which provided a forum and chat for buyers and sellers to interact with each other.

4.2 Database Complexity (RQ2)

The results for the database previously introduced complexity metrics are detailed in Table 2 and visualized in Fig. 1. Participants deemed an exercise database of six tables with an average of five attributes per table engaging. This is only slightly above the minimum number of tables, for which the whole distribution is available in Fig. 1a. Furthermore, foreign keys were typically only utilized for ensuring the cohesion of the schema, as Fig. 1d shows that hardly any participant used more than one foreign key per table. The remainder of Fig. 1 shows distribution of scores for the other included complexity metrics.

Table 2: Statistics for database complexity metrics

	Mdn	*M*	*SD*
Number of tables	6	7.29	2.952
Number of attributes per table	4.663	4.70	0.964
Number of foreign keys	6	7.39	3.622
Number of foreign keys per table	1	0.99	0.173
Number of rows per table	275	436.81	427.558
Cohesion of the schema	36	61.6	59.564
Depth referential tree	2	2.25	0.977
Depth referential tree per table	0.333	0.33	0.142

4.3 Data Quantity (RQ3)

The participants deemed that an average of several hundred rows per table (*Mdn* = 275, cf. Table 2 and Fig. 1e) was engaging for a database novice. A high standard deviation (*SD* = 427.558), however, indicates that the participants inserted varied amounts of rows for each table.

The analysis of the student reflections revealed diverse reasons for choosing specific amounts of data. Some participants handcrafted their data, while others utilized data generators. The arguments and their frequencies are summarized in Table 3 and discussed in more detail next.

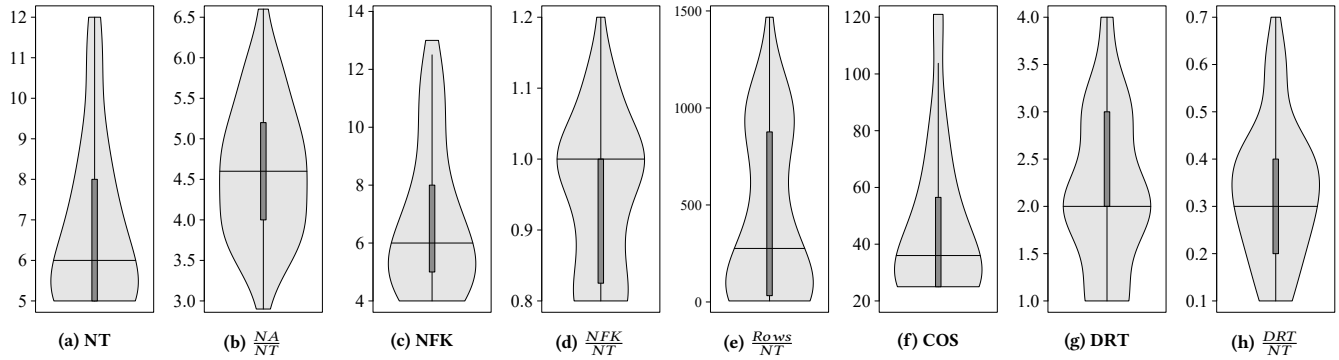

Figure 1: An engaging relational database for database education by several metrics (outliers omitted); (a) number of tables (NT); (b) number of attributes (NA) per table; (c) number of foreign keys (NFK); (d) number of foreign keys per table; (e) number of rows per table; (f) cohesion of the schema (COS); (g) depth referential tree (DRT); (h) depth referential tree per table

Realistic data: The most frequent reason for the chosen amount of data was the relationship between the database and reality. The main theme mentioned by participants was that the data quantities and ratios should match what they expect from the real world, for example *"I tried to keep the ratios of the amount of data in the different tables the same as they would in real life. There are more songs than albums, there are fewer record companies than artists, there are more songwriters than songs, and so on"* (participant P07).

However, realistic data should not necessarily be considered a synonym for large amount of data, as i.a. participant P29 chose speedrunning (i.e., competing how fast one can finish a videogame) as the domain, and wrote *"There are probably quite a small number of registered users, speedrunning is ultimately a pretty niche subculture"*.

