Journal of
Information
Systems
Education

Volume 31
Issue 4
Fall 2020

Improved Teaching of Database Schema Modeling by Visualizing Changes in Levels of Abstraction

Adi Katz

Recommended Citation: Katz, A. (2020). Improved Teaching of Database Schema Modeling by Visualizing Changes in Levels of Abstraction. *Journal of Information Systems Education,* 31(4), 294-311.

Article Link: http://jise.org/Volume31/n4/JISEv31n4p294.html

Initial Submission: 25 October 2019
Accepted: 24 March 2020
Abstract Posted Online: 8 September 2020
Published: 10 December 2020

ISSN: 2574-3872 (Online) 1055-3096 (Print)

Improved Teaching of Database Schema Modeling by Visualizing Changes in Levels of Abstraction

Adi Katz
Shamoon College of Engineering (SCE)
Ashdod, Israel
adis@sce.ac.il

ABSTRACT

Conceptual modeling of databases is a complex cognitive activity, particularly for novice database designers. The current research empirically tests a new pedagogy for this activity. It examines an instructional approach that stresses visualizing gradual transitions between levels of abstraction in different hierarchic levels of a relational database schema. The new approach builds on a four-level TSSL model from the field of human-computer interaction. TSSL, an acronym for the Task, Semantics, Syntax, and Lexical levels, is applied here to describe the levels of conceptual database modeling and to explain how improved instructional design can help minimize extraneous cognitive load during the design of database schemas. We tested the effectiveness of the proposed instructional approach via a controlled experiment carried out on IS students. We divided students into two groups, those exposed to a visual emphasis on the syntax of gradual transitions in a schema structure and those not exposed to it. We then measured performance in terms of errors in students' solutions while also recording their perceptions and attitudes toward the instructional approach and the activity of database modeling. Our results show that the new approach is an effective tool for teaching database modeling.

Keywords: Data modeling, Database design & development, Pedagogy, Visualization

1. INTRODUCTION

Information systems are built to support operations, management, and decision-making in organizations. Therefore, they need to meet the demands of the intended users efficiently and effectively. Proper modeling of user requirements is a key task of information system analysis and design meant to ensure that the information system solution meets organizational goals (Dahan, Shoval, and Sturm, 2014). Since databases are at the heart of information systems, teaching database modeling is an important part of educational programs in Information Systems (IS) and related fields such as Software Engineering. This research presents and evaluates an approach for effectively delivering the activity of relational database schema modeling to novice database designers.

Database designers are required to gather information about organizations and to capture the information in a conceptual model. Designers then build up the conceptual model with definitions of relations, attributes, and relationships among relations with integrity constraints (Elmasri and Navathe, 2011). The activity of modeling usually involves translating an oral or a textual description of various aspects of an organization into a concise and proper representation of relations, attributes, and relationships. The description is usually long and full of details, and user requirements may be complex. This makes the activity of modeling a difficult cognitive process for novice database designers, and one that is error prone. Students in database courses have demonstrable difficulties in thinking like data modelers (Watson, 2006). Data modeling is problematic for novices because of the abstract and complex nature of the database analysis and design process (Connolly and Begg, 2006). The challenges of effectively delivering the topic have been addressed in the existing database modeling pedagogy. Interesting attempts include the constructivist approach (Connolly and Begg, 2006), the integrated-spiral approach (Watson, 2006), the cognitive apprenticeship approach (Al-Dmour, 2010), and the learning-from-errors approach (Katz and Shmallo, 2016).

In this paper, we demonstrate how an approach of visually emphasizing the syntax of a hierarchical structure of database schemas containing gradual transitions between levels of abstraction is useful for teaching relational database schema modeling. We apply TSSL (Foley et al., 1990), a model from the human-computer interaction (HCI) literature, to database modeling education as a theoretical framework to explain and simplify the interaction between novice database designers and organizational scenarios. TSSL (Task, Semantics, Syntax, and Lexical levels) was originally used to describe the four levels of interaction with a system. Our instructional approach exploits the idea that there are multiple levels of human interaction with any artifact. While HCI professionals analyze the multi-level interaction in the context of human-computer interfaces, in database design the interaction is with organizational descriptions of requirements and constraints. Effective database modeling involves transitions between different levels of abstraction (Srinivasan and Te'eni, 1995). Since TSSL levels differ in their level of abstraction, the model is a suitable framework to describe the multi-level activity of database modeling and to explain how improved instructional design can help minimize the extraneous cognitive load of novice database designers. The approach we present can serve as an additional tool to reduce the conceptual complexity that characterizes database modeling.

Students in a typical database course learn many activities. Our focus is on the conceptual level, during which a relational schema is defined that expresses the database requirements of a specific organization, institution, or any other entity using an information system. At this stage a database designer needs to accurately analyze the organization's needs and constraints (business rules) and, on that basis, form a database schema which includes relations, attributes, relationships, and interrelation-referential-integrity constraints (Elmasri and Navathe, 2011). Relational database modeling is a central topic of database courses, and delivering it effectively is therefore essential.

Many of the challenges that educators face in teaching the topic of database modeling have been described in previous studies (Watson, 2006; Katz, 2018a). Transforming an organizational description into a proper database schema of relations and relationships is a mental process characterized by a high level of cognitive complexity. Cognitive Load theory provides a framework for designing instructional materials that focuses on identifying instructional designs that can effectively reduce the unnecessary cognitive burdens on the learner (van Merriënboer and Sweller, 2005). The negative effects of extraneous information processing on learning have been demonstrated in various research studies (Kalyuga and Sweller, 2014; Mayer and Fiorella, 2014). Educators can apply the theory to minimize the extraneous cognitive load that interferes with the process of learning. Since extraneous cognitive load is imposed by the ways information is presented, educators can reduce it by choosing effective instructional techniques. For example, an interesting approach when teaching database concepts to undergraduates is to use a goal-based scenario with a method called worked-out examples (Bunch, 2009). How complex or simple things seem depends critically upon the way in which they are presented. To achieve simplification, educators must find the right representation. In line with cognitive load theory, it has been found that visualization supports learning by decreasing cognitive load. Comprehension is enhanced by reducing the extraneous cognitive processing (Schwamborn et al., 2011). Learning activities that include visual representations of data can reduce complexity, enhance students' comprehension of abstract ideas, and also form mental representations of complex analytical concepts (Saundage et al., 2016). In addition, hierarchy is one of the most effective ways of organizing complexity for human comprehension (Flood and Carson, 1993). The fact that many complex systems have a nearly decomposable, hierarchic structure is a major facilitating factor enabling human beings to understand and to describe such systems (Simon, 1962). In line with these findings, the current approach emphasizes visualizing hierarchic levels in a database schema. Visualizing hierarchical data structures is a wide research field developed over the last three decades, significant in many economic and scientific applications (Müller et al., 2017).

As previously stated, TSSL is used in the current study as a lens for understanding the interaction between learners and organizational descriptions of requirements and constraints. This interaction is central to the conceptual activity of database schema modeling. Specifically, TSSL serves as a theoretical framework for explaining how visualizing the syntax of a hierarchical schema structure characterized by two simultaneous gradual transitions has a high potential for reducing the extraneous cognitive load in database modeling (Katz, 2018a). In the following section, the four levels of the TSSL model will be exemplified in the context of an academic institution modeling scenario.

2. APPROACH

2.1 Applying TSSL in Database Schema Modeling

The term 'levels of abstraction' refers to multi-level structures, which describe a particular issue or activity at different levels. Each level is described at a different degree of abstraction-concreteness or detail-generality (Te'eni and Sani-Kuperberg, 2004). People frequently use levels of abstraction when they solve problems or do other activities involving human information processing. One such activity is modeling databases on the basis of a textual description of an organizational scenario. In the area of HCI, there is a dichotomous distinction between semantic and syntactic levels of interactions to explain programmers' behaviors. Semantic knowledge consists of general and meaningful sets of information that are independent of the syntactic knowledge of particular programming languages or facilities (Shneiderman and Mayer, 1979). The four-level TSSL model elaborates this dichotomous division so that the two high levels of TSSL specify the semantic level and the two low levels specify the syntactic level. In the context of teaching relational database schema design, while the two upper levels of TSSL, task and semantics, are close to the organizations' realities, the two lower levels of TSSL, syntax and lexical, can be affected by educational practitioners.

In TSSL, each level provides the context for the level below it (Te'eni, Carey, and Zhang, 2005). However, since educational practitioners have control over the two lower levels of TSSL (Katz, 2018a, 2018b), it is these that we seek to affect. Figure 1 depicts the four-level TSSL model, with each level explained in the context of database schema modeling. The specific visualization choices made by educators at the lower and physical levels (lexical and syntactic) can promote a higher level of comprehension (semantic) and more accurate design of a schema for organizational users (task). Such a bottom-up influence can be achieved when the syntactic level integrates the building blocks of the lexical level. A deeper understanding of the entities and the relationships between entities will then be attained at the semantic level. Comprehension of the entities and relationships is essential for the database designer to properly express the organizational requirements at the highest task level. Our approach guides educators to use visualization aids that involve relative locations, sizes, and colors of objects in the database schema. These allow students to form gradual visual distinctions between the physical characteristics of the database elements that should respectively express the gradual semantic distinctions between different levels of abstraction in a database schema.

