THE EFFECTS OF NORMALISATION ON THE SATISFACTION OF NOVICE END-USER QUERYING DATABASES*

Conrad Benedict
Paul Bowen
Colin Ferguson
Fiona Rohde
Department of Commerce
The University of Queensland, Qld, 4072
e -mail: rohdc@commerce.uq.edu.au

ABSTRACT

This paper reports the results of an experiment that investigated the effects different structural characteristics of relational databases have on information satisfaction of end-users querying databases. The results show that unnormalised tables adversely affect end-user satisfaction. The adverse affect on end-user satisfaction is attributable primarily to the use of non atomic data. In this study, the affect on end user satisfaction of repeating fields was not significant. The study contributes to the further development of theories of individual adjustment to information technology in the workplace by alerting organisations and, in particular, database designers to the ways in which the structural characteristics of relational databases may affect end-user satisfaction. More importantly, the results suggest that database designers need to clearly identify the domains for each item appearing in their databases. These issues are of increasing importance because of the growth in the amount of data available to end-users in relational databases.

INTRODUCTION

The complexity and pervasiveness of information systems based on relational database management software continues to grow. End-users are increasingly expected to query these systems to obtain the information they need to perform their jobs. In addition, relational database technology has enhanced the ability of end-users to (a) design their own database structures, and (b) maintain their own data (Doll and Torkzadeh, 1993). These data have value only to the extent end-users can obtain the information they need from these databases.

The growth in end-user computing warrants an investigation of factors that improve end-user performance and satisfaction because, for example, end-user information satisfaction (UIS) is associated with systems success (Doll & Torkzadeh, 1988b). An understanding of how these factors affect UIS enables designers and managers to enhance system effectiveness through improved user education and training.

Data normalisation and, through it, task complexity are the foci of this study. Earlier research found that the perceived ease-of-use of a system is an important component of UIS (Doll & Torkzadeh, 1988b). Perceived ease-of-use of a database is diminished when the complexity of tasks increases. Therefore, through its effect on perceived ease-of-use, task complexity affects UIS. In turn, task complexity associated with the use of databases is affected by the level of normalisation. This study focuses on these relationships (as depicted in Figure 1) and investigates how different levels of normalisation and, therefore, task complexity affect novice end-users' satisfaction when querying[1] relational databases.

* The authors gratefully acknowledge the assistance of the editor and the two anonymous referees for their comments on earlier drafts of this paper. The authors also gratefully acknowledge the financial assistance provided by The University of Queensland via a new staff grant.

[1] End users can interact with the database through the use of a query language. Queries can include insert, delete, update and data selection statements. This research limits the use of the word query/querying to include only data selection statements.

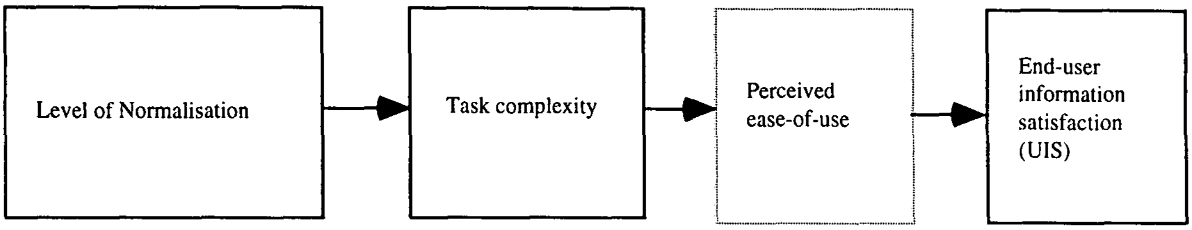

Figure 1: Normalisation, task complexity and end-user information satisfaction

Although normalisation, originally outlined by Codd (1970) when he introduced the relational model, has been the focus of earlier research (see, for example, Maciaszek, 1990), little empirical research has examined the effects of various levels of normalisation on UIS. Through an experiment, this study examines UIS associated with querying databases exhibiting three different data structures: not normalised (¬NF), first normal form (1NF), and third normal form (3NF).

The paper proceeds as follows. The next section develops the theory on which the experiment was based and describes the key research constructs. The next two sections present example queries and develop the research hypotheses. The subsequent section outlines the research method. Results are then reported and analysed. The paper concludes with a discussion of the implications of the study's findings, its limitations, and suggestions for future research.

THEORY DEVELOPMENT

Research model

The model depicted in Figure 1 shows that for databases, user information satisfaction is influenced by the level of normalisation through its effect on the level of task complexity. The following subsections discuss end-users and the components of the model.

Novice End-users

End-user computing is the direct interaction with application software by managerial, professional, and operating level personnel in user departments (Doll and Torkzadeh, 1993). End-users are often novices and frequently reluctant adopters of information systems technology (Rockhart and Flannery, 1983; Doll and Torkzadeh, 1993). Some recent research (see, for example, Mackay & Elam, 1992; Kieras & Bovair, 1984; Soloway et al, 1982) has focused on novice end-users. Overall, these studies suggest that end-users are not aware of the analysis required to solve complex problems, and that this lack of awareness is heightened in novice end-users. Novice end-users are likely to perform poorly and be dissatisfied in complex task environments. Improving their performance and increasing their levels of satisfaction are likely to lead to more effective and efficient use of information systems resources in organisations.

Task Complexity

A complex task has three primary properties: the number of dimensions of information requiring attention (*information load*), the number of alternatives associated with each load (*information diversity*), and the degree of uncertainty involved (*rate of information change*) (Schroder et al., 1971). As the magnitude of each property increases, so does task complexity (Campbell, 1988). This implies that a change in the structural characteristics of databases changes the magnitude of the dimensions of task complexity. Therefore, changes in the level of normalisation in a database structure change the level of task complexity for users of that database.

Normalisation

Normalisation is a simple elimination procedure (Codd, 1970). Codd initially proposed three normal forms that he called first, second, and third normal form. Other normal forms include Boyce-Codd, fourth, and fifth normal forms[2]. Each level of normalisation is generally considered more desirable than the levels below it (Date, 1986). Increasing the level of normalisation changes specific structural characteristics, e.g., atomicity, repeating fields, and fragmentation (see Figure 2). For a relation to be in 1NF, "at every row-and-column position within the relation, there is always exactly one data value, never a set of multiple values" (Data, 1990, p 378). Therefore, when the level of normalisation changes from ¬NF to 1NF, all repeating fields are eliminated and non-atomic[3] data items are decomposed into multiple atomic elements. When the level of normalisation increases from 1NF to 3NF, all partial and transitive dependencies (two examples of functional dependencies) are eliminated. The removal of repeating fields, non-atomic data items, and these two types of functional dependencies, however, results in an increase in the fragmentation of the database. The existence or elimination of these structural characteristics is likely to affect the complexity of the database tasks and, consequently, the level of perceived ease-of-use of the database system.

