

# A WEB-BASED TOOL TO ENHANCE TEACHING/LEARNING DATABASE NORMALIZATION

**Hsiang-Jui Kung**  
Georgia Southern University  
hjkung@georgiasouthern.edu

**Hui-Lien Tung**  
Troy University  
tungh@troy.edu

## Abstract

*It has been difficult to motivate students to learn database normalization because students think the subject dry and theoretical. A web-based tool is developed to give students an interactive hands-on experience in database normalization process. The tool is suitable for relational data modeling in systems analysis and design and data management courses. This paper describes the web-based tool and its effectiveness in teaching relational data model. The effectiveness of the tool has been evaluated in surveys. The paper shows that the tool has a positive impact on students' perception.*

**Keywords:** Database Normalization, Relational Data Model, Functional Dependency, Third Normal Form

## Introduction

Database normalization is a well developed field since the introduction of Codd's seminal work on normal forms in 1970. Bernstein (1976), Diederich and Milton (1988), Concepcion and Villafuerte (1990), and Rosenthal and Reiner (1994) proposed algorithms and tools to synthesize a normalized database using functional dependencies. Maier (1988) indicated that relational data model theory (normalization) tends to be complex for the average designers. Jarvenpaa and Machesky (1989) and Bock and Ryan (1993) showed that relational data model leads to poor designer performance. The poor performance of normalization indicated that teaching normalization is a challenge to IS/IT educators.

The classical database normalization technique has often relied on the definition of normal forms. Some database textbooks include normalization algorithms to find the canonical cover by removing extraneous attributes of functional dependencies (FDs) and then to convert each FD in the canonical cover to a relation/table (Silberschatz et al, 2002). The normalization algorithms often require extensive relational algebraic backgrounds that most IS/IT students lack. Most systems analysis and design textbooks rely on the definition of normal forms in their coverage of database normalization (Hoffer et al., 2005; Avison and Fitzgerald, 2002). They simply give the definition of first, second and third normal forms (1NF, 2NF and 3NF) and hope that students will be able to apply the definitions and normalize a set of tables. These approaches may not be the best way to help many IS/IT students effectively understand the database normalization process.

The main objective of this paper is to describe the web-based database normalization tool and its effectiveness in teaching and learning of normalization. The remainder of this paper is organized as follows. Section 2 describes the web-based tool, and section 3 illustrates research design and data collection procedure. The effectiveness of the web-based tool is evaluated and interpreted in section 4, and section 5 concludes the paper.

## The Web-based Normalization Tool

It has been noticed that the decompositions happened when attributes on the right-hand side of functional dependencies have more than one copy, and the number of the decomposed relations is exactly the same as the

number of functional dependencies. To decompose a relation into the third normal form, one simply eliminates extraneous attributes on the right-hand side of the functional dependencies. The cookbook normalization technique is easy to follow without extensive background in relational algebra and database theories. The following steps describe the normalization process by using the example of a universal table T and a set of FDs:  $FD_1$ ,  $FD_2$  and  $FD_3$ .

The universal relation T (A, B, C, D, E, F)

$$FD_1 : A \rightarrow B, C, D$$

$$FD_2 : B \rightarrow C, D$$

$$FD_3 : A, E \rightarrow B, C, D, F$$

1. All functional dependencies on the left-hand side must keep their attributes intact.
2. Extraneous attributes on the right-hand side should be eliminated. This step is to eliminate partial dependencies and transitive dependencies. Repeated right-hand side attributes should be identified in all the functional dependencies, one copy of the redundant attributes should be kept and the others should be deleted. The rules of thumb about which copy of attributes to keep are
  - 1) the attributes that have fewer numbers of attributes on the left-hand side should be kept (this step will eliminate partial dependency),  
 Example: Attributes B, C, and D depend on part of the whole key (attribute A). Attributes B, C, and D in functional dependency  $FD_3$  will be deleted, since attributes B, C, and D appear in  $FD_1$  and  $FD_1$  has only one attribute on the left-hand side. A new functional dependency  $FD_3' : A, E \rightarrow F$  is formed.
  - 2) when two FDs have the same number of attributes on the left-hand side, the attributes that have the fewer numbers of attributes on the right-hand side should be kept (this step will eliminate transitive dependency).  
 Example: Attributes C and D appear in functional dependencies  $FD_1$  and  $FD_2$ . Attributes C and D are transitively dependent on Attribute A. Attributes C and D will be deleted from functional dependency  $FD_1$ , since  $FD_1$  has more right-hand side attributes than functional dependency  $FD_2$ . A new functional dependency  $FD_1' : A \rightarrow B$  is formed.
3. Construct relations. This step is to convert the functional dependencies without extraneous right-hand side attributes to relations. The new functional dependencies are as followings:

$$FD_1' : A \rightarrow B$$

$$FD_2 : B \rightarrow C, D$$

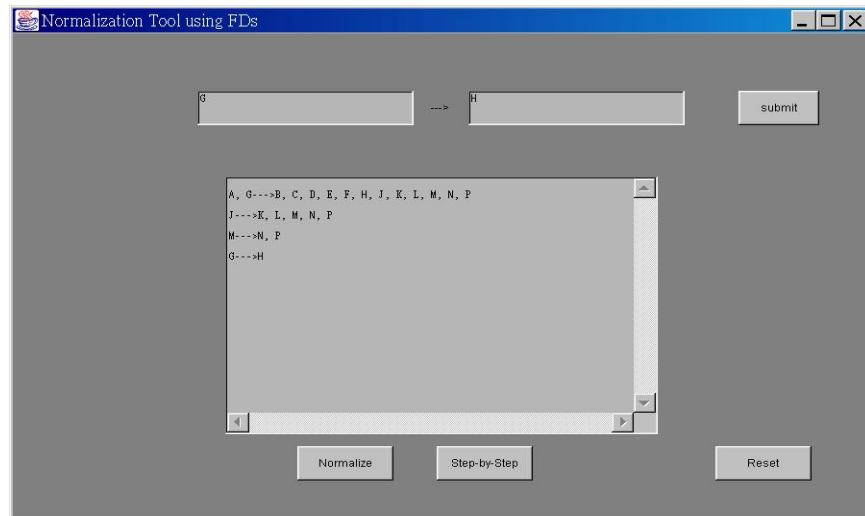
$$FD_3' : A, E \rightarrow F$$

The final normalized relations are exactly the same as the results of the decomposition algorithm and SA&D normalization techniques.

- $T_1'' (\underline{A}, B)$
- $T_2'' (\underline{B}, C, D)$
- $T_3'' (\underline{A}, \underline{E}, F)$

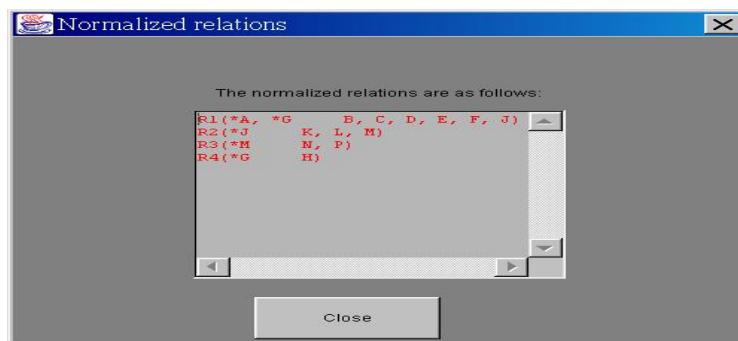
The web-based normalization tool is developed using Java applet based on the above normalization technique. The tool has the following features:

- Main window: The main window contains fields to key in and display functional dependencies and buttons to submit functional dependency one at a time, to reset/clear the memory for another exercise/practice, to normalize the database and display the normalized relations, and to display the step-by-step normalization process (Figure 1).



**Figure 1. A screenshot of the main window of the web-based normalization tool**

- Result window: Keying all the functional dependencies and then pressing  button, the user will see the normalized result in the normalized relations window (Figure 2). The tool will normalize the database based on the functional dependencies the user submitted. The tool normalizes any set of tables to 3NF if the set of functional dependencies used are non-trivial and closed. A functional dependency  $A \rightarrow U$  is non-trivial if  $A \cap U = \emptyset$ . In other words, the left-hand side and the right-hand side of a non-trivial functional dependency have no attributes in common. A functional dependency  $A \rightarrow U$  is closed under a set of functional dependencies FD if U is the set of all attributes that are functionally dependent on A given FD. The sets of functional dependencies used in this paper do meet these requirements.



**Figure 2. A screenshot of the normalized relations**

- Step-by-step window: Normalizing the database and then pressing  button, the user will see the step-by-step window showing the step-by-step normalization process (Figures 3-6). Figure 3 shows the set of FDs that the user keys in. Pressing  button, the user will see the normalization to 2NF that eliminates partial dependencies (Figure 4). Go to the next step, the user will see the normalization to 3NF that eliminates transitive dependencies (Figure 5). Figure 6 shows the normalized result as Figure 2. The user can navigate back and forth in the normalization process when he or she has problem with the normalization process.

