Analyzing a Potential Warranty Call Center Budget Overrun: Using Database Queries to Solve Business Problems

A. Faye Borthick  
Georgia State University

Donald R. Jones  
Texas Tech University

ABSTRACT: In this case, students develop and use database queries to analyze possible causes of a potential budget overrun for telephone warranty service for a company that sells personal computers (PCs). Students query a Microsoft Access database with tables for sales of PCs, estimates for technician time, employees, technician costs, and service minutes provided. The queries illustrate those that accountants could prepare to analyze transaction-level data to develop insights about business operations. Query-based approaches to analyzing transaction data are likely to become more common as businesses take advantage of the wealth of data available to them for solving business problems and exploiting emerging opportunities. This case is appropriate for students with rudimentary database querying proficiency, e.g., at the level developed by Borthick et al.'s (2001) case on assuring compliance for responses to website referrals.

Keywords: budget overrun; call center; database querying; query strategy; queries.

I. THE CASE

The Situation

The Business

A personal computer (PC) company assembles and sells PCs. It accepts orders by phone, through the mail, and on its website, one machine per order. The company generally fills these orders within one week, using a delivery company to ship the computers. It keeps information about machines sold in table tSales. A few rows of the table are shown.
Telephone Warranty Service

The company provides warranty service by telephone for each PC it ships. Based on past service costs, the company estimates the average number of minutes of this service that will be required for each computer over all warranty calls during its warranty period, as shown in table tEstimatedTime. EstWarMin contains the average total number of minutes of estimated service time by TechLevel for each machine with the ProductID. The estimates are based on averages because many machines have no warranty calls while some machines have lengthy calls.

<table>
<thead>
<tr>
<th>ProductID</th>
<th>TechLevel</th>
<th>EstWarMin</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>10</td>
</tr>
</tbody>
</table>

Technician levels by employee are kept in table tEmployee, some rows of which are shown below. For the period covered by the tServiceProvided table, technicians have stayed at the same level.

<table>
<thead>
<tr>
<th>EmployeeID</th>
<th>NameLast</th>
<th>NameFirst</th>
<th>TechLevel</th>
<th>HireDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>55</td>
<td>Gorbataya</td>
<td>Natia</td>
<td>2</td>
<td>3/10/2004</td>
</tr>
<tr>
<td>56</td>
<td>Penthievre</td>
<td>Michelle</td>
<td>1</td>
<td>3/12/2004</td>
</tr>
</tbody>
</table>

Technician cost by minute is available in table tTechCost, which is shown in its entirety below.
Analyzing a Potential Warranty Call Center Budget Overrun

### tTechCost Table (primary key = TechLevel)

<table>
<thead>
<tr>
<th>TechLevel</th>
<th>MinuteCost</th>
<th>LevelDesc</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$1.00</td>
<td>Staff Technician</td>
</tr>
<tr>
<td>2</td>
<td>$2.00</td>
<td>Senior Technician</td>
</tr>
<tr>
<td>3</td>
<td>$3.00</td>
<td>Technician Manager</td>
</tr>
</tbody>
</table>

Estimated technician expense is defined as the sum over all machines sold of tTechCost.MinuteCost times tEstimatedTime.EstWarMin by TechLevel.

The number of minutes of service provided is recorded in table tServiceProvided, a few rows of which appear below. The table contains all the warranty calls for the machines in the tSales table.

### tServiceProvided Table (primary key = MachineID, DateService, and StartTime)

<table>
<thead>
<tr>
<th>MachineID</th>
<th>DateService</th>
<th>StartTime</th>
<th>EmployeeID</th>
<th>Minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>101289</td>
<td>1/3/2005</td>
<td>10:53</td>
<td>56</td>
<td>17</td>
</tr>
<tr>
<td>101289</td>
<td>1/8/2005</td>
<td>14:56</td>
<td>62</td>
<td>15</td>
</tr>
<tr>
<td>147312</td>
<td>1/3/2005</td>
<td>9:45</td>
<td>56</td>
<td>10</td>
</tr>
<tr>
<td>155862</td>
<td>3/18/2005</td>
<td>12:49</td>
<td>56</td>
<td>15</td>
</tr>
<tr>
<td>155862</td>
<td>3/19/2005</td>
<td>11:20</td>
<td>62</td>
<td>20</td>
</tr>
</tbody>
</table>

Total actual technician expense is defined as the sum over all machines sold of tTechCost.MinuteCost times tServiceProvided.Minutes by TechLevel.