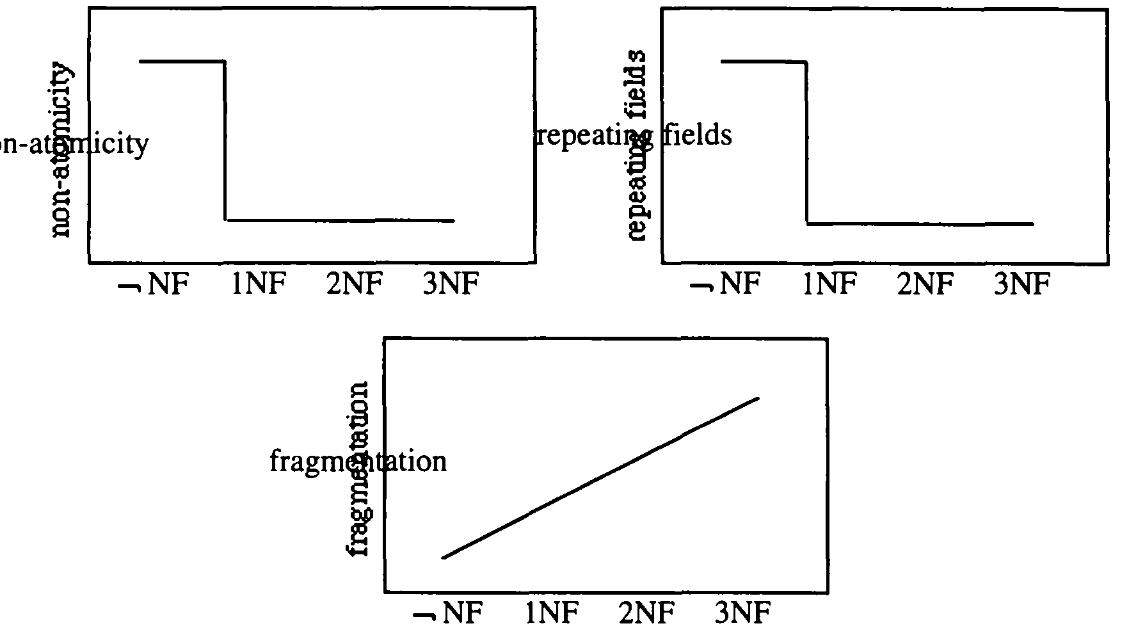

Figure 2: Structural characteristics and anomalies of normalisation

EXAMPLE QUESTIONS AND THE RESULTING QUERIES

This section presents two of the questions the experimental participants were asked to answer and provides queries that answer these questions using each of the three data structures. The differences in complexity between these queries relative to the three data structures are then discussed.

The experiment required participants to obtain information about a furniture manufacturer's just-in-time operations. The participants queried a database that contained data about raw materials inventory items including receipts, issues, and current amounts on hand. See Appendix B for the data structures and Appendix C for details of the experimental setting. For readability, SQL keywords appear in upper case letters; table names appear in small capital letters; and attribute names appear in lower case, italicized letters.

Find vendors with discount rates > 5% or discount days > 45.

¬NF query

```
SELECT DISTINCT vno, vname, terms
FROM ITEMA
WHERE TO_NUMBER(SUBSTRING(terms,1,INSTR(terms,'/')-1) > 5
      OR TO_NUMBER(SUBSTRING(terms,INSTR(terms,'/')+1,2) > 45;
```

[2]Because the experiment only used ¬NF, 1NF, and 3NF, the definitions of Boyce-Codd normal form (BCNF), fourth normal form (4NF), and fifth normal form (5NF) are not presented in this paper.

[3]The atomicity constraint requires that each individual data value be atomic in that they have no internal structure (i.e., are non-decomposable (Date, 1990). For example, a single address field that contained both city and state data violates the atomicity constraint.

Sources of complexity

One source of complexity in the query is that multiple records contain *terms* data for eachvendor, i.e., each *vno*. Assuming that each vendor uses the same discount rate and discount days for all items, the output of the query should contain only one item per vendor. Hence the keyword DISTINCT is required to satisfy this requirement. Indeed for the ¬NF data structure, there can be multiple records for each item supplied by an individual vendor.

The primary source of complexity in the above query is the lack of atomicity of the attribute *terms*. As can frequently be the case, this lack of atomicity is also associated with less specific data types. That is, rather than two numeric attributes, the discount rates and days are combined into one character attribute. This means that integrity constraints are much more difficult to implement (for example, range checks) and that a particular format must be assumed. The assumed format may be difficult to enforce or ensure. In this case, the format is assumed to be the discount rate, a slash as a separator, and the discount days. If this format is not followed for each and every record in ITEMA, then the results of the query may not be correct. Hence, to answer the above question using the ¬NF data structure requires making a strong assumption about the format of *terms*, parsing the contents to extract the desired subset of characters, and converting those characters to a numeric value.

1NF query

```
SELECT DISTINCT vno, vname, termdisc, termdays
FROM ITEMB
WHERE termdisc > 5 OR termdays > 45;
```

Relative to the queries of the other two data structures, the only additional source of complexity present in the 1NF query is that multiple records can contain *terms* data for each vendor, i.e., each *vno*. Unlike the ¬NF data structure, there are no multiple records for each item supplied by an individual vendor.

3NF query

```
SELECT itemc.vno, vname, termdisc, termdays
FROM ITEMC, VENDORC
WHERE (termdisc > 5  OR termdays > 45)
        AND ITEMC.vno = VENDORC.vno;
```

Because the discount rate and discount days data for each vendor only exists in a single record, the keyword DISTINCT is not needed in the 3NF query. Relative to the 1NF query, however, the 3NF query does contain at least three additional sources of complexity. First, because the attribute *vno* exists in both ITEMC and VENDORC, *vno* in the SELECT clause requires a table name qualifier to remove the ambiguity. Second, an additional table, VENDORC, must be specified in the FROM clause. Third, an additional condition in the WHERE clause is required to join the ITEMC and VENDORC tables so that the vendor's name can be displayed, i.e., the condition AND ITEMC.*vno* = VENDORC.*vno*.

Find items where the quantity on hand is < the average quantity issued.

¬NF query 1st attempt

```
SELECT DISTINCT itemno, idesc, vno, vname, cqtyoh, AVG(qtyiss1+qtyiss2+qtyiss3)/3
FROM ITEMA ALIAS1
WHERE ALIAS1.cqtyoh <
        (SELECT AVG(qtyiss1+qtyiss2+qtyiss3)/3
         FROM ITEMA ALIAS2
         WHERE ALIAS1.itemno = ALIAS2.itemno
         GROUP BY ALIAS2.itemno);
```

Sources of Complexity

One source of complexity in the query is that there can be multiple records for each item supplied by an individual vendor. The keyword DISTINCT is required to display only one line per item supplied by each vendor.