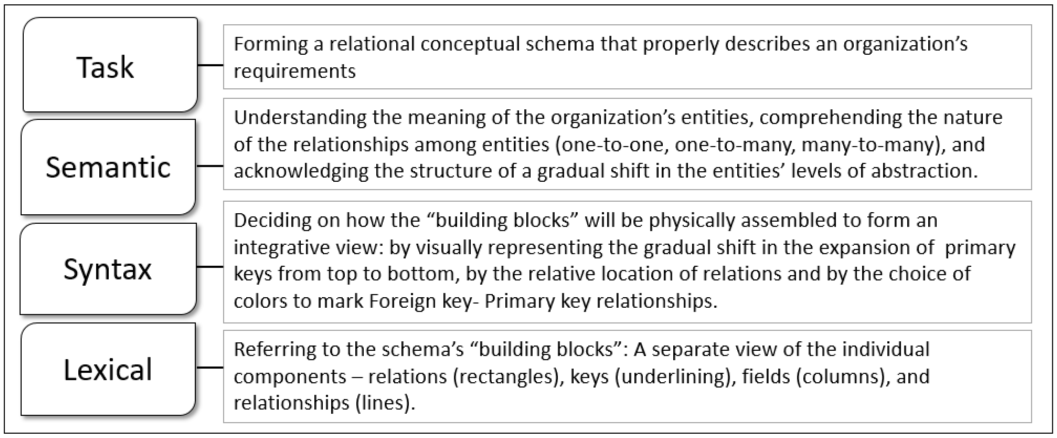

Figure 1. The Four-Level TSSL Model Explained in the Context of Database Schema Modeling

2.1.1 Task. The uppermost task level is about organizational requirements and constraints that have to be accommodated in a relational conceptual schema. The task of database designers is to define a database schema that properly and accurately expresses an organization's description of its needs. Each schema includes relations with attributes (fields), primary keys and foreign keys (hereafter PKs and FKs), and connections (relationships) between relations. Figure 2 displays a database schema for the academic scenario that appears in Section 2.2. The modeling task is considered successful if the relations, keys, and connections included in Figure 2 appropriately meet all the organizational needs, activities, and constraints described in the textual scenario. For example, the sentence "a student can have several different phone numbers" leads to the creation of the relation 'Student Phones.'

2.1.2 Semantic. The semantic level supports the task level. At this level, database designers are required to understand the meaning of entities and the meaning of the relationships between entities. The designers must define the attribute or combination of attributes that unambiguously identify an instance of an entity: the primary key (PK). Sometimes there are different options for attributes that can unambiguously identify an instance of an entity, and the designer needs to decide which option is most preferable.

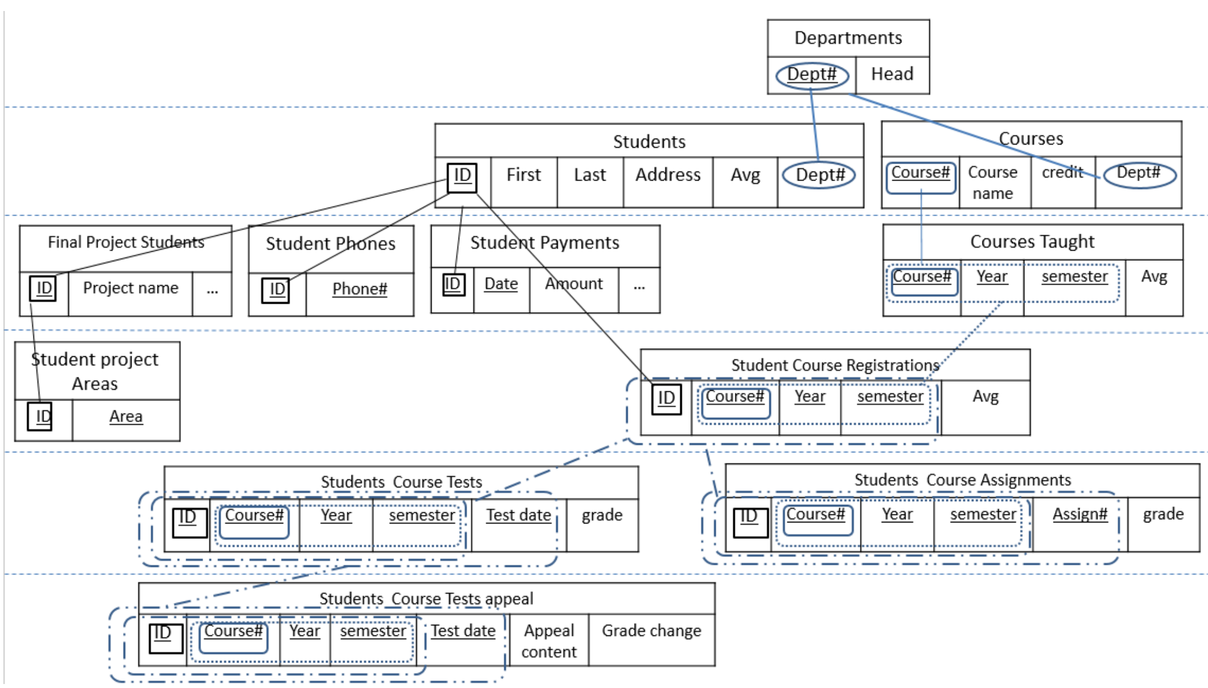

Figure 2. A Database Schema for an Academic Scenario
Note. The different types of dashed lines that surround the keys denote the different key colors originally used

It is essential for the database designer to properly grasp the meaning of the different types of entities and the relative meaning and relationships between one type and another in a specific organization. As mentioned, each "level of abstraction" is described at a different degree of abstraction-concreteness or detail-generality (Te'eni and Sani-Kuperberg, 2005). Previously, we used the term "levels of abstraction" to describe the behavior of moving from one TSSL level to another in database modeling. In addition to moving between the four TSSL levels of abstraction, there is an additional distinction between the different levels of abstraction within the semantic level. This is due to the hierarchical nature of a database schema, in which different hierarchical levels are characterized by different degrees of abstraction-concreteness or detail-generality. In other words, at the semantic level, there is a gradual change in the degree of the entities' abstraction throughout the organizational hierarchical schema. The top level starts with entities (things in the organization's reality) that are quite clear, tangible, and straightforward. Gradually going from top to bottom, entities relatively become either more abstract (e.g., events, processes), specific, or more detailed. There are quite tangible entities (departments, students, and courses in the academic scenario in Figure 2), and there are more abstract entities that represent events or processes that occur in relation to the tangible entities (e.g., students' payments and students' registrations to courses). Entities usually become more complex when moving in the hierarchy from top to bottom. We will demonstrate this again in more detail in Section 2.2.

Another crucial aspect of database modeling at the semantic level is to identify the nature of the relationships among entities derived from the organizational description in terms of cardinality ratios – one-to-one, one-to-many, and many-to-many (1:1, 1:N, M:N, respectively). For example, a final project student is a sub-type of the student entity, someone who performs a project in the final year of her or his studies. A final project student has the same attributes of a regular student (e.g., name and address), but has additional attributes that are related to the final project. Therefore, this is a 1:1 relationship and both relations have identical keys. We elaborate on and illustrate this issue later in Section 2.2.

2.1.3 Syntax. As said above, while the upper levels of TSSL, task and semantics, are close to the organizations' realities, the lower levels of TSSL, syntax and lexical, can be affected by the pedagogic implementation of educational practitioners. Educators can use different syntaxes to display database schemas. Since extraneous cognitive load is imposed by the way information is presented, educational designers should explore different presentation methods to reduce the excessive cognitive load sensed by their students. It follows that the syntax level is extremely important for coping with the cognitive load that arises from an organizational description. Since visualization supports learning by reducing extraneous cognitive processing (Schwamborn et al, 2011), a proper visual representation of information at this level can promote comprehension at the next (higher) semantic level.

Hierarchic organization of information has been found to be an effective way to reduce complexity and ease human comprehension. The hierarchic nature of many complex systems enables the comprehension of them (Simon, 1962; Flood and Carson, 1993). Hierarchical structuring has been a central tool for abstraction because it removes the complexity of large schemas (Gandhi, Robertson and Van Gucht, 1994). Research on memory indicates that a hierarchical organization of materials serves as a retrieval cue for recall since a general–specific structure helps us to locate particular items (Najarian, 1981). In line with the findings that point to the importance of visualization and to the advantages of hierarchal structures, the current approach emphasizes *visualizing the hierarchic levels* in a database schema and the *gradual shifts* between these levels.

In order to promote the semantic understanding of relations and relationships between relations of any organization, our approach focuses on a structured diagram with a syntax that visually highlights the hierarchical nature of an organization's schema. The hierarchical structure is characterized by the gradual transition of entities' level of abstraction, as previously mentioned with reference to the semantic level. But, in addition, when moving down the hierarchy from top to bottom, a parallel gradual transition occurs: the relations' PK gradually expand as they include more attributes from one level to the other. There is a repetition of the PKs of their parent relations with an additional attribute (or attributes). Relations at the next bottom level can have PKs that are formed by a combination of the PKs of their mutual parents. This PK expansion is quite simple to visualize. For example, in Figure 2, the PKs gradually expand when moving down from 'Courses' through 'Courses Taught' and 'Student Course Registration' until reaching 'Students Course Assignments.'