Some participants also deemed that, in order to engage novices in writing effective queries, there should be enough data for the differences between well or poorly performing queries to be evident, for example *"Admittedly, significantly more data would have to be generated if, for example, the performance of different query structures on fact sheets with millions or even billions of rows typical of data warehouses"* (P20).

Enough data to practice: Almost as frequently mentioned a reason as realism was the need for enough data to practice database-related topics such as querying. Many participants considered that if correct and incorrect queries return similar result tables, the database does not support engagement. P23, for example, explained that *"The data had to be such that when an SQL query was written with a logical error, the result table looked different than based on the correct SQL query."*

Closely related to the argument of different result tables for different queries, many participants emphasized the importance of creating heterogeneous data, e.g., a dozen customers all from different countries, rather a hundred from the same country. P20 clarified that *"The quality, heterogeneity, and representativeness of the data are more important here than the amount of data."*, while P19 stated that *"I added data to the database in moderation, but still enough so that the contents, functionality and special cases of the database could be demonstrated. [...] only a few rows per table could have given the impression that many special cases could not occur with such a small amount of data."*

Understandable data: Many participants noted that an engaging exercise database should have data which is conceivable for a database novice. Having a smaller dataset allows for easier inspection of the data, on both logical errors and accuracy: *"a small amount of data may make it easier to check the accuracy of the query results"* (P31).

Closely related to the notion that enough data to practice querying is engaging, some participants deemed that smaller datasets make it possible for novices to manually check if their query returns the correct result table, as explained by P52: *"a smaller amount of data makes a more manageable whole, from which logical errors are revealed more easily, because the data can be manually checked"*. P01 also added a small amount of data *"because it keeps the database manageable for learning, and the occurrence of errors is easier to detect. For example, if there were dozens of customers, the result table would be more difficult to read and notice if one person was missing."*

In addition to the three reasons discussed above, nine participants also expressed that their reasoning behind designing an engaging amount of data was based simply on *intuition*. Two of these nine participants were not able to give other arguments for their choices.

5 DISCUSSION

In this study, we aimed to neither define engagement nor inform our participants of what engagement is according to any of the perspectives mentioned in Section 2.1. Rather, we left it up to the participant to speculate and act on what is engaging. Intuitively, if we let a participant design and implement something which they would deem engaging for a novice, the participant instinctively commits to something that is engaging for them as well. If we were instructed to do "something engaging", we would not need to understand theory behind engagement, but only follow our intuition. That is, the results of this paper should not be interpreted as engaging databases based on a selected theory of engagement, but simply as engaging databases because students chose to spend their time on designing and implementing these particular databases.

Table 3: Reasons behind the participants' choices regarding the amount of data for an engaging database

Reason (# of participants)	Arguments
Realistic data (31)	The data should be connected to practice, i.e., a database with a realistic amount of data is engaging. What pertains to realism is dependent on the business domain. Visibility of query performance problems.
Enough to practice (29)	The database should have a sufficient amount of data for querying to be meaningful. If an incorrect query produces the same result table as the correct query, the data does not facilitate learning. Empty result tables are not engaging. Errors in queries should be at least partly indicated by the number of rows in result tables. Some special cases of SQL logic, e.g. using expressions on groups, cannot occur with limited amounts of data.
Understandable data (15)	An engaging database has a conceivable amount of data, so that a query writer can check why a query does not return the correct result table. Data inspection should be relatively easy, as it is not engaging to manually check hundreds of rows if a dozen rows with carefully designed values would achieve the same purpose.

Regarding structural complexity, cohesion of the schema (COS, as reported in Fig. 1f), is not necessarily informative as a violin plot of values from several databases. COS is based on how connected the database tables are, and, based on how COS is calculated, it follows that $NT \leq COS \leq (NT)^2$. All the participants' databases had maximum cohesion (i.e, for all databases, $COS = (NT)^2$), implicating that novices connect all their database tables with each other with foreign keys. That is, there were no participant databases that were comprised of more than one part.