The primary source of complexity in the above query is the presence of repeating fields, i.e., *qtyiss1*, *qtyiss2*, and *qtyiss3*. These repeating fields increase complexity in a number of ways. First, instead of specifying a single attribute in the SELECT clause, e.g., *qtyiss* in the queries below of 1NF or 3NF, an average of the three attributes must be taken. Second, because the subquery in the WHERE clause will require joining the ITEMA table to itself, the FROM clause must specify an alias for ITEMA. Third, an intricate subquery is required to extract the data needed for the WHERE clause. Although the queries of both the 1NF and 3NF data structures also require subqueries, the subquery for the ¬NF data structure contains more terms, e.g., SELECT AVG(*qtyiss1+qtyiss2+qtyiss3*)/3 versus SELECT AVG(*qtyiss*). Fourth, the results of the query will be incorrect except for those items with exactly three issues, e.g., dividing *qtyiss1* + *qtyiss2* + *qtyiss3* by 3 when fewer than three issues have been made will not yield the correct average quantity issued for that item. One of the easiest ways to formulate a correct query using the ¬NF data structure is to create a view that places the required data into a first normal form data structure as shown below. After creating this view, the query is almost identical to the query of the 1NF data structure.

¬NF query, correct

```
CREATE VIEW ISSUESA AS
        SELECT itemno, qtyiss1 AS qtyiss
        FROM ITEMA
        WHERE qtyiss1 NOT NULL
        UNION  SELECT itemno, qtyiss2 AS qtyiss
              FROM ITEMA
              WHERE qtyiss2 NOT NULL
        UNION SELECT itemno, qtyiss3 AS qtyiss
              FROM ITEMA
              WHERE qtyiss3 NOT NULL;


SELECT DISTINCT ITEMA.itemno, idesc, vno, vname, cqtyoh, AVG(qtyiss)
 FROM ITEMA, ISSUESA
WHERE ITEMA.itemno = ISSUESA.itemno AND
        cqtyoh <  (SELECT AVG(qtyiss)
                   FROM ISSUESA
                   WHERE ITEMA.itemno = ISSUESA.itemno
                   GROUP BY itemno);
```

1NF query

```
SELECT ITEMB.itemno, idesc, vno, vname, cqtyoh, AVG(qtyiss)
 FROM ITEMB, ISSUESB
WHERE ITEMB.itemno = ISSUESB.itemno AND
        cqtyoh < (SELECT AVG(qtyiss)
                 FROM ISSUESB
                 WHERE ITEMB.itemno = ISSUESB.itemno
                 GROUP BY itemno);
```

Relative to the queries of the other two data structures, the 1NF query is the least complex, e.g., requires the least number of terms, qualifiers, and conditions.

3NF query

```
SELECT ITEMC.itemno, idesc, ITEMC.vno, vname, cqtyoh, AVG(qtyiss)
FROM ITEMC, ISSUESC, VENDORC
WHERE ITEMC.itemno = ISSUESC.itemno AND
        ITEMC.vno = VENDORC.vno AND
        cqtyoh < (SELECT AVG(qtyiss)
                 FROM ISSUESC
                 WHERE ITEMC.itemno = ISSUESC.itemno
                 GROUP BY itemno);
```

Relative to the 1NF query, the 3NF query contains the same additional sources of complexity as noted in the previous example.

HYPOTHESES FORMULATION

The foregoing theory development and example queries suggest task complexity, normalisation, and UIS provide foundations on which to analyse task complexity-data structure relationships and predict end-user attitudes. These constructs motivate research hypotheses that investigate end-users' query satisfaction at different levels of normalisation. End-users' UIS responses are explored relative to variations in task complexity caused by (a) lack of atomicity, (b) repeating fields, and (c) data fragmentation.

End-user Satisfaction: 1NF versus ¬NF

Non-atomic data and repeating fields cause complexity to increase. The increased complexity decreases end-user satisfaction. Normalisation removes repeating fields and makes all attributes conform to the atomicity constraint. Hence:

H1: End-users are more satisfied querying 1NF data structures than ¬NF data structures.

H2: The increased dissatisfaction of ¬NF end-users relative to 1NF end-users is associated with non-atomic data.

H3: The increased dissatisfaction of ¬NF end-users relative to 1NF end-users is associated with repeating fields.

End-user Satisfaction: 3NF versus 1NF

Higher levels of normalization cause fragmentation, i.e., increasing normalization from 1NF to 3NF results in a larger number of tables. This fragmentation requires joins to reassemble the data. These joins increase complexity and decrease satisfaction. Hence:

H4: End-users are more satisfied querying 1NF data structures than 3NF data structures.

H5: The increased dissatisfaction of 3NF end-users relative to 1NF end-users is associated with increased data fragmentation.

End-user Satisfaction: 3NF versus ¬NF

Relative to 1NF, complexity associated with non-atomic data and repeating fields in ¬NF data structures exceeds that associated with the increased fragmentation in 3NF data structures. Hence:

H6: End-users are more satisfied querying 3NF data structures than ¬NF data structures.

H7: The increased dissatisfaction of ¬NF end-users relative to 3NF end-users is associated with non-atomic data.

H8: The increased dissatisfaction of ¬NF end-users relative to 3NF end-users is associated with repeating fields.

RESEARCH METHOD

Research design, participants, and data collection

This study uses a laboratory experiment to control for various extraneous variables that may confound the observed results. The laboratory experiment facilitates control of data structures, equalises subject motivation, optimises subject participation, and allows random assignment of subjects to experimental groups (Campbell and Stanley, 1963). The experiment uses the posttest-only control group design explicated by Campbell and Stanley (1963)[4]. Using this design enhances the study's internal validity by controlling for potential problems such as history, maturation, statistical regression, testing, instrumentation, and selection-maturation[5].

The experiment required participants to record their attitudes after querying a database established for a Just-in-Time inventory system. The experiment was developed by Liu (1995). The databases of these information systems used three data structures: not normalised (¬NF), first normal form (1NF), and third normal form (3NF).

[4]This is one of three true experimental designs discussed by Campbell and Stanley (1963). In general, they prefer this experimental design over the other two: the pretest-posttest control group design and the Solomon four-group design.

[5]For a detailed discussion of these potential threats to internal validity, see Campbell and Stanley, 1963.