In TSSL, when moving in a bottom-up direction, the syntax level integrates the "building blocks" (components) of the lowest lexical level. The viewer is aware of the relative distinctions and similarities among the building blocks. Distinctions and similarities are stressed in the schema by the choice of location, size, color, and other visual attributes. The individual visual choices made by educators for representing each schema component at the lexical level are examined together at the syntax level. While at the lexical level the emphasis is on the separate parts, at the syntactic level the viewer can identify a pattern in the picture as a whole. This is in line with the idea of synergism: a database schema is viewed as a whole, not as a loose collection of parts. The well-known rules of Gestalt theory, also known as the *Laws of Simplicity,* play a significant role in information visualization. Gestalt psychologists were the first to study the perceptual organization principles involved in grouping (Wertheimer, 1912a, 1912b; Arnheim, 1949; Ehrenstein, 2008). They dealt with the question of how individual elements group into parts that in turn group into larger wholes separated from other wholes. They describe how to arrange visual symbols in a graphical display optimized to achieve a better, more effective visualization. The Gestalt principles pertaining to *grouping* affirm that humans perceive objects as organized patterns and objects. According to Gestalt psychologists, the human mind has an innate disposition to perceive patterns in the stimulus based on certain rules (Wagemans et al., 2012). The Gestalt principles of *proximity* and *similarity* can be utilized by educators as visualization aids for highlighting the gradual transitions between hierarchical levels of a database schema. According to the Gestalt grouping principle of *similarity*, when all else is equal, the most similar elements (in color, brightness, size, texture, shapes, etc.) are seen as grouped together. We apply the similarity principle by using colors to mark the PK attributes of the relations. Repeating the same colors to indicate common attributes in FKs and PKs makes the gradual expansion of the PKs visually prominent. The use of different colors is an effective way to easily distinguish between hierarchy levels (Karstens, Kreuseler, and Schumann, 2003) and between different levels of abstraction (Te'eni and Sani-Kuperberg, 2005). According to the *proximity* principle, objects that are near or proximate to each other tend to be seen as grouped together. We apply the proximity principle by horizontally dividing the modeling solution area (whiteboard or sheet of paper) into sections. Each section represents a different hierarchical level of the schema, and the relations in the sections are arranged so that parent relations are always located closely and above their child/children relations.

Pinna (2010) listed the main phenomenal rules governing the formation of shape and meaning and suggested a link between perceptual grouping, shape perception, and visual meaning. The term "perceptual meaning" refers to what is expressed, indicated, or conveyed by a grouping and a shape through its "amodal wholeness" and "modal partialness." Among seven properties related to the complexity manifested in perceptual meaning, three properties are relevant to the visualization aids of similarity and proximity as we use them to emphasize the gradual transitions between the levels of abstraction in different hierarchic levels of a relational database schema. The properties are *emergence*, *hierarchical organization*, and *variability*. Emergence is related to the fact that the perceived meanings are not present in any of the individual subcomponents taken alone, but emerge from component interactions. While individual subcomponents belong to the lexical level of TSSL, the relative variances and similarities in the PKs of relations and the relative location of subcomponents in the schema belong to the syntactic level. At the syntactic level, the viewer compares the visual elements and perceives a pattern with a certain gradual change. Hierarchical organization and centralized control means that the complex form of meaning manifests a hierarchy in which power is spread over a decentralized structure that involves all components. A number of units combine to generate a system of meanings so that the meaning of one component depends on the meanings of other components. Emphasizing the hierarchical organization of database schemas by dividing the modeling area into separated horizontal sections makes the hierarchical structure of parent-and-child relations clearly evident. This structure allows the educator to follow the three pattern types of FK-PK relationships that are described below in section 2.2. Variability refers to the fact that very tiny variations in the transitions can induce a huge variation in their meanings. As we show, the visual differences between one hierarchical level to the other are gradual. A gradual variation of colors is used as an external marker of the semantic difference in the meaning of the entities, from concrete to abstract and from general to more specific and detailed.

In addition to the gradual variations in color and locations, we recommend that educators choose relation labels in a way that will highlight the gradual semantic change in the level of abstraction. The labels in a database schema should gradually expand from top to bottom, in accordance with the shift from general entities to ones that are more specific, detailed, and concrete. This will be demonstrated in Section 2.2.

All types of relationship cardinality ratios, one-to-one (1:1), one-to-many (1:N), and many-to-many (M:N), can be represented by this hierarchical structure with two parallel gradual transitions: the syntax of the gradual PK expansion and the gradual semantic change in entities' level of abstraction. A FK of a child relation is a PK of a parent relation used to reference the parent. In section 2.2, three exhaustive patterns of Foreign key – Primary key (FK-PK) relationships will be presented for demonstrating the gradual transitions.

2.1.4 Lexical. In database modeling, this level refers to the visual appearance of each separate component in the schema's diagram. For example, it is common to display a relation as a table with columns, to specify that an attribute is a PK by underlining its text, and to draw lines that connect relations to express relationships between them (all of these visual elements are seen in Figure 2). This lowest TSSL level holds the building blocks of the schema's visual representation. While at this level the emphasis is on the separate appearance of the schema's components, the syntax is about similarities and differences between these components, as explained above.

2.1.5 Semantic and syntactic parallel transitions in database modeling. Effective data modeling involves transitions between different levels of abstraction. Schema modeling is a process in which top-down and bottom-up are interchangeably-used techniques (Srinivasan and Te'eni, 1995). The fact that the semantic and syntactic gradual transitions occur in parallel supports and enables the integration of both techniques in database schema modeling. On the one hand, the semantic level serves as a conceptual guide for the visual choices made at the syntactic level, such as deciding on the relative location of relations and properly defining the PKs. On the other hand, as mentioned, the syntactic level of visual representations increases the comprehensibility of the hierarchic schema structure and the semantic meaning of the entities. In Section 2.1, we maintained that educators have control over the two lower levels of TSSL (lexical and syntax). Accordingly, we emphasize the bottom-up influence in which certain visualization choices made by educators at those levels promote the higher levels of comprehending (semantic) and fulfill the goal of designing an accurate organizational schema (task). This bottom-up technique is in line with the idea that perception is not just about groups and shapes, but also about meanings. What we perceive always has a meaning, and vision is also about perceiving meanings (Pinna, 2010). We apply TSSL by visualizing the gradual transitions in levels of abstraction between hierarchical levels of database schemas. During class meetings, the instructor constantly emphasizes how gradual transitions of visual components at the syntax level represent the gradual transitions in the abstraction of entities at the semantic level. Tying these two gradual transitions together and showing their simultaneous occurrence is a significant part of our new approach. In the following section, we demonstrate how to visually and conceptually emphasize the two gradual transitions in an academic scenario.

2.2 Demonstrating Gradual Transitions in Database Schema Modeling – An Academic Scenario

We now illustrate the emphasis on the syntax of the hierarchical structure of a database schema with gradual transitions by means of an academic scenario.

An academic institution must maintain data on departments, students, courses, and student participation in courses. Each student belongs to one department. Each course has a unique code, name, and credits. Each course may be taught in different academic periods, but in a given academic period it will be taught only once. When a course is taught in a particular academic period, there is an average grade for all the students who are enrolled in it.

Students pay tuition. A student can pay different fees on different dates. The academic institution maintains the following payment information (assuming one payment per student at each date): credit card company, credit card number, validity, CVV, the amount paid, and the number of payments. A student can have several different phone numbers in the database.

Only students in the last year of their studies perform a final project. Each project has a name and may have several keywords that describe the areas related to the project's topic. It is important to keep additional data about the project, such as the advisor, separate scores of the advisor and the judging committee, and the submission date.

A student may enroll in a course in more than one academic period (to improve his or her achievement in the course). Every time a student enrolls in a course, he has a final grade in the course. Students submit assignments in courses and take exams in courses. A course can have several assignments, and a student in a course receives a separate grade for each assignment. Each course has several exam dates, and each student enrolled in a course can be tested in more than one exam in that course. For each exam, he receives a score.

A student enrolled in a course can submit an appeal for a grade that he or she received on an exam in that course. A student has the option to submit one appeal per exam (but can submit an appeal for each exam in the course). The system keeps the content of the appeal, the lecturer's (or TA's) textual reply to his appeal, and how many points were added or subtracted from the score (if any) by the course lecturer following the appeal.

Figure 2 displays a database schema for this academic scenario. Owing to space limitations and since we mainly focus on defining relations, relations' PKs, and FK-PK relationships, not all attributes mentioned in the textual description appear in the figure.

Three exhaustive pattern types of FK-PK relationships are reflected in the academic schema:

1. The FK is a regular field in the child relation, expressing a 1:N, parent-child relationship. This type appears twice in Figure 2: The relationship between 'Students' and 'Departments' and the relationship between 'Courses' and 'Departments.' Each student is considered a child of a certain department (belongs or is registered to one specific department) and the same can be said about each course (each course is offered by a specific department).
2. The FK is part of the PK in the child relation, expressing a pattern of an M:N relationship. Symmetrically, each entity can be treated as both the parent and the child of the other. In relational databases, M:N is implemented by means of a cross-reference (also called a junction) relation in a way that forms a pair of 1:N relationships. Treating an M:N relationship as two symmetric hierarchic relationships by considering two parents of a joint child (the cross-reference relation) simplifies a relatively complex relationship. This type is represented in Figure 2 as the M:N relationship between 'Students' and 'Courses Taught' and their joint child relation 'Student Course Registrations.' The joint child's PK is an integration of his parents' PKs. A Student can be treated as the parent of all the courses he is enrolled in, and a course can be treated as the parent of all the students who enrolled in it. It often happens that when M:N relationships are derived from an organizational description, only one of two parents is shown in the schema. Several such cases appear in the academic scenario. For example, 'Courses Taught' has only one parent ('Courses') in the schema. The other parent, 'Academic Period' (composed by the combination of a year and a semester), does not appear as a relation (having no additional attributes in the scenario), but appears in 'Courses Taught' via attributes needed for identifying occurrences of a course. One can mistakenly perceive this as a 1:N relationship, but educators should emphasize that it is a partial presentation of the second FK-PK pattern, since there is a M:N relationship between courses and academic periods.
3. The FK in a child relation is identical to the PK, expressing a 1:1 relationship. This pattern usually expresses an 'IS A' relationship, known as a generalization-specialization pattern. The child relation is a specific sub-type of its parent (paralleling the inheritance concept in the object-oriented approach). This is the only case in which the PK of a child does not expand the PK of its parent. This type appears twice in Figure 2: as the relationship between 'Final Project Students' and 'Students' and as the relationship between 'Students Course Tests Appeal' and 'Students Course Tests.' Referring to 'Final Project Students' and 'Students' having additional unique attributes (advisor, scores of the advisor and the judging committee, submission date, etc.), a Final Project Student IS-A specific sub-type of the general student type. Since only students in their last year of studies perform a final project, specifying the unique attributes that accompany students that conduct a final project in the above 'Students' student relation would create multiple blank fields in the database and therefore inefficiently waste storage space. Referring to 'Students Course Tests Appeal' and 'Students Course Tests,' an appeal is an event that may or may not occur after a student's test was checked. When it occurs, it adds specific new detail to the student's test.