The participants designed databases with longest paths that were not particularly long or particularly short. Fig. 1h and Table 2 show that the median for $\frac{DRT}{NT}$ is approximately 0.33. The authors of the chosen metric state that DRT is not particularly informative by itself [4, 5]. What DRT tells us, however, is that the participants did not favor data structures where the data's lifecycle is linear. That is, there were few designs where the presence of rows in table A is necessitated by linked rows in table B, which is necessitated by linked rows in table C, etc. Additionally, the foreign keys were typically designed in a non-linear fashion. This might be due to several reasons. The most intuitive explanation is that the participants designed the database structures to naturally follow those of the chosen business domain, without focusing primarily on the foreign keys. Another reason might be that long, linear relationships between tables would make data modifications more difficult, which supports the viewpoint of affective engagement (cf. [2]) towards novices.

Rather surprisingly, approximately half of the participants chose general and well-understood business domains for their databases. The others chose specific and modern domains such as digital music platforms or mobile application stores. Despite the popularity of social media services, only a few participants chose to include some form of social interaction in their database. It may be that social media was not considered an engaging topic for novices, or it may be that the participants found these domains difficult to design and develop. These speculated difficulties may lie with database structures regarding relationship-heavy characteristics such as social media followers, friendships, and messaging, which could be considered more difficult to model than fact-heavy characteristics such as customers, products and hotel rooms.

Finally, it is worth noting that engagement is hardly the sole metric to be considered in teaching. If the results from this study suggested using as simple as possible databases to facilitate engagement, it would have been justified to ask if students can learn by using only simple databases. On the other hand, an exercise database designed on an educator's whim might not be engaging and not serve any other desirable goal at the cost of engagement. Perhaps this propounds the view that the results should not be considered to describe an engaging database, but to give scientific evidence towards understanding what can be engaging.

5.1 Practical Implications

As it seems that students prefer simpler databases and easily understood domains, we should use those in teaching, as querying has been shown to be difficult for novices as is [18, 27], without unnecessary complications in database structures, the amount of data, or database domains. On the other hand, there seems to be a school of thought that emphasizes realism and complexity with solid arguments such as preparing students for their future work [32, 33]. These two points of view can also be seen in the arguments presented in Table 3. Some participants stipulated engagement through realistic data, while others considered a conceivable or understandable amount more engaging. With these two (often conflicting) viewpoints in mind, many participants argued for *enough* data for querying to be engaging. According to the arguments presented by the participants, enough data is not simply a matter of quantity, but quality as well. That is, there should be enough data to practice querying with meaningful result tables, and it is crucial that the data is carefully crafted to exhibit heterogeneous values.

On a general level, the database domains in Table 1 may serve as guidelines or inspiration for educators who are choosing or designing an exercise database to be used by novices. The results concerning structural complexity in Fig. 1 can be used in selecting an appropriate logical structure for the database, as well as making an informed decision on how to populate the exercise database. It may also be worth considering the background of students when making the decision of whether to create a realistic or an understandable amount of data, as some studies have recommended moving from unambiguous tasks to more ambiguous ones as the query writer's skill increases [6, 27].

For a more specific practical implication, based on the results yielded by this study, we constructed a database that should be engaging for novices (Fig. 2, structure definitions and data can be