Eighty undergraduate and masters level commerce and information technology students participated in the experiment[6]. Most participants had little experience with relational database querying yet were familiar with general computing concepts and activities. Prior to the experiment, all participants completed two projects in which they were required to create tables and forms and then query the tables in SQL.
Based on their information systems background, participants were randomly assigned to one of the three treatment groups as follows: an information systems expert ranked each participant in descending order, i.e. the person considered to have the most extensive information systems background was ranked number 80, the person with the next most extensive background was ranked 79, etc. The order of the three treatments was then randomised, with the order being: ¬NF, 3NF, and 1NF. Participants were randomly[7] assigned to the groups according to their information systems background, so as to eliminate any experience effect.
Data used to measure the end-user satisfaction variable was collected using a questionnaire. Existing UIS instruments were not used because they could not investigate specific database characteristics without substantial modification and extension. A number of these instruments, however, were used as the basis for developing the UIS scales used in this study (see, for example, Ives, Olson & Baroudi, 1983; Doll & Torkzadeh, 1988). The questionnaire was administered after the completion of the experiment. The response rate was 95 percent, i.e., 76 questionnaires were returned of the 80 distributed.

Measures

Participants performed queries on parts of the experimental data structures that exhibited characteristics of: (a) non-atomic data, (b) repeating fields, and (c) increased fragmentation. The level of end-user satisfaction for querying activities associated with each characteristic was measured using specifically developed scales[8].
A two-item scale was used to measure the UIS of users querying data structures with varying levels of fragmentation. Item two's responses were reversed, added to item one, and their total divided by two. A single item scale was used to measure UIS (non-atomicity), a seven item scale was used to measure UIS (repeating fields), and a three item scale was used to measure UIS (overall). These scales are included in Appendix A.

RESULTS

Table 1 summarises the hypotheses and their results.

Hypothesis	Statement of Hypothesis	Result
H1	End-users are more satisfied querying 1NF data structures than ¬NF data structures	Supported
H2	The increased dissatisfaction of ¬NF end-users relative to 1NF end-users is associated with non-atomic data.	Supported
H3	The increased dissatisfaction of ¬NF end-users relative to 1NF end-users is associated with repeating fields.	Not supported, however, in the predicted direction.
H4	End-users are more satisfied querying 1NF data structures than 3NF data structures	Not supported

[6]Eining and Dorr (1991) argue that students are appropriate participants for research concerned with novice decision-makers.
[7]The method of randomisation was to assign participant 80 to ¬NF, participant 79 to 3NF, 78 to 1NF, 77 to 1NF, 76 to 3NF, 75 to ¬NF, 74 to ¬NF, and so on.
[8]Cronbach alphas were calculated to ensure the reliability of scales with more than one item. Only those questions that loaded significantly together in the factor analysis and had Cronbach alpha coefficients greater than 0.80 were selected as combined scales to measure specific UIS responses.

H5	The increased dissatisfaction of 3NF end-users relative to 1NF end-users is associated with increased data fragmentation.	Not supported, however, in the predicted direction.
H6	End-users are more satisfied querying 3NF data structures than ¬NF data structures.	Not supported, however, in the predicted direction.
H7	The increased dissatisfaction of ¬NF end-users relative to 3NF end-users is associated with non-atomic data.	Supported
H8	The increased dissatisfaction of ¬NF end-users relative to 3NF end-users is associated with repeating fields.	Not supported, however, in the predicted direction.

Table 1: Summary of Research Hypotheses and Results

Table 2 provides descriptive statistics of the UIS measures. Counts vary between measurements because of incomplete responses on the questionnaires.

Model	Mean	Std Dev	Count (N)
Non Atomicity UIS			
¬NF	2.5385	1.4486	26
1NF	3.7391	1.9590	23
3NF	3.2692	1.5889	26
Repeating Fields UIS			
¬NF	3.9167	1.2084	20
1NF	4.5431	1.6841	21
3NF	4.0500	1.5170	20
Fragmentation UIS			
¬NF	3.5962	1.0051	25
1NF	4.6737	1.3020	23
3NF	4.3800	1.0025	26
Querying UIS			
¬NF	3.1250	1.4166	25
1NF	4.0434	1.4165	23
3NF	4.4400	1.3613	24

Table 2: End-user information satisfaction - descriptive statistics

UIS (non-atomicity)

Analysis of variance (ANOVA) was used to determine whether end-users are more satisfied querying databases containing purely atomic data. As hypothesised (H2, H7), the results show that participants who used either 1NF or 3NF databases are more satisfied than those who used ¬NF databases ($F_{2,72} = 3.25$, $p = 0.0444$). The least square means, reporting the significance of the pairwise comparisons of the UIS values, are shown in Table 3.

UIS (repeating fields)

Statistically significant support was not found for the hypothesis that end-users are more satisfied querying databases with data structures containing no repeating fields (H3, H8) ($F_{2, 58} = .14$, $p = 0.3009$). The least square means (Table 3) show, however, that the relationships were in the predicted directions.
Inspection of their queries indicated that users of ¬NF, 1NF, and 3NF data structures entered similar queries. Therefore, the disadvantages of querying databases containing repeating fields was overcome by users repeatedly executing the same query and simply substituting field names.

UIS (fragmentation)

ANOVA results suggest a significant relationship between levels of normalisation and UIS arising from fragmentation (H5) ($F_{2,71} = 3.39$ $p = 0.0197$). Table 3 shows that no statistical difference was found between 1NF and 3NF, however, the relationship was in the predicted direction. Table 3 also reveals that users querying the ¬NF data structure were less satisfied than users querying the 1NF data structure. Similarly, users querying the ¬NF data structure were less satisfied than users querying the 3NF data structure. These results are the opposite to those predicted. These results suggest that users are more satisfied querying databases with a small amount of fragmentation than querying databases containing non atomic data times and repeating fields. This assertion is further supported in the next subsection.

Model	Normalisation Level	Least Square Means	Significance of pairwise comparison ¬NF	Significance of pairwise comparison 1NF
UIS (non atomicity) = F (norm_level)	¬NF	2.5385		
	1NF	3.7391	.0141	
	3NF	3.2692	.1183	.3280
UIS (repeating fields) = F (norm_level)	¬NF	3.4714		
	1NF	3.8980	.2885	
	3NF	3.2929	.6593	.1340
UIS (fragmentation) = F (norm_level)	¬NF	3.5962		
	1NF	4.6737	.0075	
	3NF	4.3800	.0341	.2514
UIS (querying overall)	¬NF	3.1818	-	
	1NF	3.6232	.0599	
	3NF	3.5278	.1082	.3644

(Note: norm_level = level of normalisation)

Table 3: Least squares means analysis

UIS (overall)

Statistically significant support was not found for the hypothesis that end-users are more satisfied querying databases with data structures in 1NF than either ¬NF or 3NF (H1, H4, H6) ($F_{2,69} = 1.37$, $p = 0.1305$). The least square means (Table 3) show, however, that the relationship was in the predicted direction.
The least square means results reported in Table 3 show a significant difference, in the direction predicted, between the ¬NF and 1NF data structures (H1). Users were more satisfied querying databases in 1NF than databases in ¬NF. Statistically significant support was not found for the proposition that users querying 3NF databases were less satisfied than users querying 1NF databases (H4). Investigation of the users' query commands also revealed that many users did not perform the query task intended to test the structural disadvantages of querying in 3NF, i.e., few questions were attempted that involved the fragmentation inherent in higher levels of normalisation. Statistically significant support was not found of the proposition

that users querying 3NF databases were more satisfied that users querying ¬NF databases (H6), however, the results were in the predicted direction.