Let us address what we asserted above about relationship pattern types 2-3. Following studies stressing that hierarchy is one of the most effective ways of organizing complexity (Simon, 1962; Flood and Carson, 1993), we recommended a simplified view of these relationships by treating them as hierarchical although they are not. Of course, students need to first comprehend the three distinct types of relationships that exist between entities. Once they have, they can be offered a simplified hierarchical view of many-to-many (two symmetric sets of hierarchical relationships) and of one-to-one (hierarchy of a type and sub-types) relationships. Importantly, in a database course, the distinction between the hierarchical data model and the relational data model must be taught from the onset.

In Figure 2, we see that parent relations are placed directly above their child relations. In keeping with the Gestalt principle of Proximity, this relative positioning of relations is a choice that we recommend educators make at the syntax level. The current approach positions parent relations above and closest to their child relations. This helps visualize the hierarchic database structure and prevents the creation of incorrect FK-PK relationships. For clarity, direct parents and children should usually be placed at a minimum distance from each other (one hierarchy level), but this is not always possible. Sometimes the parents of a joint child are not at the same hierarchical level, and then the distance between the 'higher' parent and the child is the distance from the higher' parent to the 'lower' parent plus one more level. This can be seen in the academic scenario in the 'Student Course Registrations' relation.

A gradual semantic transition between levels of abstraction exists in the academic scenario. From top to bottom, entities gradually transition from being simple and tangible to being either more abstract or more detailed and specific (concrete): 'Department,' 'Students,' and 'Courses' are simple and easy to comprehend. However, it should be noted that 'Students' and 'Courses' located below the 'Department' level serve to expand the information about each department ('Students' lists the students enrolled in the department and 'Courses' lists all the courses offered by each department). At the next hierarchical level, there are relatively abstract entities that represent events (such as 'Student Payments' and 'Courses Taught'), or more specific types of entities (such as 'Final Project Students' which is a student type with specific attributes such as a project's name), or more detailed entities (such as 'Student Phones' that adds details to 'Students' and 'Student Course Assignments' that adds more detail to 'Student Course Registrations'). Our recommendation is to start by tracking down the simplest entities that are conceptually easiest to define and afterwards handle the more abstract entities that are related to the simple ones.

Referring to the gradual expansion of the PKs: when looking at the visual presentation, it can be seen that when moving top-down, the relations' PK gradually comes to include more attributes. Following the hierarchical syntax of this gradual transition can reduce the definition of erroneous referential integrity constraints. Ignoring the gradual expansion of PKs might lead to adding invalid 'grandfather-grandchild' relations along with proper 'father-child' relations (Katz, 2018a). The consequence is redundancy since there is already an implicit FK relationship from a child to his grandfather via the father. At the semantic level, students also need to understand that the consequence of redundancy is forcing unnecessarily checks of compliance with the defined referential integrity constraints on the system.

We recommend that educators refer to relations in a way that will highlight the gradual semantic change in the level of abstraction. The written labels can be relatively short (for time and space saving considerations), but as much as possible, the uttered references to relations should highlight the gradual extension from top to bottom in accordance with the shift from general to more specific, detailed, and concrete entities. For example, from top to bottom: Department >> departments' courses >> departments' courses taught >> students' registration to departments' courses taught >> students' tests in departments' courses taught that they had registered to >> students' appeals in tests in departments' courses taught that they had registered to.

In the section that follows, we describe an empirical experiment conducted to test whether the current approach is effective in educating modeling of relational databases.

3. METHOD

This study adopts an "educational action research" methodology in which the motivation for being involved in an educational activity is the improvement of the teaching and learning quality. Educational action research aims at the development of autonomous improvement ability for educators using systematic self-observations and testing pedagogic ideas using research procedures in class (Fessakis, Dimitracopoulou, and Komis, 2005). We tested the current pedagogic approach in a "Database" course in the form of a controlled experiment to investigate whether our approach significantly improves database modeling. The course's curriculum includes a series of activities developed to learn and practice the topic of database modeling. We manipulated one variable, the instructional approach, and observed the impact on performance in a database modeling task and on subjective perceptions of the students. Accordingly, the experimental study is designed as one independent factor between groups with two treatments. This kind of assignment of participants to treatment groups is often used in experimental evaluation of modeling techniques (Dahan, Shoval, and Sturn, 2014).

Following previous studies that showed how visual representations and hierarchic organizations can improve instructional design, we emphasize the bottom-up direction of TSSL to show that instructors can help minimize extraneous cognitive load in learning database schema modeling. We previously described the bottom-up influence as follows: the way in which the visual building blocks (lexical) are integrated and assembled (syntax) can promote comprehension of the entities and relationships (semantic) to eventually achieve an appropriate database schema for a certain organizational context (task). Consequently, we manipulated the instructional approach by teaching the subjects with and without an emphasis on visualizing the gradual changes in the levels of abstraction of database schemas. In other words, the controlled experiment was conducted to compare a group of students who were exposed to a learning process that emphasizes the syntax of gradual transitions between hierarchical levels (an experimental group) to a group of students who learned schema modeling without this emphasis (a control group). We tested the differences between the groups at the two upper levels of TSSL:

at the semantic level, we compared the comprehension of various schema components (subjective perceptions); and at the task level, we compared the appropriateness of the modeling solution (performance in terms of errors).

Below we describe the experimental task, the experimental design, the procedure, and the measures. Importantly, the IRB (ethics committee) of the academic institute in which the experiment was conducted approved the research project, including the experimental task, the testing procedure, and the collection of data.

3.1 The Experimental Task

The task was to conceptually model a database according to an organization scenario. Participants had to create a database schema from a textual description of an online flower shop. The scenario was identical for both experimental groups. The scenario was given to each participant on a printed sheet of paper. All participants created the schema in class at the same time and were asked to work independently. The subjects were told that they could not ask the instructor any questions during the experiment. In addition, they were informed that they are expected to do their best to model according to their understanding of the scenario and to base their solution on what they had learned in preceding class sessions. It was emphasized to them that the solution is anonymous – that is, the person who will check the solution will not know the identity of the solver and, therefore, the quality of their solution will not affect their course grade or any other judgement. Participants were promised that at the end of the experiment they would receive the task's solution and a detailed explanation in the following class meeting.

3.2 Experimental Design, Procedure, and Measures

3.2.1 Sample and procedure. We conducted the experiment twice, a year apart in two separate but identical runs. All subjects were third-year Information Systems (IS) track students in the department of Industrial Engineering and Management at an academic college of engineering. All participants were enrolled in a "Database" course and volunteered to participate at their instructor's request. One run of this experiment took place in May 2018 and the other in May 2019. The double run was due to the limited number of students in a single course group. It is important to mention that both courses were taught by the same lecturer and same teaching assistants. For the 2018 experiment, 32 students volunteered, and for the 2019 experiment, 23 students volunteered, reaching an overall number of 55 subjects. In each run, we randomly divided the subjects to one of the two treatment groups. There were 30 men (17 in 2018 and 13 in 2019) and 25 women (15 in 2018 and 10 in 2019). There were 28 subjects (16 in 2018 and 12 in 2019) in the experimental group and 27 subjects (16 in 2018 and 11 in 2019) in the control group. The groups were labeled as group 1 (control) and 2 (experimental). To avoid bias, participants did not receive information about the experimental difference between the groups and only knew if they belonged to group 1 or 2. Also, being aware that a lecturer's level of excitement about a new approach may introduce unwanted noise that can create differences between groups, a conscious effort was made to not exhibit emotional-affect differences or any other differences that might affect the results other than the manipulated independent variable. For the first phase, the groups separately learned database modeling (in separate class meetings). For the second phase, they all simultaneously solved an identical modeling exercise in a classroom under the instructor's supervision.

3.2.2 Experimental Design, Manipulations, and Dependent Variables. We manipulated the instructional approach and observed the impact on performance in a database-modeling task and on subjective perceptions of the way the topic was delivered in class. During the learning phase, each group separately learned and practiced identical textual scenarios that describe organization requirements in class meetings with the same instructor. An example of a textual scenario that was used is the academic scenario presented above in Figure 2. For the experimental group, subjects were taught how to transform textual scenarios to database schemas with an emphasis on visualizing the gradual changes in the levels of abstraction of database schemas. The emphasis was achieved at the syntax level by a strict horizontal division of the solution area into rows. Each row represented a different hierarchical level of the schema. The location of the relations in the rows was such that parent relations were always located closely above their child/children relations as previously explained and demonstrated (see Section 2.2). In addition, the same colors were used for identical attributes in the PKs of the relations to visualize the expansion of PKs from top to bottom down the hierarchy. The particular choice of colors is not important: what matters is consistency so that the same key field appears everywhere in the graphical display in the same color. In the control group, however, the location of the relations was quite random without horizontal separation of the whiteboard into hierarchical levels and without using colors to highlight the shared key fields of the various relations. Figure 3 shows photos of the database schema solution for the academic scenario on the class whiteboard. The upper part (A) shows the solution generated for the control group, and the lower part (B) shows the solution generated for the experimental group. It can be seen that the same building blocks appear in both photos (lexical level), but the differences and similarities in terms of PK colors and the relative location of relations in the working area present a different visualization (syntax level).