### The Situation

The warranty service manager suspects that actual expense, comprised of overhead expense and technician expense, for providing telephone warranty service is exceeding budgeted expense. Staying within budget is a big deal for the manager because the company is evaluating how much of the call center’s operation to move to a lower-wage country.

### Required

#### Part 1: Querying to Investigate a Potential Budget Overrun

Respond to the warranty service manager’s suspicion by querying the database. As you consider the situation of the manager of call center warranty operations, think through the following questions:

1. What general risks are associated with call center warranty operations?
2. What specific risks are associated with call center warranty operations?
3. Of the risks identified in 2, which ones can be analyzed with the information in the database?
4. Of the risks identified in 2, which ones cannot be analyzed with the information in the database?
5. How can you use the risks to guide querying?
6. With respect to minimizing the analysis effort, does the order in which you address the risks/questions matter? If so, how?
7. What questions should an analysis of the database address? For example:
   a. Is there a budget overrun overall?
   b. Are there differences by product?
c. Are there differences by product by technician level?
d. Are there differences by product by technician level by technician?

**Part 2: Objective Questions**

**Required.** Select the best response for each question based on the business situation and database. The questions are independent of each other.

1. A query to determine the name of the technician with the longest call requires the following table(s):
   a. tEmployee, tTechCost
   b. tEmployee, tServiceProvided
   c. tEmployee, tEstimatedTime
   d. tEmployee, tServiceProvided, tTechCost

2. A query to determine, for each technician level, which technicians had the longest calls requires the following table(s):
   a. tEmployee, tTechCost
   b. tEmployee, tEstimatedTime
   c. tEmployee, tServiceProvided
   d. tServiceProvided

3. A query to show total call time for each product requires the following tables:
   a. tEmployee, tEstimatedTime
   b. tSales, tEstimatedTime
   c. tEmployee, tSales, tServiceProvided
   d. tSales, tServiceProvided

4. A query sequence to show, for each product, the average call time for machines sold requires the following operations:
   a. (1) sum of call time by product and (2) number of calls by product
   b. (1) sum of call time by product and (2) number of machines by product
   c. (1) sum of estimated warranty time by product and (2) number of calls by product
   d. (1) sum of estimated warranty time by product and (2) number of machines by product

5. To avoid unnecessary waste of effort, the project team’s most efficient starting point would be to identify whether:
   a. the newest technicians are spending more time on calls than budgeted
   b. some products are requiring more call time than the time budgeted
   c. actual minutes exceed budgeted minutes for some technician levels
   d. actual technician expense exceeds budgeted technician expense

6. In a lunchtime conversation, a group of the technicians revealed to a team member that most of their calling time seemed to be due to one or two products. To follow up on this observation, the team should investigate which combination of relationships:
   1. frequency of calls by product
   2. average time by product by technician level
   3. average time per call by product
   4. average time by technician by product
   a. 1 and 2
   b. 1 and 3
   c. 2 and 4
   d. 3 and 4
7. Suppose the query qEstMinByMachine appears in design view as:

<table>
<thead>
<tr>
<th>Field</th>
<th>MachineID</th>
<th>EstWarMin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>tSales</td>
<td>tEstimatedTime</td>
</tr>
<tr>
<td>Total:</td>
<td>Group By</td>
<td>Sum</td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To create a query that shows the total number of estimated minutes for all machines, the best approach is to:

a. do nothing—the query already shows the total number of estimated minutes for all machines
b. create a new query, based on qEstMinByMachine, having the design view:

<table>
<thead>
<tr>
<th>Field</th>
<th>MachineID</th>
<th>EstWarMin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>qEstMinByMachine</td>
<td>qEstMinByMachine</td>
</tr>
<tr>
<td>Total:</td>
<td>Sum</td>
<td></td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To create a new query, based on qEstMinByMachine, having the design view:

<table>
<thead>
<tr>
<th>Field</th>
<th>SumofEstWarMin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>qEstMinByMachine</td>
</tr>
<tr>
<td>Total:</td>
<td>Sum</td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td></td>
</tr>
</tbody>
</table>

To create a new query, based on qEstMinByMachine, having the design view:

<table>
<thead>
<tr>
<th>Field</th>
<th>MachineID</th>
<th>EstWarMin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>qEstMinByMachine</td>
<td>qEstMinByMachine</td>
</tr>
<tr>
<td>Total:</td>
<td>Group By</td>
<td>Sum</td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. A query to find the dollar cost of the warranty service minutes provided requires access to which tables:
a. tEmployee, tSales, and tTechCost  
b. tSales, tServiceProvided, and tTechCost  
c. tEmployee, tServiceProvided, and tTechCost  
d. tServiceProvided and tTechCost

9. A query to find the total dollar cost of the warranty service minutes provided would perform the following actions:
   a. for each call, multiply minutes provided by the cost per minute and sum the costs of calls  
   b. multiply the total number of minutes provided by the cost per minute  
   c. for each technical level, multiply the minutes provided by the cost per minute of that technical level  
   d. multiply the total number of minutes for each machine by the cost per minute

10. Suppose the total number of minutes provided is less than the total number of minutes estimated, but the total dollar cost of the minutes provided is greater than the budgeted cost of estimated minutes. This outcome is most likely to be a function of:
   a. in at least one of the higher technical levels, more minutes being provided than budgeted  
   b. in at least one of the higher technical levels, fewer minutes being provided than budgeted  
   c. technicians being less competent than anticipated at their level  
   d. technicians being more competent than anticipated at their level

11. Suppose the total number of minutes provided exceeds the total number of minutes estimated and that the total dollar cost of the minutes provided exceeds the budgeted cost of estimated minutes. This outcome indicates the need to investigate the time and cost differences for:
   a. each product, each technical level, and each technician  
   b. each product and each technical level  
   c. each technical level and each technician  
   d. each product and each technician

12. Suppose the purpose of the following query, shown in design view, is to find the total number of minutes for each product. Tables tSales and tServiceProvided were joined on MachineID.

<table>
<thead>
<tr>
<th>Field</th>
<th>ProductID</th>
<th>Minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>tSales</td>
<td>tServiceProvided</td>
</tr>
<tr>
<td>Total:</td>
<td>Group By</td>
<td>Sum</td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td>✚</td>
<td>✚</td>
</tr>
<tr>
<td>Criteria:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This query result would:
   a. fail to show ProductIDs for which no minutes were provided  
   b. fail to associate the correct ProductID with minutes provided  
   c. match all rows in tSales with all rows in tServiceProvided  
   d. show the total for all products rather than by product
13. Suppose the objective is to find the average number of minutes provided per machine sold for each product, and the following queries and results are available:
   - Query1: Sum of all minutes provided, grouped by product
   - Query2: Count of machines, grouped by product
   - Query3: Count of service calls, grouped by product
   
   The best approach would be to create a query that uses:
   a. Query1 and Query2, dividing the sum of minutes by the count of machines for each product
   b. Query1 and Query3, dividing the sum of minutes by the count of service calls for each product
   
14. The purpose of the following query, shown in design view, is to sum all minutes provided by product.

<table>
<thead>
<tr>
<th>Field</th>
<th>MachineID</th>
<th>ProductID</th>
<th>Minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>tServiceProvided</td>
<td>tSales</td>
<td>tServiceProvided</td>
</tr>
<tr>
<td>Total:</td>
<td>Group By</td>
<td>Group By</td>
<td>Sum</td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td></td>
<td>❑</td>
<td>❑</td>
</tr>
<tr>
<td>Criteria:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   The result of this query would have:
   a. the right number of rows
   b. more rows than are needed
   c. fewer rows than are needed
   d. the right number of rows but the wrong values

15. Suppose the technicians of TechLevel = 1 have become proficient at handling easy questions and have also become so good at handling harder questions that they hardly ever shift calls to TechLevels 2 and 3. Nonetheless, because they are not as experienced they take about 50 percent longer than higher technician levels for hard questions. Assuming all products’ estimated warranty minutes were fairly accurate, this situation would most likely lead to the outcome of:
   a. total minutes provided being less than estimated and cost for service less than budgeted
   b. total minutes provided being more than estimated and cost for service more than budgeted
   c. total minutes provided being less than estimated but cost for service more than budgeted
   d. total minutes provided being more than estimated but cost for service less than budgeted

16. Suppose that among technicians of the same technical level the average length of calls is significantly different while the number of calls is about the same for each technician. This result indicates the need to investigate:
a. individual technician performance and technician performance by product
b. individual technician performance
c. technician performance by product
d. technician performance by level