DISCUSSION & CONCLUSIONS

This study provided empirical evidence about the effects of various levels of normalisation on UIS. The results show that lack of atomicity adversely affects user satisfaction. The results also show that different levels of fragmentation affect user satisfaction when querying databases. The results show that unnormalised tables adversely affect end-user satisfaction. The adverse affect on end-user satisfaction is attributable primarily to the use of non atomic data. In this study, the affect on end user satisfaction of repeating fields was not significant. The negative effects on UIS increased fragmentation associated with normalising databases to first normal form are overshadowed by the positive affects on UIS of eliminating non atomic data and repeating fields. The increased fragmentation resulting from further normalisation, however, appears to be associated with decreased UIS. Overall, the results suggest that end-user satisfaction is greatest when querying databases in 1NF, as this is the level with the least complex queries, i.e., atomic data, no repeating fields, and very little fragmentation. This reduction in task complexity culminates in greater end-user satisfaction.

These results are important to organisations because they impact the relationship between database designers and end-users. Database designers must be aware that normalisation levels affect both end-users' performance and satisfaction when querying databases. The results also emphasise the need for the provision of specific training programs for end-users. End-user training is of increasing importance as organisations increase their investment in information technology. Investigating the effects of normalisation on UIS enables organisations to properly direct their training programs and to maximise the benefits they receive from their information technology investments.

Two specific areas should be the focus of these training programs. First, end-users creating their own databases should be educated on the advantages afforded by creating databases in 1NF so they as end-users, as well as other end-users, can query the databases with relative ease. Users should also be aware of other problems associated with 1NF databases in relation to update anomalies and learn about the advantages and disadvantages of each level of normalisation. They should also be instructed on the creation of views to emulate a 1NF database so that querying, one of their more frequent tasks, can be accomplished with efficiency and effectiveness.

Second, information technology professionals need to be aware that the databases created by them may be highly fragmented. The negative effect that this fragmentation may have on query complexity and UIS suggest that end-users should be educated on how to overcome the difficulties of querying fragmented databases. When end-users consistently use databases created in higher normal forms further training may help them create views (as was required for example in one question discussed in section 3) of the data that may be approximately equivalent to 1NF to make querying easier and more satisfying. Another issue arising from this research is the significant negative affect that non atomic data has on user satisfaction. This finding has implications for database designers in that they need to clearly identify the domains for each item appearing in their databases, and ensure that they have no internal structure, that is, are atomic in nature.

This study adopted Doll and Torkzadeh's (1993) assumption that the majority of end-users have low skill levels and generally fall outside the category of experts in most common computer applications. Future research should assess cognitive differences between novice and expert end-users. This assessment would assist information technology professionals who create databases for end-users. Such research would also help such professionals to better assist end-users who develop and query their own databases. Greater knowledge of cognitive characteristics of end-users could also improve the content and relative emphasis of the training provided to them. Future research should also test user satisfaction in across a broader range of user queries. Completion of this research in future would enable an overall strategy to be adopted that would lead to both efficient and effective database operations and high levels of user satisfaction.

REFERENCES

Batra, D., Hoffer, J.A., & Bostrom, R. P. (1990) "Comparing Representations With Relational And EER Models", **Management And Computing,** Vol 33 (2) pp 126-138.

Bowen, P.L., (1995) "The Effects Of Different Levels Of Normalisation On Data Entry And Data Entry Controls: An Experimental Evaluation", Working Paper, University Of Queensland.

Box, G.E.P., Hunter, G.W., & Hunter, J.S., (1979) **Statistics For Experimenters: An Introduction To Design, Data Analysis, And Model Building,** John Wiley & Sons, Brisbane.

REFERENCES

Batra, D., Hoffer, J.A., & Bostrom, R. P. (1990) "Comparing Representations With Relational And EER Models", **Management And Computing**, Vol 33 (2) pp 126-138.

Bowen, P.L., (1995) "The Effects Of Different Levels Of Normalisation On Data Entry And Data Entry Controls: An Experimental Evaluation", Working Paper, University Of Queensland.

Box, G.E.P., Hunter, G.W., & Hunter, J.S., (1979) **Statistics For Experimenters: An Introduction To Design, Data Analysis, And Model Building**, John Wiley & Sons, Brisbane.

Campbell, D.J., (1988) "Task Complexity: A Review And Analysis", **Academy Of Management Review**, Vol 13 (1) pp 40-52.

Campbell, D.J., (1984) "The Effects Of Goal-Contingent Payment On The Performance Of A Complex Task", **Personnel Psychology**, Vol 37, pp 23-40.

Campbell, D., & Gingrich, K.F., (1986) "The Interactive Effects Of Task Complexity And Participation On Task Performance: A Field Experiment", **Organisational Behaviour And Human Decision Processes**, Vol 38, pp 162-180.

Campbell, D.T., & Stanley, J.C., (1963) **Experimental And Quasi-Experimental Designs For Research**, Houghton Mifflin Company, Boston.

Card, S.K., Moran, T.P. & Newell, A. (1983) **The Psychology Of Human-Computer Interaction**, Erlbaum, Hillsdale.

Codd, E.F., (1970) "A Relational Model Of Data For Large Shared Data Banks", **Communications Of The ACM**, Vol 13 (6) pp 377-87.

Codd, E.F., (1974) "Recent Investigations In Relational Data Base Systems", **Information Processing**, Vol 74 pp 1017-87.

Cook, T.D., & Campbell, D.T., (1979) **Quasi-Experimentation Design And Analysis And Issues For Field Settings**, Houghton Mifflin, Boston.

Date, C.J., (1986) **Relational Database: Selected Writings, Reading** Addison-Wesley Publishing Company, Massachusetts.

Date, C.J., (1990) **An Introduction To Database Systems**, 5th Ed. Addison-Wesley Publishing Company, Massachusetts.

Doll, W.J., & Torkzadeh, G., (1993) "The Place And Value Of Documentation In End-User Computing", **Information And Management**, Vol 24 pp 147-158.

Doll, W.J., & Torkzadeh, G., (1988a) "The Quality Of User Documentation", **Information And Management**, June, pp 258-274.

Doll, W. J., & Torkzadeh, G. (1988b) "The Measurement Of End-User Computing Satisfaction", **MIS Quarterly**, June, pp 259 - 274.

Elmasri R., & Navathe S. B., (1989) **Fundamentals Of Database Systems**, The Benjamin-Cummings Publishing Company, Redwood City, California.

Ives, B., Olson, M., & Baroudi, S., (1983) "The Measurement Of User Information Satisfaction", **Communications Of The ACM**, Vol 26 (10) pp 785-793.