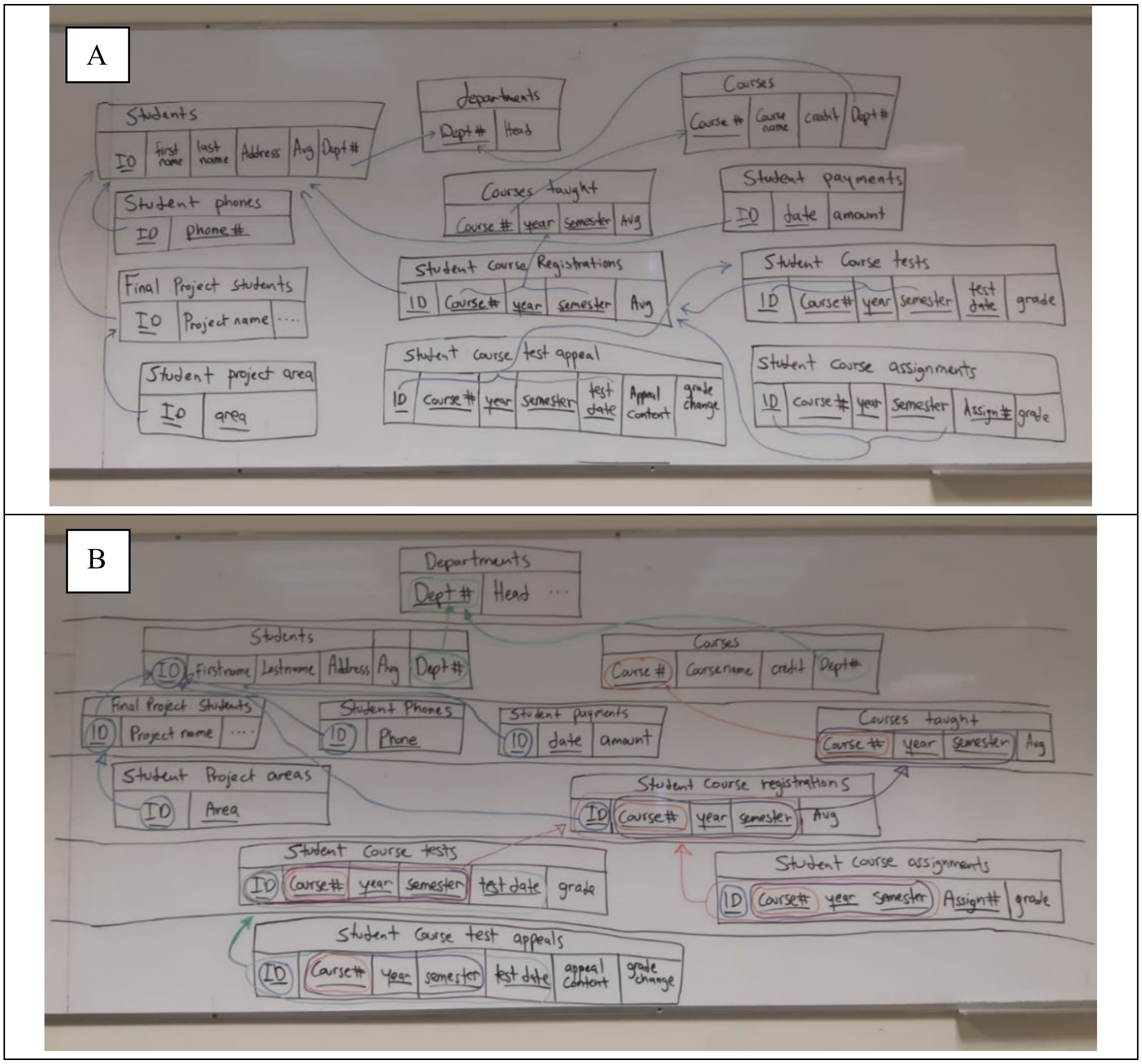

Figure 3. Photos of the Academic Scenario Solutions Generated During the Learning Phase on a Whiteboard for each Group

Once the learning phase was completed, a testing phase took place during the next class meeting in which both groups' participants were given an identical, unseen textual scenario of an online flower shop. As part of the test, student participants were asked to create a database schema. The subjects worked in the classroom under the instructor's supervision. The instructor first explained the task, stressing that the goal was to successfully model the database schema to best correspond to the textual description. More specifically, the students were instructed that the database schema solution should resemble the solutions presented in the previous learning sessions; in other words, that they should list relations with all their fields, PKs, and FK-PK relationships. In line with the TSSL bottom-up direction, we examined the influence of our new instructional approach by testing the differences between the groups at the two upper levels of TSSL. For the semantic level, we measured the perceived comprehensibility of various schema components by means of a survey. For the task level, we analyzed the subjects' database modeling performance in terms of the quality of their solutions, evaluated by the number and types of errors found. The error analysis follows pre-defined categorizations of errors retrieved from a previous pedagogic study in the area of database modeling (Katz and Shmallo, 2015). We summarize the types of errors measuring performance in Table 1.

Analyzing the data collected from the solutions will show whether participants in the experimental group are more accurate in defining the PKs of relations, make fewer errors in terms of needless additions (such as adding redundant relations or redundant relationships between relations), and in terms of omitting required elements (such as missing relations or failing to define crucial relationships between relations).

Error type	Description
Improper PK definition – Redundant/Missing	Creating a super key or a defining a key with an attribute not mentioned in the scenario or excluding necessary fields from the PK
Improper FK definition – Redundant/Missing	Including needless foreign key fields or excluding necessary fields from the FK
Redundant relations	Creating a relation by an unnecessary fragmentation of fields
Redundant fields	Including a field appearing as an attribute in the scenario in an unsuitable relation
Redundant FK-PK connections	Splitting a unified FK into several FKs, making connections between indirect relations (such as grandson-grandfather)
Missing relations	Excluding a relation by uniting its fields in another relation
Missing Fields	Excluding a field appearing as an attribute in the scenario from its suitable relation
Missing FK-PK connections	Not making connections between direct hierarchical relations (son-father)

Table 1. Modeling Performance Measured by Types of Modeling Errors

In both groups, the task was limited to 45 minutes. After 45 minutes, students were asked to hand in their solutions. Upon submission, each student received a short printed survey and was asked to complete it. The survey examined the participants' attitudes towards the pedagogic approach she or he was exposed to, their satisfaction with the process of learning database modeling, and the level of their perceived comprehensibility of aspects of database schema modeling. As explained before, comprehensibility is related to the semantic level in TSSL. The survey was anonymous, meaning that subjects were instructed not to write their names. Instead, they were asked to specify their group (1 or 2). The survey measured student perceptions of nine items appearing in Table 2, analyzed by the six-point Likert-type scale. At the end of the survey sheet, there were empty lines allowing participants to add comments if they so wanted. The text above the lines encouraged the additional input with the prompt: "It would be helpful if you add feedback here by writing additional comments that are general or specific extensions to specific survey items."

3.3 Hypotheses

The comparison between the treatment groups in terms of performance and subjective attitudes will allow us to reach a conclusion regarding the effectiveness of the current approach in educating database schema modeling. The expectations are that we will find higher-quality solutions and more positive attitudes toward learning data modeling when using the pedagogical approach of visually emphasizing the hierarchical nature of schemas that have gradual transitions between hierarchical levels. Table 3 specifies nine hypotheses regarding the students' modeling performance in terms of modeling errors that indicate the quality of their solutions. Each modeling error type that appears in Table 1 has a corresponding hypothesis in Table 3 (H1-H8). An additional hypothesis refers to the overall number of errors (H9). General formulations of the null hypothesis and the alternative hypothesis appear at the top of Table 3. The specific hypotheses are formulated by replacing 'errors' with a specific error type that appears below in rows 1-9. Table 4 specifies the hypotheses regarding students' perceptions and attitudes towards learning data modeling. Each survey item that appears in Table 2 has a corresponding hypothesis in Table 4. In Table 3, the formulation of hypotheses 11-18 are achieved by replacing the blank lines appearing in the second row with the text that appears in rows numbered 11-18, respectively.

Perception Item	Statement
Difficulty – Modeling	To what extent was it difficult for you to model (create a relational DB schema from a textual description)?
Interest – Modeling	To what extent was it interesting for you to model (create a relational DB schema from a textual description)?
Comprehensibility – Modeling	To what extent do you feel that you have understood the modeling process (creating a relational DB schema from a textual description)?
Clarity – Relations	To what extent do you feel that it is clear – what are the necessary relations?
Clarity – PKs	To what extent do you feel that it is clear – what will be the primary keys of the relations?
Clarity – Fields	To what extent do you feel that it is clear – which fields (attributes) are required for each relation?
Clarity – Connections	To what extent do you feel that it is clear – which connections are needed between the relations?
Structure – Learning Approach	To what extent do you feel that the instructional approach of database modeling was structured correctly (organized, systematic, etc.)?
Clarity – Learning Approach	To what extent do you feel that the instructional approach of database modeling was clear to you?