17. Making tServiceProvided and tTechCost available to a new query and inserting the fields TechLevel, MachineID, and Minutes results in a query with the following number of rows:
   a. number of rows of tServices Provided
   b. number of rows of tTechCost
   c. number of rows of tServiceProvided plus number of rows of tTechCost
   d. number of rows of tServiceProvided times the number of rows of tTechCost

18. Consider a query for which the tSales table and the tServiceProvided table have been made available. In the calculated expression “Sum([tServiceProvided]. [Minutes]),” the term “[tServiceProvided]”:
   a. is necessary
   b. is unnecessary
   c. causes a syntax error
   d. causes multiple joins

19. Obtaining information about which of the following outcomes is likely to be beyond the scope of querying this database?
   a. whether some products require more service minutes than budgeted
   b. whether some technicians are less effective in providing service than others
   c. whether some technician training programs are more effective than others
   d. whether some products at some technical levels require more service than others

20. The query that counts the number of machines by product for which the company is responsible for telephone warranty service is:
   a. with tSales and tServiceProvided available:

<table>
<thead>
<tr>
<th>Field</th>
<th>ProductID</th>
<th>ProductID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>tServiceProvided</td>
<td>tServiceProvided</td>
</tr>
<tr>
<td>Total:</td>
<td>Group By</td>
<td>Count</td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Criteria:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   b. with tSales available:

<table>
<thead>
<tr>
<th>Field</th>
<th>ProductID</th>
<th>MachineID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table:</td>
<td>tSales</td>
<td>tSales</td>
</tr>
<tr>
<td>Total:</td>
<td>Group By</td>
<td>Count</td>
</tr>
<tr>
<td>Sort:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Show:</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Criteria:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Analyzing a Potential Warranty Call Center Budget Overrun

II. LEARNING OBJECTIVES

Querying to Solve Business Problems

Query-based approaches to analyzing transaction data are likely to become more common as businesses take advantage of the wealth of data available to them for solving business problems and exploiting new opportunities. An accountant querying a database is an example of a professional employee trying to make sense of the information in large databases of transaction and other data that organizations maintain (Weick 1995, 2001). Making sense of what is available is a form of organizational learning with the potential to enable responses to new situations as they arise (Grant 1996a, 1996b; Brown and Duguid 1998; Orlikowski 2002). The intent of this kind of learning experience is to prepare students “for the ambiguous business world they will encounter upon graduation” (Albrecht and Sack 2000, 43). To enable this objective, the case immerses learners in an ambiguous situation characterized by incomplete, multifaceted data.

The general domain of the case, telephone call center operations, has become an essential aspect of the service to product users for many companies. Call centers have been studied from a situated change perspective (Orlikowski 1996) and have been caught up in political discussions about the implications of shifting jobs to lower-wage areas of the world (Hilsenrath 2004). The specific context for the case, assessing the extent of a possible budget overrun for the call center warranty service for a company that makes PCs, is enabled through a database with tables for sales of PCs, estimates for technician time, employees, technician costs, and service minutes provided. Students analyze transaction-level data to develop insights about business operations.

Learning Objectives

The objectives for this case are for students to learn to (1) identify risks in a business situation, (2) decide which risks can be investigated with transactional data in an operational
database, (3) query the database to determine the extent to which the risks were realized, and (4) characterize the risks for which the database lacks relevant information. The case operationalizes learning objectives in the information use category (McKinnon and Bruns 1992; Borthick 1996) applied to a business situation for which management wants information of the kind that accountants might be asked to produce by querying a database (Borthick 1992; Speier and Morris 2003).

Prerequisite Skills

The case is appropriate for students with rudimentary database querying proficiency. The prerequisite query skills include joining tables; building expressions; using built-in functions; applying Group By; formatting and sorting results; and naming, saving, and retrieving queries. A case that develops rudimentary query skills is Borthick et al.’s (2001) case on assuring compliance for responses to website referrals. Table 1 compares this case and the one in Borthick et al. (2001).

Developing the queries to solve the business problem does not require prior database design training or experience. Attributes are named consistently across tables, which makes it possible to decide which tables to join (and how) based on their meaning in the situation. With the situation and data structured this way, learners are afforded the engaging experience of using the data in a database to investigate a business problem without having spent substantial time mastering database theory. Then, when they take up the study of database design theory, learners will have already experienced some aspects of database design from the practice of database querying. From this perspective, database querying motivates the need for sound database design approaches.