Kent, W. (1983) "A Simple Guide To Five Normal Forms In Relational Database Theory", **Communications Of The ACM**, Vol 26 (2) pp 120-125.

Kieras, D.E., & Bovair, S., (1984) "The Role Of Mental Model In Learning To Operate A Device", **Cognitive Science**, Vol 8 pp 254-273.

Liu, M.R., (1995) "The Effects Of Normalisation On Data Error Finding: An Experimental Evaluation", Unpublished Masters Thesis, University Of Queensland.

Maciaszek, L.A., (1990) **Database Design And Implementation**, Prentice-Hall, Sydney.

Mackay, J.M. & Elam, J. J., (1992) "A Comparative Study Of How Experts And Novices Use A Decision Aid To Solve Problems In Complex Knowledge Domains", **Information Systems Research**, Vol 3 pp 150-172.

Rockart, J.F., & Flannery, L.S., (1983) "The Management Of End-User Computing", **Communications Of The ACM**, Vol 26 (10) pp 776-784.

Schroder, H., & Suedfeld, P., (1971) **Personality Theory And Information Processing**, Ronald Press, New York.

Schwab, D.P., & Cummings, P.L., (1973) "Theories Of Performance And Satisfaction: A Review", **Readings In Organizational Behaviour And Performance**, W.E. Scott, And L.L. Cummings (Eds.), Richard D Irwin Inc. Homewood, pp 130-153.

Shasta, D.E. (1992) **Database Tuning A Principled Approach**. Prentice Hall, Englewood Cliffs, New Jersey.

Soloway, E., Lochhead, J., & Clement, J., (1982) "Does Computer Programming Enhance Solving Ability? Some Positive Evidence On Algebra Word Problems", In R. Seidel, R., Anderson, B., Eds, **Computer Literacy,** Academic Press, New York.

Smith, H.T., & Green, T.R.G., Eds., (1980) **Human Interaction With Computers**, Academic Press, London.

Tsai A.Y.H. (1988) **Database Systems Management And Use**, Prentice-Hall, Scarborough, Canada.

Wood, R.E., (1986) "Task Complexity: Definition Of The Construct", **Organisational Behaviour And Human Processes**, Vol 37 pp 60-82.

APPENDIX A

Description of scales used in the study

Variable	*Scale items and response formats*	*No. of items and source*
UIS (non-atomicity)	Querying *terms of payment* data was: 1 = extremely difficult; to 7 = extremely easy. Circle the number that you feel is most appropriate.	1 item; developed by the researchers.
UIS (repeating fields)	• Querying issue date data was: • Querying quantity issued data was: • Querying quantity defective data was: • Querying production variance data was: • Querying received date data was: • Querying received quantity data was: • Querying unit cost data was: 1 = extremely difficult; to 7 = extremely easy. Circle the number that you feel is most appropriate.	7 items; developed by the researchers.
UIS (fragmentation)	• The database contains too *many* tables. • The database contains too *few* tables. 1 = strongly disagree; to 7 = strongly agree. Circle the number that you feel is most appropriate.	2 items; developed by the researchers.
UIS (overall)	The *Overall* quality of the data structure was: 1 = very poorly designed; to 7 = very well designed. *Overall* querying the database was: 1 = extremely difficult; to 7 = extremely easy. 1 = extremely inefficient; to 7 = extremely efficient. Circle the number that you feel is most appropriate.	3 items; developed by the researchers.

APPENDIX B

Description of data structures used in the study

Non-Normalised (¬NF)

Abbreviation	*Type*	*Desciption*	*Comments*
ITEMA Table			
itemno	Char(6)	Item number of item	
vno	Char(5)	Vendor number	
vname	Char(30)	Name of the vendor	
idesc	Char(30)	Description of item	
terms	Char(5)	Terms of payment	(Violates atomicity -Violates 1NF)
ctqtyoh	Number	Current quantity on hand	
prodjno1	Char(5)	Production job number1	(Repeating fields - Violates 1NF)
issdate1	Date	Date the item1 was issued	
qtyiss1	Number	Quantity of the item issued	
qtydef1	Number	Quantity of the defective items1	
prodvar1	Number	Production time variance1. E.g. 1.10 indicates production required 110% of standard time, i.e., a 10% unfavorable variance.	
prodjno2	Char(5)	Production job number2	
issdate2	Date	Date the item2 was issued	
qtyiss2	Number	Quantity of the item issued	
qtydef2	Number	Quantity of the defective items2	
prodvar2	Number	Production time variance2.	
prodjno3	Char(5)	Production job number3	
issdate3	Date	Date the item3 was issued	
qtyiss3	Number	Quantity of the item3 issued	
qtydef3	Number	Quantity of the defective items3	
prodvar3	Number	Production time variance3.	
recrepno1	Char(6)	No of the first report	(Repeating fields - Violates 1NF)
recdate1	Date	Date shipment1 received	
qtyrec1	Number	Quantity of item1 received	
unitcost1	Number	Cost per unit1	
paydate1	Date	Date payment1 made	(corresponds to receiving report)
payamt1	Number	The amount1 paid	
recrepno2	Char(6)	No of the second report	
recdate2	Date	Date shipment2 received	
qtyrec2	Number	Quantity of item2 received	
unitcost2	Number	Cost per unit2	
paydate2	Date	Date payment2 made	(corresponds to receiving report)
payamt2	Number	The amount2 paid	

First Normal Form (1NF)

Abbreviation	*Type*	*Desciption*
ITEMB Table		
itemno	Char(6)	Item number of item
vno	Char(5)	Vendor number
vname	Char(30)	Name of the vendor
idesc	Char(30)	Description of item
termdisc	Number	Terms of payment - discount percent
termdays	Number	Terms of payment - number of days discount applies
cqtyoh	Number	Current quantity on hand
ISSUESB Table		
prodjno	Char(5)	Production job number
itemno	Char(6)	Item number of item
issdate	Date	Date the item was issued
qtyiss	Number	Quantity of the item issued
qtydef	Number	Quantity of the defective items
prodvar	Number	Effect on production time, i.e., production time variance. For example 1.10 indicates production required 110% of standard time, i.e., a 10% unfavorable variance.
RECEIPTSB Table		
recrepno	Char(6)	The number of the receiving report
itemno	Char(6)	Item Number of Item
recdate	Date	Date shipment received
qtyrec	Number	Quantity received
unitcost	Number	Cost per unit
paydate	Date	Date payment made (corresponds to the receiving report)
payamt	Number	The amount paid

Third Normal Form (3NF)