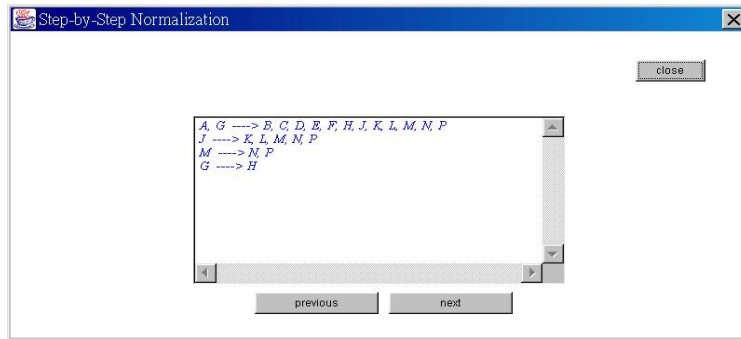


Figure 3. A screenshot of the step-by-step normalization process (1)

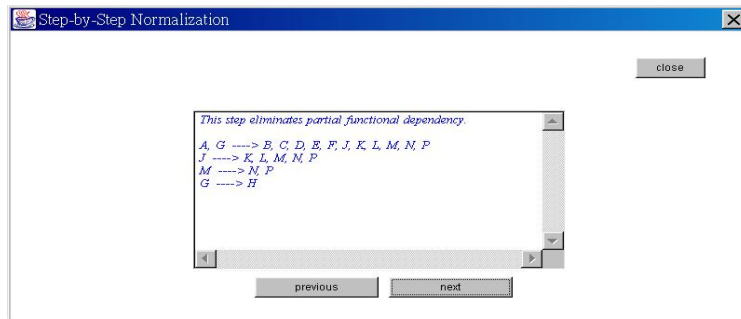


Figure 4. A screenshot of the step-by-step normalization process (2)

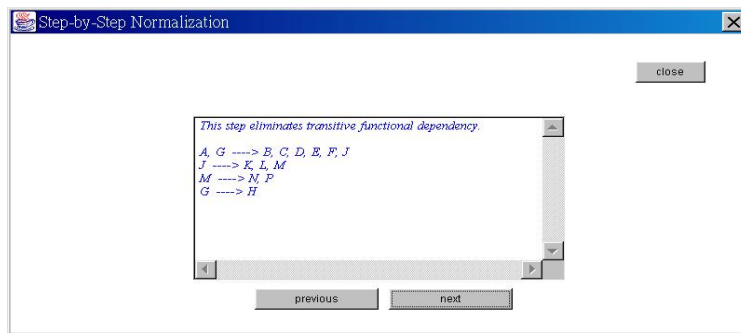


Figure 5. A screenshot of the step-by-step normalization process (3)

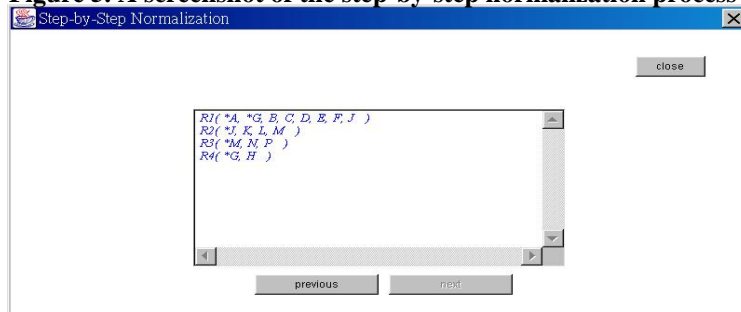


Figure 6. A screenshot of the step-by-step normalization process (4)

## Research Method

### Design of the Study

Approximately 45 students in two sections of a junior level systems analysis and design course participated in this study. The 15-week course met twice weekly. The Hoffer et al. (2005) textbook was used to cover feasibility study, data modeling, process modeling and physical design. Students spent four, 75-minute sessions on database normalization processes. The instructor spent one session explaining the importance of database normalization. The next session, the instructor demonstrated the textbook normalization technique using three normal forms with examples. The third session, students worked on two exercises using normalization techniques learned from previous session. The fourth session, students exposed to the cookbook normalization technique and the web-based tool to solve the in-class exercises.

The research design is a single group pretest-posttest design. This design includes pretest measure followed by a treatment and a posttest for a single group. Surveys were used to measure students' perceptions about the effectiveness of the textbook technique and the web-based tool in helping them learning database normalization. The survey also asked students to provide their perceptions of their performance of the textbook technique and the tool.