Case Design Rationale

The case context was designed with the least number of tables and attributes necessary to represent an ad hoc querying situation like those that arise in organizations. Because the tables are in third-normal form (3NF), learners are faced with deciding which tables to join for different queries and when to perform the joins relative to performing calculations such as summing and averaging estimated and provided service minutes. Unlike Borthick et al.’s (2001) web referral case, the tables in this case have too much data for hand verification of results to be feasible.

In Part 1 (querying to investigate a potential budget overrun), learners (1) use default and specified table linking to make the information accessible, (2) create mathematical expressions to manipulate attribute values in rows, (3) format and sort result columns, (4) apply functions to columns, and (5) combine query results to permit performance comparisons. The querying outcomes identify some specific budget overrun conditions. In Part 2 (objective assessment), learners respond to multiple-choice questions whose learning objectives appear in Table 2.

Student Feedback on Learning

While they seem to need varying levels of assistance to prepare a cogent analysis, students have uniformly found the case to be both challenging and engaging. It is challenging because it prompts students to apply the querying skills they learned from Borthick et al. (2001) in a new context, and it admits multiple interpretations that need to be resolved. It is engaging because it features authentic work of the kind many organizations are undertaking and because the interplay between using the information at hand and imagining more helpful information is inherently interesting. Another benefit of the case is that it enables a newfound identity for students as business problem solvers.

Journal of Information Systems, Spring 2005
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Web Referral Case Borthick et al. (2001)</th>
<th>Warranty Call Center Case</th>
</tr>
</thead>
</table>
| Learning objectives for business problem solving | 1. Decide what information would be relevant to solving a business problem  
2. Extract the needed information  
3. Put the information in a form conducive to solving the problem  
4. Analyze the information to solve the problem.                                                       | 1. Identify risks in a business situation  
2. Decide which risks can be investigated with transactional data in an operational database  
3. Query the database to determine the extent to which the risks were realized, and characterize the risks for which the database lacks relevant information |
| Learning objectives for query proficiency       | Recognize the need for and perform the following query operations:  
1. Join tables; select attributes  
2. Build expressions  
3. Use built-in functions  
4. Apply Group By  
5. Format and sort results  
6. Name, save, and retrieve queries                                                                 | 1. Increase competence and independence in querying databases                                                                 |
| Learner scaffolding possibilities       | Questions about:  
1. The business problem  
2. Query strategy  
Full-text solution for:  
1. The business problem  
2. Query strategy  
3. Querying by keystroke                                                                 | Questions about:  
1. The business problem  
2. Query strategy  
Full-text solution for:  
1. The business problem  
2. Query strategy                                                                 |
| Prerequisite: Query proficiency       | None                                                                                                      | Ability to recognize the need for and to perform the following query operations:  
1. Join tables; select attributes  
2. Build expressions  
3. Use built-in functions  
4. Apply Group By  
5. Format and sort results  
6. Name, save, and retrieve queries                                                                 |
| Data volume | Minimal: Small enough to permit hand verification of results                                                                 | Moderate: Large enough to preclude hand verification of results |
| Objective assessment | 10 multiple-choice questions                                                                 | 20 multiple-choice questions |
| Course suitability | Any undergraduate or graduate course with an objective of developing rudimentary query skills in which students:  
1. Have no query experience or  
2. Need to refresh query proficiency                                                                 | Any undergraduate or graduate course with an objective of increasing query competence and independence in which students:  
1. Have rudimentary query skills                                                                 |
| Time | 2–6 hours depending on existing skills                                                                 | 2–6 hours depending on existing skills                                                                 |
TABLE 2
Objective Assessment Learning Objectives

<table>
<thead>
<tr>
<th>Learning Objective</th>
<th>Question</th>
<th>Query Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Locate relevant information in tables</td>
<td>1, 2, 3, 8</td>
<td></td>
</tr>
<tr>
<td>Sequence individual queries</td>
<td>4, 7, 9, 13</td>
<td>✓</td>
</tr>
<tr>
<td>Select first query objective</td>
<td>5</td>
<td>✓</td>
</tr>
<tr>
<td>Investigate common wisdom about situation</td>
<td>6</td>
<td>✓</td>
</tr>
<tr>
<td>Interpret query results</td>
<td>10, 11, 15, 16</td>
<td>✓</td>
</tr>
<tr>
<td>Debug queries</td>
<td>12, 14</td>
<td>✓</td>
</tr>
<tr>
<td>Construct queries</td>
<td>17, 18, 20</td>
<td>✓</td>
</tr>
<tr>
<td>Decide relevancy of database</td>
<td>19</td>
<td>✓</td>
</tr>
</tbody>
</table>