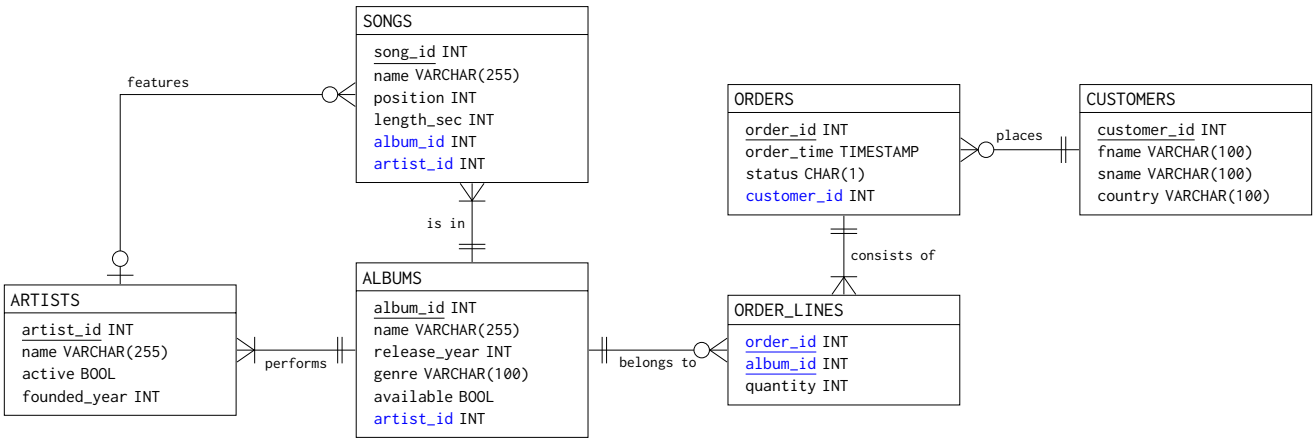

Figure 2: An example of an engaging database based on the results yielded by this study; the domain is relatively easily understood and common, and enables the delivery of digital goods; *NT* = 6, *NA* = 27, *NFK* = 6, *COS* = 36, *DRT* = 2, and with a median of 275 rows per table (not visualized here); foreign keys are indicated in blue

downloaded from GitHub[1]). It is worth noting that this database is based on an abstraction, its structural complexity is based on medians in the quantitative analysis rather than on one selected school of thought, and that the domain is selected from a diverse rather than saturated set of domains suggested by the participants. The database has a COS that is equal to the square of the number of tables, i.e., the database schema is as cohesive as possible. The database contains three longest paths with a DRT of 2, i.e., from *order_line* to *album* to *artist*, from *song* to *album* to *artist*, and from *order_line* to *order* to *customer*.

For the data, we populated the database presented in Fig. 2 with heterogeneous data, so that the novice user may manually check the data to see if their queries contain logical errors. The data were crafted to account for many of the logical query formulation errors described in prior studies [1, 20, 29]. We also designed the database data to contain missing and anomalous values to cater to the demand for realism. This approach has received arguments for [32] and against [29] in prior studies concerned with exercise data.

5.2 Limitations and Threats to Validity

As discussed in Section 2.1, the concept of engagement has been shown to be highly perceptual and subject to interpretation, which presents a threat to how to reliably ask participants to design something based on engagement. With this in mind, we did not ask the participants to *describe* a database that is engaging for a novice, but to build one. Designing and building a database requires considerable effort when compared to merely describing a database. Therefore we reasoned that when a participant commits to a business domain, its corresponding database structure, and data, they are likely to feel engagement towards these choices. Still, it is possible that while the participants were engaged and committed to their database, they might not have made their choices based on what is engaging for a novice, but what is engaging for them, personally. However, the participants of this study had taken merely one course on databases prior to the one the data were collected from. This might mitigate this threat to validity, as the participants were arguably significantly closer to novices than e.g., database textbook authors, educators, or database researchers.

Experiences in prior courses taken by the participants may have influenced or inspired them to these database domains and structures. On the other hand, it may also be that while courses and textbooks give examples on database structures, the participants may have naturally found those examples engaging, and reflected on their positive learning experiences by designing similar databases. Additionally, the assignment requirements (i.e., a minimum of five tables were required) might have influenced the results relating to the number of tables. Regarding the qualitative analysis of participant answers on why a certain amount of data is engaging, 19 participants failed to answer, stating being affected by either assignment restrictions or personal time limitations. This may indicate that the datasets reported in this study are smaller than the participants actually deemed engaging. Despite the fact that inter-cohort differences were mitigated by collecting the data from three student cohorts, and that there were several majors represented by the participants, the data were collected from one university.

6 CONCLUSION

The question of how to choose an exercise database business domain, structural complexity, and the amount of data has been resting on the (nonetheless educated) intuition of educators and textbook authors rather than scientific evidence. In this study, we approached the topic of engaging databases for database education by asking novices to design and implement engaging databases. The results show that to be engaging for a novice, the database should *(i)* have a mundane and common business domain such as an online shop or digital music platform, *(ii)* be relatively simple in terms of structure, and *(iii)* contain a realistic amount of data in order for a novice to practice querying and get meaningful results, yet the amount of data should be conceivable and the values heterogeneous. These results may be utilized by educators and textbook authors in designing exercise databases that cater for novice engagement.