Table 2. Students' Perceptions Measured by Six-Point Likert-Type Statements

H_i	H_1: Group 2 students will have significantly less errors in comparison to Group 1 students;: $\mu2 < \mu1$ H_0: Group 2 students will not have significantly less errors in comparison to Group 1 students; $\mu2 \geq \mu1$
1	errors of improperly defining PKs
2	errors of improperly defining FKs
3	errors of defining redundant relations
4	errors of defining redundant fields
5	errors of defining redundant FK-PK connections
6	errors of omitting (missing) relations
7	errors of omitting (missing) fields
8	errors of omitting (missing) FK-PK connections
9	overall number of errors

Table 3. Hypotheses Regarding Students' Performance in Terms of Different Types of Modeling Errors

H_i	
10	H_1: Group 2 students will have significantly lower mean scores on the perception of modeling difficulty as compared to Group 1 students; $\mu2 < \mu1$ H_0: Group 2 students will not have significantly lower mean scores on the perception of modeling difficulty as compared to Group 1 students; $\mu2 \geq \mu1$
	H_1: Group 2 students will have significantly higher mean scores on the ___________ as compared to Group 1 students; $\mu2 > \mu1$ H_0: Group 2 students will not have significantly higher mean scores on the ___________ as compared to Group 1 students; $\mu2 \leq \mu1$
11	degree of interest in database modeling
12	degree of comprehension of database modeling
13	clarity of defining relations
14	clarity of defining relation PKs
15	clarity of defining fields (attributes)
16	clarity of defining connections between relations
17	correctness of the instructional approach's structure
18	clarity of the instructional approach

Table 4. Hypotheses Regarding Students' Perceptions and Attitudes

4. RESULTS

4.1 Descriptive Statistics

Table 5 presents descriptive statistics for students' modeling performance measures (quality of the solutions) in terms of modeling error types. Table 6 presents descriptive statistics of students' perceptions and attitudes towards learning data modeling collected from the survey.

Error type	Min	Max	Mean	Std. Dev.	Var.
Improper PK definition – Redundant/Missing	0	5	1.96	1.347	1.813
Improper FK definition – Redundant/Missing	0	9	1.33	1.656	2.743
Redundant relations	0	2	0.33	0.546	0.298
Redundant fields	0	7	3.42	1.474	2.174
Redundant FK-PK connections	0	4	1.56	1.273	1.621
Missing relations	0	4	2.29	1.133	1.284
Missing Fields	0	4	1.44	1.273	1.621
Missing FK-PK connections	0	4	1.47	1.052	1.106
Total errors	5	27	14.73	5.201	27.054

Table 5. Descriptive Statistics – Students' Performance in Terms of Errors, N = 55

Survey Item	Min	Max	Mean	Std. Dev	Var.
Difficulty – fModeling	1	6	3.80	1.061	1.126
Interest – Modeling	1	5	3.58	0.956	0.914
Comprehensibility – Modeling	2	6	3.60	1.116	1.244
Clarity – Relations	3	6	4.00	0.667	0.444
Clarity – PKs	3	6	4.07	0.790	0.624
Clarity – Fields	3	6	4.38	0.680	0.463
Clarity – Connections	2	5	3.47	0.959	0.921
Structure - Learning Approach	3	6	4.38	0.871	0.759
Clarity – Learning Approach	2	6	4.09	1.005	1.010

Table 6. Descriptive Statistics – Students' Perceptions and Attitudes, N = 55

4.2 Hypotheses – The Effects of the Instructional Approach on Modeling Performance and on Perceptions and Attitudes towards Modeling

Participants in groups 1 (control) and 2 (experimental) were taught database modeling without and with an emphasis on visualizing the gradual changes in levels of abstraction between hierarchical levels of database schemas. We tested the differences between the groups with respect to students' perceptions and attitudes via a Mann-Whitney U-test. This non-parametric method is appropriate for the perceptions and attitudes variables that we measured by Likert scales which are ordinal-level (Kuzon, Urbanchek, and McCabe, 1996; Jamieson, 2004) and for relatively small sample sizes (Norman, 2010).

Although all performance measures (quality of the solutions) in terms of modeling error types are interval-level data, for some performance measures we compared the difference between groups using independent-samples t-tests, and for others we used the Mann-Whitney U-test. It is well established that the t-test has a power advantage for normal distributions and is robust to modest deviations from the test assumptions (de Winter and Dodou, 2010). Accordingly, we conducted normal distribution analysis in SPSS using normality plots (Q-Q), histograms, boxplots, and the Shapiro-Wilk test. We found that only some performance measures were normally distributed: improperly defining PKs, redundant fields, missing relations, missing FK-PK connections, and total number of errors. For those five performance measures alone, we used independent-samples t-tests; for the rest, we ran Mann-Whitney U-tests.

Table 7 presents the results of the t-tests and descriptive statistics of students' performance measures, and Table 8 presents the results of the Mann-Whitney U-tests of students' performance measures and perception measures, both split by instructional approach.

As already noted, the experiment was run twice, a year apart in two separate but identical experiments. We manipulated only the instructional approach and therefore had to ensure that there was no effect stemming from the date the course the participants were enrolled in (whether they took the course in 2018 or 2019). Using t-tests and Mann-Whitney U-tests, we found that there was no significant difference in any of the performance (task level) and perceptions (semantic level) measures between the 2018 and the 2019 groups. Table 9 presents the results of t-test and descriptive statistics of students' performance measures, and Table 10 presents the results of the Mann-Whitney U-tests of students' performance measures and perception measures, both split by experimental run.

Outcome			Group				
			Control (1) N = 27		Experiment (2) N = 28		df = 53
			M	**SD**	**M**	**SD**	**t**
Students' performance in terms of errors	H1	PK definition	2.59	1.248	1.36	1.162	3.801**
	H4	Redundant fields	3.67	1.664	3.18	1.249	1.233
	H6	Missing relations	2.44	1.188	2.14	1.079	0.986
	H8	Missing FK-PK connections	2.07	0.997	0.89	0.737	5.008**
	H9	Total number of errors	17.96	4.719	11.61	3.478	5.700**

** $p < 0.01$

Table 7. Results of T-Tests and Descriptive Statistics Split by Instructional Approach

		Outcome	Group: Control (1) N = 27	Group: Experiment (2) N = 28		
			Mean Rank	Mean Rank	U	P
Students' performance in terms of errors	H2	FK definition	36.07	20.21	160	0.000
	H3	Redundant relations	30.40	25.68	313	0.179
	H5	Redundant FK-PK connections	38.43	17.95	96.5	0.000
	H7	Missing Fields	28.72	27.30	358.5	0.743
Students' perceptions and attitudes	H10	Difficulty – Modeling	36.50	19.80	148.5	0.000
	H11	Interest – Modeling	25.57	30.34	312.5	0.248
	H12	Comprehensibility – Modeling	19.02	36.66	135.5	0.000
	H13	Clarity – Relations	22.59	33.21	232	0.004
	H14	Clarity – PKs	21.02	34.73	189.5	0.001
	H15	Clarity – Fields	25.74	30.18	317	0.254
	H16	Clarity - Connections	18.35	37.30	117.5	0.000
	H17	Structure – Learning Approach	17.98	37.66	107.5	0.000
	H18	Clarity – Learning Approach	18.65	37.02	107.5	0.000

Table 8. Results of Mann-Whitney U-Tests Split by Instructional Approach

		Outcome	Group: 2018 N = 32		Group: 2019 N = 23		df = 53
			M	SD	M	SD	t
Students' performance in terms of errors	H1	PK definition	1.91	1.353	2.04	1.364	-0.370
	H4	Redundant fields	3.56	1.564	3.22	1.347	0.854
	H6	Missing relations	2.44	1.105	2.09	1.164	1.135
	H8	Missing FK-PK connections	1.53	0.983	1.39	1.158	0.483
	H9	Total number of errors	15.31	5.076	13.91	5.376	0.984

** $p < .01$

Table 9. Results of T-Tests and Descriptive Statistics Split by Experimental Run

		Outcome	Group			
			2018	2019		
			N = 32	N = 23		
			Mean Rank	**Mean Rank**	**U**	**P**
Students' performance in terms of errors	H2	FK definition	26.17	30.54	309.5	0.305
	H3	Redundant relations	26.05	30.72	305.5	0.211
	H5	Redundant FK-PK connections	29.50	25.91	320	0.409
	H7	Missing Fields	23.75	33.91	232	0.170
Students' perceptions and attitudes	H10	Difficulty – Modeling	27.45	28.76	350	0.765
	H11	Interest – Modeling	27.97	28.04	367	0.995
	H12	Comprehensibility – Modeling	26.11	30.63	307.5	0.291
	H13	Clarity – Relations	27.64	28.50	356	0.832
	H14	Clarity – PKs	26.78	29.70	392	0.503
	H15	Clarity – Fields	25.92	30.89	301.5	0.221
	H16	Clarity - Connections	25.44	31.57	289	0.143
	H17	Structure – Learning Approach	27.88	28.17	364	0.953
	H18	Clarity – Learning Approach	29.20	26.33	329.5	0.953

Table 10. Results of Mann-Whitney U-Tests Split by Experimental Run

As mentioned above, at the end of the survey, participants were encouraged to offer general comments or specific extensions to specific survey items. Table 11 presents selected quotes from participants at each group:

Group 1	The lesson was monotonously delivered.
	The lesson method was good, but its implementation is very difficult and modeling is not simple at all.
	I just could not understand the topic of connecting relations.
	The subject is difficult and broad, there must be a way to simplify it.
	The method of modeling was conveyed in a rather clear manner, but there is a need to sharpen the topic of relationships between relations.
	The main difficulty is the connection between the relations.
	More emphasis should be placed on connections and on the determination of keys.
Group 2	The modeling process was clear and the learning method was excellent.
	The division of the white board into several hierarchical levels helps.
	The method was organized and clear. I lacked more practice for solving the scenario independently, but in the end the method is good and effective, only requires refinement.
	Although there is still ambiguity on the subject, the teaching method was good and understandable.
	I liked the illustrations (of keys) in colors, it made a visual order.
	It was neat to see how the keys expand from top to bottom. It helped me understand the more complicated relations.
	The method was organized, so it was easy to approach the exercise. It's always easier when there is order.