### In-Class Exercises

Two exercises were given to students as follows:

Exercise A:

$S(\underline{A}, B, C, D, E, F, \underline{G}, H, J, K, L, M, N, P)$

FDs:

$FDC_1: A, G \rightarrow B, C, D, E, F, H, J, K, L, M, N, P$

$FDC_2: J \rightarrow K, L, M, N, P$

$FDC_3: M \rightarrow N, P$

$FDC_4: G \rightarrow H$

Exercise B:

$Sales(\underline{Invoice\#}, Date, \underline{Line\#}, Quantity, S\_Price, Coupon\_Deduction, Savings, Sub\_Total, Product\#, P\_Name, P\_Price, Vendor\#, V\_Name, V\_City)$

FDs:

$FDD_1: Invoice\#, Line\# \rightarrow Date, Quantity, S\_Price, Coupon\_Deduction, Savings, Sub\_Total, Product\#, P\_Name, P\_Price, Vendor\#, V\_Name, V\_City$

$FDD_2: Product\# \rightarrow P\_Name, P\_Price, Vendor\#, V\_Name, V\_City$

$FDD_3: Vendor\# \rightarrow V\_Name, V\_City$

$FDD_4: Invoice\# \rightarrow Date$

The two exercises are almost identical. They have the same number of functional dependencies, and the same number of attributes. The only difference between these two exercises is that Exercise A is a hypothetical example and Exercise B is a close to real life example (Table 1).

**Table 1: Mapping Attributes between Two Exercises**

A	Line#	E	Savings	J	Product#	N	V_Name
B	Quantity	F	Sub_Total	K	P_Name	P	V_City
C	S_Price	G	Invoice#	L	P_Price		
D	Coupon_Deduction	H	Date	M	Vendor#		

### Procedure

Each student was asked to complete a survey questionnaire after he/she applied the textbook technique and the web-based tool in order to assess the students' perceptions of the instructional effectiveness of the textbook technique and the tool.

The questionnaire, which was administered at the end of the third and forth sessions, consisted of 5 questions, with two asking for demographic information (gender and number of courses taken previously in discrete mathematics, data structure, algorithms, and databases) and three questions that asked students about

1. the difficulty of understanding the textbook technique/tool;
2. the helpfulness of the textbook technique/tool;
3. the perceived grade/score of normalization after using the textbook technique/tool.

The questions were answered on a five-point Likert scale. An open-ended question was also included on the survey to enable the students to include their comments about the textbook normalization technique and the tool.

## Results and Analyses

Forty-four students completed the survey of the textbook technique and forty-five completed the survey of the tool (completion was voluntary). The sample was predominantly male (37 male vs. 7). Nineteen students entered the SA&D course with no prior database knowledge while 26 had some database knowledge from previous courses. Table 2 summarizes the distribution of student demographics across treatment conditions.

**Table 2: Demographic Frequency Counts**

Technique	Gender		Course	
	Female	Male	0	≥1
Textbook	7	37	19	25
Tool	7	38	19	26

The next set of analyses was performed to provide a general overview of student perceptions of the textbook technique and the web-based tool. The means and standard deviations for the three items on Part II of the survey questionnaire were calculated for both the textbook and tool (see Table 3). With three being the mid-point on the scale, Table 5 illustrates that students generally viewed the tool more positively (with mean scores hovering around 4.0) than the textbook technique (whose mean scores hovered around 2.4 to 3.1). Table 3 also indicates that students generally perceived that they knew normalization better, grade-wise, using the tool.

**Table 3: Mean Student Survey Responses for Textbook and Tool**

	Textbook		Tool	
	Mean	Std. Dev.	Mean	Std. Dev.
Difficulty	2.39	.970	3.98	.753
Helpfulness	3.00	1.012	4.22	.765
Perceived Grade	3.07	.789	3.89	.832

One-way multivariate analyses of variance (MANOVA) was used to compare the impact of the investigation's independent variables [technique type (textbook vs. tool), gender (male vs. female), and number of previous database courses (none vs. one or more)] on the pattern of student responses to survey Items 1-3 of the survey questionnaire. The one-way MANOVA revealed that the pattern of means for the textbook normalization technique observed in Table 2 is statistically significant from that for the tool. The means reported in Table 2 suggested that student perceptions of the tool were collectively more positive than their perceptions of the textbook normalization technique. The significant multivariate  $F$  observed in the MANOVA indicates that this is indeed the case. Gender and courses had no significant impact on students' responses.