Abbreviation	*Type*	*Desciption*
ITEMC Table		
*itemno	Char(6)	Item number of item
idesc	Char(30)	Description of item
vno +	Char(5)	Vendor number
cqtyoh	Number	Current quantity on hand
VENDORC Table		
*vno	Char(5)	Vendor number
vname	Char(30)	Vendor name
termdisc	Number	Terms of payment - discount percent
termdays	Number	Terms of payment - number of days discount applies
ISSUESC Table		
*{ prodjno } +	Char(5)	Production job number
{ itemno + }	Char(6)	Item number of item
{ issdate	Date	Date the item was issued
qtyiss	Number	Quantity of the item issued
qtydef	Number	Quantity of the defective items
PRODUCTIONC Table		
*{ prodjno	Char(5)	Production job number
{itemno +	Char(6)	Item number of item
prodvar	Number	Effect on production time, i.e., production time variance. For example 1.10 indicates production required 110% of standard time, i.e., a 10% unfavorable variance.
RECEIPTSC Table		
*{ recrepno } +	Char(6)	The number of the receiving report
{ itemno + }	Char(6)	Item number of item
{ recdate	Date	Date shipment received
qtyrec	Number	Quantity received
unitcost	Number	Cost per unit
PAYMENTSC Table		
*{ recrepno	Char(6)	The number of the receiving report
{ itemno +	Char(6)	Item number of item
paydate	Date	Date payment made (corresponds to the receiving report)
payamt	Number	The amount paid

* - primary key for the relation
+ - foreigh key for the relation

APPENDIX C

Instructions given to participants for ¬NF, 1NF and 3NF

Some of the purposes of this exercise are:

1. To test how well you can use SQL to search for data errors within a database.
2. Allow you to relate input control to specific data errors by searching for data errors that input controls should have prevented.

Task Overview

Task 1 - Read the Scenario and Chris's Questions
Task 2 - Record your session

Part 1 - Answer Chris's Stated Questions

Task 3 - Locate errors and problems Chris (your boss) head of the internal audit department, specifically asked you to look for. (40 mins.)
Task 4 - Report records that contained errors or problems Chris specifically asked you to look for.(10 mins.)

Part 2 - Use Your Ingenuity

Task 5 - Use your ingenuity to locate errors and problems not specifically requested by Chris (40 mins.)
Task 6 - Report records that contained errors or problems not specifically requested by Chris (10 mins.)
Task 7 - Transmit log files
Task 8 - Complete the survey

Scenario

Background

Comfortable Furniture Limited manufactures household and office furniture for distribution throughout the world. The company operates from 9:00am to 5:00pm, Monday through to Saturday. Comfortable Furniture adopted the Just In Time (JIT) II method a little over a year ago for their inventories. JIT II involves the use of contractual agreements between an organisation and its suppliers where the suppliers assume direct responsibility for entire categories of the organisation's inventory. Vendors: (1) provide the required items on a just-in-time basis for production schedules; (2) provide these items at favourable, if not preferential prices; and (3) over the long term, make innovations in their products, production, and pricing to better match the organisation's requirements. In JIT II situations, vendor representatives often occupy offices in the organisation's facilities. The organisation grants the vendor's representatives access to the organisation's data and freedom to inspect physical inventory. The vendor representatives, rather than personnel in the organisation's purchasing department, place the orders for the input materials needed for the organisation's production runs.

JIT II reduces ordering costs, delivery times, handling costs, and inventory holding expenses. For these systems to yield the intended benefits, suppliers must not abuse their direct ordering capabilities. Converting to JIT II means many of the internal controls for traditional purchasing procedures are eliminated. Management expects the internal auditors to query the information systems to determine that JIT II vendor relationships meet internal control objectives.

Task
On 2 July 1995, Chris Kaniuk, the manager of the internal audit group (your boss), calls you into her office. She wants you to scrutinise the JIT II database using SQL SELECT queries to locate any data errors before Deer, Price, and Persnickety, the external auditors, begin their investigation of the system. Furthermore, Ian McMurdy, president of Comfortable Furniture, asked Chris to look for problems or potential problems with the JIT II system. He particularly wants to know about potential overpayments to vendors and whether the overpayments arose from errors or irregularities. Because you must investigate the data anyway, Chris just delegated you this task as well. After completing your investigation, prepare a memo (in bullet format) to Chris pointing out the errors and problems you found in the JIT II database.

Part 1 - *Answer Chris's stated questions*.

1. Items with no item description
2. Negative quantities issued for items
3. Production variances less than 0.8 or greater than 1.5.
4. (a) Term discount rates greater than 5%.
   (b) Term discount days greater than 45 days.
5. Same item description for different item numbers
6. Items that have current quantity on hand less than the average quantity issued.
7. Items with unit prices that have differences greater than 50% between different receipts.
8. Discount [or possibly even an incorrect discount %] taken after the discount period
9. Production variance greater than 1.1 *and* quantity defective = 0
10. Payment date less than receipt date plus half the number of term days
11. Items that have total receipts greater than 10 times the current quantity on hand.
12. Vendor with the highest percentage of defective items

Part 2 - Use Your Ingenuity.

Chris realises that more errors and problems may exist. She would like you to identify as many of the errors as possible in your investigation. The questions asked by Chris in Part 1 should suggest other errors and problems. Your knowledge of input controls should help you generate some questions. Your knowledge of business and your common sense should suggest additional questions.

APPENDIX D

Sample Queries

QUESTION	¬NF Query	1NF Query	3NF Query
Items with no item description	SELECT DISTINCT *itemno* FROM ITEMA WHERE *idesc*=' ' OR *idesc* IS NULL;	SELECT *itemno* FROM ITEMB WHERE *idesc*=' ' OR *idesc* IS NULL;	SELECT *itemno* FROM ITEMC WHERE *idesc*=' ' OR *idesc* IS NULL;
Negative quantities issued for items	SELECT *itemno, issdate1, qtyiss1, issdate2, qtyiss2, issdate3, qtyiss3* FROM ITEMA WHERE *qtyiss1* < 0 OR *qtyiss2* < 0 OR *qtyiss3* < 0;	SELECT *itemno, issdate, qtyiss* FROM ISSUESB WHERE *qtyiss* < 0;	SELECT *itemno, issuedate, qtyiss* FROM ISSUESC WHERE *qtyiss* < 0;
Production variances less than 0.8 OR greater than 1.5	SELECT *itemno, prodjno1, prodvar1, prodjno2, prodvar2, prodjno3, prodvar3* FROM ITEMA WHERE (*prodvar1* > 1.5 OR *prodvar1* < 0.8) OR (*prodvar2* > 1.5 OR *prodvar2* < 0.8) OR (*prodvar3* > 1.5 OR *prodvar3* < 0.8);	SELECT *itemno, prodjno, prodvar* FROM ISSUESB WHERE *prodvar* > 1.5 OR *prodvar* < 0.8;	SELECT *itemno, prodjno, prodvar* FROM PRODUCTIONC WHERE *prodvar* > 1.5 OR *prodvar* < 0.8;
(a) Term discount rates greater than 5%. (b) Term discount days greater than 45 days.	SELECT DISTINCT *vno, vname, terms* FROM ITEMA WHERE TO_NUMBER (SUBSTR(*terms*,1,INSTR(*terms*,'/')-1))>5 OR TO_NUMBER (SUBSTR(*terms*,3,INSTR(*terms*,'/')+2))>45	SELECT *vno* FROM ITEMB WHERE *termdisc* > 5 OR *termdays* > 45;	SELECT *vno* FROM VENDORC WHERE *termdisc* > 5 OR *termdays* > 45;
Same item description for different item numbers	SELECT DISTINCT A.*itemno*, A.*idesc* FROM ITEMA A, ITEMA B WHERE A.*itemno* <> B.*itemno* AND A.*idesc* = B.*idesc*;	SELECT A.*itemno*, A.*idesc* FROM ITEMB A, ITEMB B WHERE A.*itemno* <> B.*itemno* AND A.*idesc* = B.*idesc*;	SELECT A.*itemno*, A.*idesc* FROM ITEMC A, ITEMC B WHERE A.*itemno* <> B.*itemno* AND A.*idesc* = B.*idesc*;