[1]https://github.com/tonitaip-2020/engaging_database

REFERENCES

[1] Alireza Ahadi, Julia Prior, Vahid Behbood, and Raymond Lister. 2016. Students' Semantic Mistakes in Writing Seven Different Types of SQL Queries. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE)*. ACM Press, New York, New York, USA, 272–277. https://doi.org/10.1145/2899415.2899464

[2] Jeanne M Butler. 2011. Using standardized tests to assess institution-wide student engagement. *Promoting Student Engagement* 1 (2011), 258–264.

[3] Michael D Byrne, Richard Catrambone, and John T Stasko. 1999. Evaluating animations as student aids in learning computer algorithms. *Computers & Education* 33, 4 (1999), 253–278.

[4] Coral Calero, Mario Piattini, and Marcela Genero. 2001. Empirical validation of referential integrity metrics. *Information and Software Technology* 43, 15 (2001), 949–957. https://doi.org/doi.org/10.1016/S0950-5849(01)00202-6

[5] Coral Calero, Mario Piattini, and Marcela Genero. 2001. Metrics for controlling database complexity. In *Developing Quality Complex Database Systems: Practices, Techniques and Technologies*. IGI Global, 48–68. https://doi.org/10.4018/9781878289889.ch003

[6] Gretchen Irwin Casterella and Leo Vijayasarathy. 2013. An Experimental Investigation of Complexity in Database Query Formulation Tasks. *Journal of Information Systems Education* 24, 3 (2013), 211–221. http://jise.org/Volume24/24-3/pdf/Vol24-3pg211.pdf

[7] Thomas Connolly and Carolyn Begg. 2015. *Database Systems (6th. ed.)*. Pearson.

[8] Suzanne W. Dietrich, Don Goelman, Jennifer Broatch, Sharon Crook, Becky Ball, Kimberly Kobojek, and Jennifer Ortiz. 2021. Introducing Databases in Context Through Customizable Visualizations. *Frontiers in Education* 6 (2021). https://doi.org/10.3389/feduc.2021.719134

[9] Ramez Elmasri and Shamkant B. Navathe. 2016. *Fundamentals of Database Systems (7th. ed.)*. Pearson.

[10] Jeffrey A Hoffer, Venkataraman Ramesh, and Heikki Topi. 2011. *Modern database management*. Upper Saddle River, NJ: Prentice Hall.

[11] Jeffrey A Hoffer, Heikki Topi, and Venkataraman Ramesh. 2014. *Essentials of Database Management*. Pearson Education.

[12] Hsiu-Fang Hsieh and Sarah E Shannon. 2005. Three Approaches to Qualitative Content Analysis. *Qualitative Health Research* 15, 9 (2005), 1277–1288. https://doi.org/10.1177/1049732305276687

[13] Joint Task Force on Computing Curricula, Association for Computing Machinery (ACM) and IEEE Computer Society. 2013. *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. Technical Report. New York, NY, USA. https://doi.org/10.1145/2534860 999133.

[14] Nenad Jukic and Paul Gray. 2008. Using Real Data to Invigorate Student Learning. *SIGCSE Bulletin* 40, 2 (2008), 6–10. https://doi.org/10.1145/1383602.1383604

[15] Ella R Kahu. 2013. Framing student engagement in higher education. *Studies in Higher Education* 38, 5 (2013), 758–773.

[16] David Kroenke and David J. Auer. 2016. *Database Processing: Fundamentals, Design, and Implementation (14th. ed.)*. Pearson Education.

[17] B Jean Mandernach. 2015. Assessment of student engagement in higher education: A synthesis of literature and assessment tools. *International Journal of Learning, Teaching and Educational Research* 12, 2 (2015).