Table 11. Quotes of Participants' Comments Collected from the Survey

The results indicate significant differences between the experimental groups, with a distinct advantage in most of the performance measures and in subjective perceptions in the group that studied modeling using the new method. This means that one group was given a better experimental treatment than the other. In order to repair this lack of equivalence between students belonging to the two different groups, in the class meetings that followed the experiment, student participants who were assigned to the control group learned the topic of modeling in the new method that emphasizes visualizing the gradual changes in the levels of abstraction of database schemas.

5. DISCUSSION

The purpose of this research was to investigate the effectiveness of a new pedagogy to deliver the topic of database modeling. It tested an instructional approach that stresses visualizing the gradual transitions between levels of abstraction in different hierarchic levels of a relational database schema.

In general, the results support the effectiveness of our new pedagogy in terms of performance in modeling tasks. In terms of TSSL, we found that students who learned database modeling with the visual emphasis showed modeling solutions of a higher quality at the task level. Overall, they made fewer errors (H9) than students that learned without the visual emphasis. Specifically, they made fewer errors of the following types: improperly defining PKs (H1), improperly defining FKs (H2), defining redundant FK-PK (H5), and missing proper FK-PK connections (H8). The last two types are erroneous referential integrity constraints that include making invalid 'grandparent-grandchild' (indirect) relations and omitting proper 'parent-child' (direct) relations.

In the experimental group, in which we exposed students to our new pedagogic approach, we marked PKs and FKs with colors. We used identical colors for marking common fields (of PKs and FKs) of different relations. This way, we implemented the Gestalt Similarity principle, creating a part-whole grouping that puts together elements (individual attributes/fields) perceived as parts of a whole grouping (FKs and PKs). The colors visually highlighted the general pattern of the expansion of PKs from top to bottom. In addition, the close placement of child relations under their direct parent relation also contributed to these students' ability to see the pattern of a gradual expansion of the PKs and the joining of new fields (and colors) to the PKs from top to bottom. The use of colors to mark PKs and FKs and the adherence to hierarchical locations of child and parent relations linked by FK-PK connection lines are syntax level choices that produced a better comprehension of the entities and their relationships at the semantic level. We increased the awareness of differences and similarities in keys between hierarchy levels by using different and similar colors. In other words, the gradual differences in colors from top to bottom serves as a clear external marker of the gradual semantic transitions in the level of abstraction of entities, changing from concrete to abstract or from general to more specific and detailed.

We now would like to address the effectiveness of the new pedagogy as reflected in students' perceptions and attitudes. In general, the results show that most student perceptions regarding the modeling task and attitudes to the instructional method favored the new approach. In terms of TSSL, at the semantic level we found that students who learned database modeling with the new approach perceived the modeling process as less difficult (H10) and more comprehensible (H12). The process of defining relations (H13), PKs (H14), and connections between relations (H16) was clearer to those who were exposed to the visual emphasis of gradual changes in the levels of abstraction of database schemas. In addition, students exposed to the new instructional approach sensed it as more properly structured (H17) and clearer (H18) in comparison to the control group students. These findings are supported by the TSSL framework and Gestalt principles. The TSSL framework lets us explain how educators can strive for a bottom-up influence by using visualization aids involving relative locations and colors of objects in the database schema. Gradual visual distinctions between the physical characteristics of the database elements should respectively express the gradual semantic distinctions between different levels of abstraction in a database schema. A deeper understanding of the entities and relationships between entities will then be attained at the semantic level. Comprehension of the entities and their relationships is a prerequisite for properly expressing the organizational requirements at the highest task level. As described, at the syntax level, our approach uses visualization aids that apply the Gestalt principles of Similarity and Proximity to emphasize patterns of gradual transitions in a database schema. The Simplicity principle means that a good Gestalt organization is a simple organization (Stickel, Ebner, and Holzinger, 2010). Visual complexity can be considered as extraneous load (Schmutz et al., 2009), influencing the cognitive load and mental effort of the viewer (Holzinger, Kickmeier-Rust, and Albert, 2008; Harper, Michailidou, and Stevens, 2009) and his or her emotions (Tsai et al., 2008).

Applying the new approach in database courses offers students a strategy for better coping with complex textual descriptions of organizational requirements and constraints. TSSL explains how database schema modeling is an activity that involves physical actions at the concrete levels of visual representations (lexical and syntax) and also involves more abstract higher level cognitive processes that are related to comprehension and decision-making (semantic and task). In our approach, the physical-visual syntax of gradual transitions between the different hierarchy levels is tightly tied to the conceptual semantic gradual transitions between entity abstraction levels. Repeated practice of emphasizing these parallel syntactic-semantic (physical-conceptual, respectively) transitions throughout the database schema structure will accustom students to apply it naturally when encountering new scenarios. Acknowledging the "behavior" of gradual changes will guide novice database designers to look for them in organizational descriptions. Our results predict that they will succeed in the crucial task of creating accurate and proper models.

5.1 Limitations and Future Directions

Our experiment was of short duration and carried out in the context of an entire course in which many topics are related. Database courses are characterized by a high level of element interactivity since the different topics are understood and learned with reference to other topics and cannot be considered independently (Katz and Shmallo, 2016). Therefore, future research is needed to re-examine the current findings in a database course in which visualization aids will be used throughout the whole course in all points of reference to the contexts for conceptual modeling.

We believe that lessons learned from database conceptual modeling can be usefully projected onto other topics in IS education. There are many other areas where visualization aids can effectively support the learning process. For example, a series of previous studies tested a visualization tool tailored to assist learning and understanding a range of programming concepts. Results showed that program visualization helps novice programmers to develop appropriate mental models (Ma et al., 2009).

5.2 Implications

Since the current pedagogic approach has been found useful, it could guide the development of computer-based environments for learning and exercising database schema modeling which would be used in database courses. Such a tool may contain a pool of pre-written textual scenarios and a working area divided horizontally into different hierarchical levels. Students would be able to add or remove hierarchical levels with dedicated hierarchical icons. Visualization techniques such as colors and shapes would automatically mark identified gradual transitions between hierarchical levels of the schema referring to mutual fields (attributes) of different relations as we demonstrated manually. The tool should visualize errors in cases of violation of the gradual changes in the expanding PKs, explain the semantic implications of the violation, and propose a solution for the correction. A detailed description of the design of such a tool is beyond the scope of this paper.

In Section 2.1, when referring to the semantic level of TSSL, we explained that conceptual database modeling involves different levels of abstraction. According to Te'eni (2018), at any moment, problem solvers focus on a particular level of abstraction to achieve an immediate goal, while under certain conditions they move back and forth to other levels until the overall goal is achieved. This movement is an adaptive behavior, necessary for a high level of performance in problem-solving tasks. Designers of computer-based learning environments should think of ways to support this adaptive behavior. Te'eni demonstrates this idea in the context of database modeling and shows design features that support adaptive behavior for the task of building an entity-relationship diagram (ERD) of a system. In the context of designing a computer-based environment for teaching database modeling, a future direction is that the system would dynamically suggest to novice modelers to move to the appropriate level of abstraction. This computer-based support can be explored in the context of moving between the four levels of TSSL, as mentioned before in Section 2.1, in reference to the semantic and syntactic parallel transitions in database modeling. In addition, there are transitions within the semantic level of TSSL between entities with different levels of abstractions (see the semantic level part in Section 2.1). Another movement between levels of abstraction that may be explored and supported is from the lowest level of defining the attributes of a specific relation, through the level of defining a particular relationship between relations, to the highest level of deciding on how the modeling is progressing toward the goal of a complete a correct diagram.

Another future research direction is to compare alternative visualization aids at the lexical and syntax levels to find the most efficient and useful ways of emphasizing the gradual changes between different hierarchy levels of a database schema.

6. CONCLUSIONS

Modeling a database schema to reflect organizational scenarios is an essential skill, but it is a complex cognitive process for the novice database designer. Finding an appropriate pedagogy to teach database modeling to novice database designers is a challenge for IS educators. The objective of this study was to empirically test whether a particular pedagogic approach for educating database design is effective for delivering the cognitively complex material of schema modeling. This approach emphasizes visualizing the gradual transitions between hierarchic levels in a database schema. We empirically tested the pedagogic approach in a controlled experiment. Results show that it is effective for teaching relational database modeling. Visualizing the gradual transitions between the levels of abstraction in different hierarchic levels of a relational database schema is effective for reaching solutions that are more precise. We measured the quality of the solutions (performance) in terms of the type and amount of modeling errors. Visualizing the gradual transitions between levels of abstraction also provides a better learning experience in terms of students' perceptions and attitudes toward the instructional approach and the activity of database modeling. The multi-layer TSSL model, with origins in the field of HCI, served as a conceptual framework for understanding the activity of database modeling. The TSSL lens combined with Gestalt visualization principles offer ways of accounting for our results.

This study contributes to the field of IS education by demonstrating in a controlled experiment how an instructional approach can improve students' performance in modeling tasks and help students feel more positive towards the complex topic of conceptual modeling. We believe that our ideas for visualizing the gradual transitions between hierarchic levels can be used both in classroom settings and for the future design of computer-based environments for learning and exercising database schema modeling.

7. REFERENCES

Al-Dmour, A. (2010). A Cognitive Apprenticeship Based Approach to Teaching Relational Database Analysis and Design. *Journal of Information & Computational Science*, 7, 2495-2502.

Arnheim, R. (1949). The Gestalt Theory of Expression. *Psychological Review*, 56(3), 156-171.