**Table 3: Effects of Independent Variables on Student Response Pattern**

Independent Variable	Multivariate F	Significance
Technique (textbook vs. tool)	13.122 (df = 3, 79)	$p < .001$
Gender (male vs. female)	.241 (df = 3, 79)	ns
Courses (none vs. one or more)	1.541 (df = 3, 79)	ns

The analyses that were performed consistently demonstrated a student preference for the tool over the textbook technique. This preference was observed across a large range of survey responses in MANOVAs. Preference for the

tool over textbook normalization technique was also observed in responses to the open-ended question on the survey questionnaire. Most comments about the traditional technique were negatively toned. They suggest that students felt that the textbook technique took more time and effort to master (especially after exposure to the tool). In contrast, most comments about the tool were positively toned. They suggest that it helped students validate/check their learning of normalization process.

The potential reasons that previous coursework was not observed to be a predictor of the perceptions of normalization techniques are many. First, the courses that students normally take prior to enrolling in the SA&D course typically include content that is restricted to theoretical foundations of data modeling. In the SA&D course, however, the focus is on the application of normalization techniques, not the theoretical underpinnings of these techniques. Thus, the content of previous IS/IT coursework may account for why it had no significant impact on perceptions of the normalization techniques that participants were exposed to in this course/investigation. A second explanation may be the student's ability to perceive the relevance of previous coursework to current instruction. Felder (1994), for example, states that students rarely come to class thinking of the connection or relationship between previous lessons and the current ones. This may result from students forgetting the materials in previous courses or an inability to make connections between theories and applications.

## Conclusions and Future Research Direction

A web-based normalization tool has been developed to enhance teaching and learning of database normalization. The tool is user friendly and can be accessed through the Internet. It was evaluated by students and found to be robust. Students' responses to the tool were mostly favorable. The students indicated that they had found the tool easy to use and the step-by-step feature helped them gain understanding of database normalization process. The tool has a positive impact on students' perceptions.

Currently, the web-base tool allows a set of 10 functional dependencies, which is adequate for teaching purposes. More features, e.g., 'Load', 'Save', and 'Print' are still under development. The tool can be expanded to draw entity-relationship (ER) diagram. The incorporation of relational model and ER model is suggested for future work. Future studies are needed to test the robustness of these findings and the limits to which they are generalizable.

## References

- Avison, D. E. and Fitzgerald, G., 2002. *Information Systems Development: Methodologies, Techniques and Tools*, 3<sup>rd</sup> Ed., London, UK: McGraw Hill.
- Bernstein, P. A. 1976. Synthesizing Third Normal Form Relations from Functional Dependencies, *ACM Transactions on Database Systems*, Vol. 1. No. 4, pp 277-298.
- Bock, D. B., Ryan, T., 1993. Accuracy in modeling with extended entity relationship and object oriented data models. *Journal of Database Management*, 4 (4), 30-39.
- Codd, E. F., 1970. A Relational Model of Data for Large Relational Databases, *Communications of the ACM*, 13 (June), 377-387.
- Concepcion A. I., & Villafuerte R. M., 1990. Expert DB: An assistant database design system, *Proceedings of the third international conference on Industrial and engineering applications of artificial intelligence and expert systems - Volume 1*.
- Date, C. J., 2000. *An Introduction to Database Systems* 7<sup>th</sup> Ed., Addison-Wesley, Reading, MA.
- Diederich, J. and Milton, J., 1988. "New Methods and Fast Algorithms for Database Normalization", *ACM Transactions on Database Systems*, 13 (3), 339-365
- Felder, R., 1994. "Things I Wish They Had Told Me", *Journal of Information Systems Education*, 5 (1).
- Hoffer, J. A., George, J. F., and Valacich, J. S., 2005. *Modern Systems Analysis & Design*, 4<sup>th</sup> Ed., Prentice Hall.
- Jarvenpaa, S.L., Machesky, J.J., 1989. Data analysis and learning: an experimental study of data modelling tools. *International Journal of Man-Machine Studies*, 31, 367-391.

- Maier, D., 1988. *The Theory of Relational Databases*. Computer Science Press: Rockville, MD.
- Rosenthal, A., & Reiner, D., 1994. Tools and Transformations — Rigorous and Otherwise — for Practical Database Design, *ACM Transactions on Database Systems*, 19 (2), 167-211.
- Silberschatz, A., Korth, H. F., and Sudarshan, S., 2002, *Database System Concepts*, 4<sup>th</sup> Edition, Boston, MA: McGraw Hill.