[18] Daphne Miedema, George Fletcher, and Efthimia Aivaloglou. 2022. Expert Perspectives on Student Errors in SQL. *ACM Transactions on Computing Education* (2022). https://doi.org/10.1145/3551392

[19] Daphne Miedema, Toni Taipalus, and Efthimia Aivaloglou. 2023. Students' Perceptions on Engaging Database Domains and Structures. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (Toronto ON, Canada) *(SIGCSE 2023)*. Association for Computing Machinery, New York, NY, USA, 122–128. https://doi.org/10.1145/3545945.3569727

[20] Andrew Migler and Alex Dekhtyar. 2020. Mapping the SQL Learning Process in Introductory Database Courses. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education* (Portland, OR, USA) *(SIGCSE '20)*. Association for Computing Machinery, New York, NY, USA, 619–625. https://doi.org/10.1145/3328778.3366869

[21] A. Mitrovic. 2002. NORMIT: a Web-enabled tutor for database normalization. In *International Conference on Computers in Education*. 1276–1280. https://doi.org/10.1109/CIE.2002.1186210

[22] Sushama Nagpal, Anjana Gosain, and Sangeeta Sabharwal. 2012. Complexity metric for multidimensional models for data warehouse. In *Proceedings of the CUBE International Information Technology Conference*. 360–365.

[23] Larian M Nkomo, Ben K Daniel, and Russell J Butson. 2021. Synthesis of student engagement with digital technologies: a systematic review of the literature. *International Journal of Educational Technology in Higher Education* 18, 1 (2021), 1–26.

[24] Mile Pavlic, Marin Kaluza, and Neven Vrcek. 2008. Database complexity measuring method. In *Central European Conference on Information and Intelligent Systems*. Faculty of Organization and Informatics Varazdin.

[25] Arnold N. Pears. 2010. Enhancing student engagement in an introductory programming course. In *2010 IEEE Frontiers in Education Conference (FIE)*. F1E–1–F1E–2. https://doi.org/10.1109/FIE.2010.5673334

[26] Toni Taipalus. 2020. The effects of database complexity on SQL query formulation. *Journal of Systems and Software* 165 (2020), 110576. https://doi.org/10.1016/j.jss.2020.110576

[27] Toni Taipalus. 2020. Explaining Causes Behind SQL Query Formulation Errors. In *2020 IEEE Frontiers in Education Conference (FIE)*. 1–9. https://doi.org/10.1109/FIE44824.2020.9274114

[28] Toni Taipalus and Ville Seppänen. 2020. SQL Education: A Systematic Mapping Study and Future Research Agenda. *ACM Transactions on Computing Education* 20, 3, Article 20 (2020), 33 pages. https://doi.org/10.1145/3398377

[29] Toni Taipalus, Mikko Siponen, and Tero Vartiainen. 2018. Errors and Complications in SQL Query Formulation. *ACM Transactions on Computing Education* 18, 3, Article 15 (August 2018), 29 pages. https://doi.org/10.1145/3231712

[30] The Joint Task Force on Computing Curricula. 2015. *Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering*. Technical Report. New York, NY, USA. https://dl.acm.org/citation.cfm?id=2965631

[31] Heikki Topi, Kate M. Kaiser, Janice C. Sipior, Joseph S. Valacich, J. F. Nunamaker, Jr., G. J. de Vreede, and Ryan Wright. 2010. *Curriculum Guidelines for Undergraduate Degree Programs in Information Systems*. Technical Report. New York, NY, USA. https://dl.acm.org/citation.cfm?id=2593310

[32] Paul J. Wagner, Elizabeth Shoop, and John V. Carlis. 2003. Using Scientific Data to Teach a Database Systems Course. In *Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education* (Reno, Navada, USA). ACM, New York, NY, USA, 224–228. https://doi.org/10.1145/611892.611975

[33] Kwok-Bun Yue. 2013. Using a Semi-Realistic Database to Support a Database Course. *Journal of Information Systems Education* 24, 4 (2013), 327–336. http://jise.org/Volume24/n4/JISEv24n4p327.pdf

[34] Nick Zepke and Linda Leach. 2010. Improving student engagement: Ten proposals for action. *Active Learning in Higher Education* 11, 3 (2010), 167–177.