Bunch, J. M. (2009). An Approach to Reducing Cognitive Load in the Teaching of Introductory Database Concepts. *Journal of Information Systems Education*, 20(3), 269-275.

Connolly, T. M. & Begg, C. E. (2006). A Constructivist-Based Approach to Teaching Database Analysis and Design. *Journal of Information Systems Education*, 17(1), 43-54.

Dahan, M., Shoval, P., & Sturm, A. (2014). Comparing the Impact of the OO-DFD and the Use Case Methods for Modeling Functional Requirements on Comprehension and Quality of Models: A Controlled Experiment. *Requirements Engineering*, 19(1), 27-43.

de Winter, J. F. C. & Dodou, D. (2010). Five-Point Likert Items: T Test Versus Mann-Whitney-Wilcoxon (Addendum added October 2012). *Practical Assessment, Research, and Evaluation*, 15(1), 11.

Ehrenstein, W. H. (2008). Gestalt Psychology. *Encyclopedia of Neuroscience*. Springer-Verlag.

Elmasri, R. & Navathe, S. B. (2011). *Fundamentals of Database Systems*. Pearson.

Fessakis, G., Dimitracopoulou, A., & Komis, V. (2005). Improving Database Design Teaching in Secondary Education: Action Research Implementation for Documentation of Didactic Requirements and Strategies. *Computers in Human Behavior*, 21(2), 159-194.

Flood, R. L. & Carson, E. R. (1993). *Dealing with Complexity: An Introduction to the Theory and Application of Systems Science*. Springer Science & Business Media.

Foley, J. D., Van Dam, A., Feiner, S. K., & Hughes, J. F. (1990). *Computer Graphics Principles and Practice.* Reading, Massachusetts: Addison-Wesley.

Gandhi, M., Robertson, E. L., & Van Gucht, D. (1994). Leveled Entity Relationship Model. In *International Conference on Conceptual Modeling*, (pp. 420-436), Heidelberg, Germany: Springer.

Harper, S., Michailidou, E., & Stevens, R. (2009). Toward a Definition of Visual Complexity as an Implicit Measure of Cognitive Load. *ACM Transactions on Applied Perception (TAP),* 6(2), 1–18.

Holzinger, A., Kickmeier-Rust, M., & Albert, D. (2008). Dynamic Media in Computer Science Education; Content Complexity and Learning Performance: Is Less More? *Educational Technology & Society*, 11(1), 279–290.

Jamieson, S. (2004). Likert Scales: How to (ab)Use Them. *Medical Education*, 38(12), 1217-1218.

Kalyuga, S. & Sweller, J. (2014). The Redundancy Principle in Multimedia Learning. In R. E. Mayer (Ed.), *The Cambridge Handbook of Multimedia Learning* (2nd rev. ed., pp. 247–262). New York, New York: Cambridge University Press.

Karstens, B., Kreuseler, M., & Schumann, H. (2003). Visualization of Complex Structures on Mobile Handhelds. In *Proc. International Workshop on Mobile Computing*.

Katz, A. (2018a). Applying TSSL in Database Schema Modeling: Visualizing the Syntax of Gradual Transitions. *Journal of Technology and Science Education*, 8(4), 238-253.

Katz, A. (2018b). Visualizing the Syntax of Gradual Transitions in Teaching Database Modeling, In M. B. Nunes, P. Isaias, and P. Powell (Eds.) *Proceedings of the 11th International Conference on Information Systems IADIS IS 2018*, (pp. 228-232), Lisbon, Portugal: IADIS Press.

Katz, A. & Shmallo, R. (2015). Improving Relational Data Modeling through Learning from Errors. In A. P. Abraham, A. Palma dos Reis, and J. Roth (Eds.) *Proceedings of the International Conference on Information Systems IADIS TPMC 2015*, (pp. 198-202), Las Palmas de Gran Canaria, Spain: IADIS Press.

Katz, A. & Shmallo, R. (2016). Learning from Errors as a Pedagogic Approach for Reaching a Higher Conceptual Level in Database Modeling, *COGNISE workshop, as part of CAiSE 2016, the 28th International Conference on Advanced Information Systems Engineering,* (pp. 93-102), Ljubljana, Slovenia: Springer.

Kuzon, W. M., Urbanchek, M. G., & McCabe, S. (1996). The Seven Deadly Sins of Statistical Analysis. *Annals of Plastic Surgery*, 37, 265–272.

Ma, L., Ferguson, J., Roper, M., Ross, I., & Wood, M. (2009). Improving the Mental Models Held by Novice Programmers using Cognitive Conflict and Jeliot Visualisations. *ACM SIGCSE Bulletin,* 41(3), 166-170.

Mayer, R. E. & Fiorella, L. (2014). Principles for Reducing Extraneous Processing in Multimedia Learning: Coherence, Signaling, Redundancy, Spatial Contiguity, and Temporal Contiguity Principles. In R. E. Mayer (Ed.), *The Cambridge Handbook of Multimedia Learning* (2nd rev. ed., pp. 279–315), New York, New York: Cambridge University Press.

Müller, N. H., Liebold, B., Pietschmann, D., Ohler, P., & Rosenthal, P. (2017). Hierarchy Visualization Designs and their Impact on Perception and Problem Solving Strategies. In *Proceedings of the International Conference on Advances in Computer-Human Interactions*, (pp. 93-101).

Najarian, S. E. (1981). Organizational Factors in Human Memory: Implications for Library Organization and Access Systems. *The Library Quarterly*, 51(3), 269-291.
Norman, G. (2010). Likert Scales, Levels of Measurement and the "Laws" of Statistics. *Advances in Health Sciences Education*, 15(5), 625-632.
Pinna, B. (2010). New Gestalt Principles of Perceptual Organization: An Extension from Grouping to Shape and Meaning. *Gestalt Theory*, 32(1), 11-78.
Saundage, D., Cybulski, J. L., Keller, S., & Dharmasena, L. (2016). Teaching Data Analysis with Interactive Visual Narratives. *Journal of Information Systems Education*, 27(4), 233-247.
Schmutz, P., Heinz, S., Métrailler, Y., & Opwis, K. (2009). Cognitive Load in Ecommerce Applications – Measurement and Effects on User Satisfaction. *Advances in Human-Computer Interaction*, Article ID 121494.
Schwamborn, A., Thillmann, H., Opfermann, M., & Leutner, D. (2011). Cognitive Load and Instructionally Supported Learning with Provided and Learner-Generated Visualizations. *Computers in Human Behavior*, 27(1), 89-93.
Shneiderman, B. & Mayer, R. (1979). Syntactic/Semantic Interactions in Programmer Behavior: A Model and Experimental Results. *International Journal of Parallel Programming*, 8(3), 219-238.
Simon, H. A. (1962). The Architecture of Complexity. *Proceedings of the American Philosophical Society*, 106(6), 467–482.
Srinivasan, A. & Te'eni, D. (1995). Modeling as Constrained Problem Solving: An Empirical Study of the Data Modeling Process. *Management Science*, 41(3), 419-434.
Stickel, C., Ebner, M., & Holzinger, A. (2010). The XAOS Metric – Understanding Visual Complexity as Measure of Usability. In *Symposium of the Austrian HCI and Usability Engineering Group*, (pp. 278-290), Heidelberg, Germany: Springer.
Te'eni, D, (2018). Levels of Abstractions in HCI: Theory, Implications and Design. *SIGHCI 2018 Proceedings*.
Te'eni, D., Carey, J. M., & Zhang, P. (2005). *Human-Computer Interaction: Developing Effective Organizational Information Systems*. John Wiley & Sons, Inc.
Te'eni, D. & Sani-Kuperberg, Z. (2005). Levels of Abstraction in Designs of Human–Computer Interaction: The Case of E-mail. *Computers in Human Behavior*, 21(5), 817-830.
Tsai, T. W., Chang, T. C., Chuang, M. C., & Wang, D. M. (2008). Exploration in Emotion and Visual Information Uncertainty of Websites in Culture Relations. *International Journal of Design*, 2(2), 55–66.
van Merriënboer, J. J. & Sweller, J. (2005). Cognitive Load Theory and Complex Learning: Recent Developments and Future Directions. *Educational Psychology Review*, 17(2), 147-177.
Wagemans, J., Elder, J. H., Kubovy, M., Palmer, S. E., Peterson, M. A., Singh, M., & von der Heydt, R. (2012). A Century of Gestalt Psychology in Visual Perception: I. Perceptual Grouping and Figure–Ground Organization. *Psychological Bulletin*, 138(6), 1172-1217.
Watson, R. T. (2006). The Essential Skills of Data Modeling. *Journal of Information Systems Education*, 17(1), 39-41.
Wertheimer, M. (1912a): Über das Denken der Naturvölker. *Zeitschrift für Psychologie*, 60, 321-378.
Wertheimer, M. (1912b): Untersuchungen über das Sehen von Bewegung. *Zeitschrift für Psychologie*, 61, 161- 265.

AUTHOR BIOGRAPHY

Adi Katz is the head of the industrial and management department in Sami Shamoon College of Engineering (SCE) in Ashdod, Israel. She is also the head of the information systems track in the department. Katz specializes in the areas of human-computer interaction (HCI), computer-mediated communication (CMC), and computer supported cooperative work (CSCW). Currently, her main research interests include cognitive and affective aspects of user interface (UI) design, visualization, and conceptual database modeling. Katz has published papers in journals such as *Organizational Science* and *Interacting with Computers (IwC)*. Her paper in IwC, entitled "What is Beautiful is Usable" (co-authored with N. Tractinsky and D. Ikar), has been cited over 1,640 times.

iscap

Information Systems & Computing
Academic Professionals

EDSIG

Serving Information Systems Educators

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 2574-3872