QUESTION	¬NF Query	1NF Query	3NF Query
Items that have current quantity on hand less than the average quantity issued.	SELECT DISTINCT A.itemno FROM ITEMA A WHERE A.cqtyoh < (SELECT AVG ((B.qtyiss1+B.qtyiss2+B.qtyiss3)/3) FROM ITEMA B WHERE B.itemno=A.itemno GROUP BY B.itemno);	SELECT ITEMB.itemno,AVG(qtyiss) FROM ITEMB, ISSUESB WHERE ITEMB.itemno=ISSUESB.itemno AND cqtyoh < (SELECT AVG(qtyiss) FROM ISSUESB WHERE ISSUESB.itemno=ITEMB.itemno) GROUP BY ITEMB.itemno;	SELECT ITEMC.itemno,AVG(qtyiss) FROM ITEMC, ISSUESC WHERE ITEMC.itemno=ISSUESC.itemno AND cqtyoh < (SELECT AVG(qtyiss) FROM ISSUESC WHERE ISSUESC.itemno=ITEMC.itemno) GROUP BY ITEMC.itemno;
Discount [or possibly even an incorrect discount %] taken after the discount period	SELECT itemno, recrepno1, recdate1, paydate1, payamt1, recrepno2, recdate2, paydate2, payamt2 FROM ITEMA WHERE(paydate1 >recdate1 +(TO_NUMBER (SUBSTR(terms,3,INSTR(terms,'/')+2))) AND (payamt1 < unitcost1*qtyrec1) OR (paydate2 > recdate2 + (TO_NUMBER (SUBSTR(terms,3,INSTR(terms,'/')+2))) AND (payamt2 < unitcost2 * qtyrec2);	SELECT RECEIPTSB.itemno,RECEIPTSB.recrepno, RECEIPTSB.recdate,RECEIPTSB.paydate, RECEIPTSB.payamt FROM RECEIPTSB, ITEMB WHERE ITEMB.itemno=RECEIPTSB.itemno AND paydate > recdate + termdays AND payamt < unitcost*qtyrec;	SELECT RECEIPTSC.itemno, RECEIPTSC.recrepno RECEIPTSC.recdate, PAYMENTSC.paydate, PAYMENTSC.payamt FROM RECEIPTSC, VENDORC, PAYMENTSC, ITEMC WHERE ITEMC.itemno=RECEIPTSC.itemno AND RECEIPTSC.recrepno=PAYMENTSC.recrepno AND itemc.vno=VENDORC.vno AND paydate > recdate + termdays AND payamt < unitcost*qtyrec;
Production variance greater than 1.1 *and* quantity defective = 0	SELECT DISTINCT itemno FROM ITEMA WHERE ((prodvar1 >1.1) AND (qtydef1 =0)) OR ((prodvar3 > 1.1) AND (qtydef3 = 0)) OR ((prodvar2 > 1.1) AND (qtydef2 =0));	SELECT ITEMB.itemno FROM ITEMB, ISSUESB WHERE ITEMB.itemno=ISSUESB.itemno AND prodvar > 1.1 AND qtydef = 0;	SELECT ISSUESC.itemno FROM ISSUESC, PRODUCTIONC WHERE ISSUESC.prodjobno=PRODUCTIONC.prodjno AND prodvar > 1.1 AND qtydef = 0;
Payment date less than receipt date plus half the number of term days	SELECT itemno, recrepno1, recrepno2 FROM ITEMA WHERE paydate1 < (recdate1+(0.5*(TO_NUMBER (SUBSTR(terms,3,INSTR(terms,'/')+2))))) OR paydate2 < (recdate2+(0.5*(TO_NUMBER (SUBSTR(terms,3,INSTR(terms,'/')+2)))));	SELECT ITEMB.itemno, recrepno FROM ITEMB, RECEIPTSB WHERE ITEMB.itemno=RECEIPTSB.itemno AND paydate < (recdate+(0.5*termdays));	SELECT ITEMC.itemno, RECEIPTSC.recrepno FROM ITEMC, VENDORC, RECEIPTSC, PAYMENTSC WHERE ITEMC.vno=VENDORC.vno AND RECEIPTSC.itemno=ITEMC.itemno AND PAYMENTSC.recrepno=RECEIPTSC.recrepno AND paydate < (recdate+(0.5*termdays));

QUESTION	¬NF Query	1NF Query	3NF Query
Vendor with the highest percentage of defective items	SELECT *itemno*, (SUM(*qtydef1*+*qtydef2*+*qtydef3*)/ SUM(*qtyiss1*+*qtyiss2*+*qtyiss3*)) FROM ITEMA GROUP BY *itemno* HAVING (SUM(*qtydef1*+*qtydef2*+*qtydef3*)/ SUM(*qtyiss1*+*qtyiss2*+*qtyiss3*)) = (SELECT MAX (SUM(*qtydef1*+*qtydef2*+*qtydef3*) /SUM(*qtyiss1*+*qtyiss2*+*qtyiss3*)) FROM ITEMA GROUP BY *itemno*);	SELECT *itemno*, (SUM(*qtydef*)/SUM(*qtyiss*)) FROM ISSUESB GROUP BY *itemno* HAVING (SUM(*qtydef*)/SUM(*qtyiss*)) = (SELECT MAX(SUM(*qtydef*)/SUM(*qtyiss*)) FROM ISSUESB GROUP BY *itemno*);	SELECT *itemno*, (SUM(*qtydef*)/SUM(*qtyiss*)) FROM ISSUESC GROUP BY *itemno* HAVING (SUM(*qtydef*)/SUM(*qtyiss*)) = (SELECT MAX(SUM(*qtydef*)/SUM(*qtyiss*)) FROM ISSUESC GROUP BY *itemno*);

NOTE: Questions 7 AND 11 were not attempted by any participants