III. IMPLEMENTATION GUIDANCE

Matching the Case to Courses

The case has been used in an undergraduate accounting information systems (AIS) course to assess the querying competence that students developed with another querying case (Borthick et al. 2001) and in a master’s level information systems (IS) assurance course to ensure sufficient competence for querying databases in IS audit situations (Borthick and Jones 2000). The case could be used in other accounting and business courses featuring business processes or assurance services in which an objective is to learn to use database querying to solve business problems or exploit emerging opportunities.

Calibrating Course Requirements with Learner Proficiencies

The case is usable by learners with rudimentary proficiency in formulating query strategies and in using a database query interface. The prerequisite query skills include joining tables; building expressions; using built-in functions; applying Group By; formatting and sorting results; and naming, saving, and retrieving queries. Rudimentary query skills can be developed with Borthick et al.’s (2001) web referral case. Learners without this level of proficiency are likely to feel frustrated. Beyond the minimum proficiency level, instructors can adjust the assistance they provide learners with respect to query strategy. For example, instructors wanting their students to have the experience of developing more of the query strategy could withhold the strategies implied by the four sub-questions in Part 1.7 that signal the need to investigate the specific aspects of (1) whether a budget overrun exists (1.7.a.), (2) differences by product (1.7.b.), (3) differences by product by technician level (1.7.c.), and (4) differences by product by technician level by technician (1.7.d.). To make the case even more challenging, instructors could truncate the requirements in Part 1 to just “Respond to the warranty service manager’s suspicion by querying the database.”

Helping Students Thrive with Ambiguity

Regardless of initial learner proficiency in formulating query strategies, it may be helpful for students to begin by participating in a discussion of possible query strategies. For learners struggling to develop a query strategy, it may be helpful for them to participate in another discussion of query strategy after they have developed some queries. The discussion could focus on (1) how to interpret the meaning of existing query results, (2) how to use existing query results as a means of informing further querying, (3) how to restructure their
queries to yield better overall results, and (4) how to determine when one has sufficient query results given the problem and the available data. If students seem determined to analyze technician behavior independent of product attributes, a usually helpful remedy is to prompt students to think of the business process to identify the fundamental drivers of the need for warranty service, which refocuses the querying on product as the top-level entity rather than technician.

To deter students from thinking that obtaining query results is synonymous with generating a complete list of possible causes for the business problem, it is helpful for students to participate in a discussion of potential causes for the results that could not be investigated through the database. To emphasize the importance of considering a wide range of possible causes, instructors can prompt students to think of the implications of acting prematurely on the wrong cause, e.g., applying significant resources that do not remedy the problem.

Potential Extensions

The business situation makes explicit the assumption that a sales order corresponds to an order for a single machine. Instructors that want students to be aware of how to accommodate multi-item orders could require them to redesign the database to accommodate multi-machine orders. Doing so requires splitting the tSales table into a sales table with summary data about the order and a line-item table with data about each line on the order.

The student materials do not include an E-R diagram or other graphical representation of database schema because it would not exist in many ad hoc query situations like this one. There might be an E-R diagram for the whole database from which data are extracted, but its complexity would likely deter use. Instructors who want their students to use an E-R diagram or other graphical representation could ask them to construct one. Because attributes are consistently named, constructing a graphical representation is straightforward.

Realizing Learning Gains through Collaboration

The case can be assigned to individual learners or to teams of learners. In our experience, teams usually exhibit the advantages of more thorough analyses, less frustration on the part of individual learners, and higher satisfaction with the learning experience due to its collaborative nature, which enables students to learn from each other (Rogoff 1998; Borthick et al. 2003). Working with others also lets students experience the demand for business professionals to collaborate on problems for which new approaches or solutions are required (Schrage 1990; Raelin 1997).

Collaboration creates opportunities for group members to confront each other with alternatives. For example, although they might readily agree on the desirability of investigating whether there really is a budget overrun, group members are unlikely to agree initially on the best way to go about it or what to do after they get a result to that question. In the process of confronting each other with different views, learners may realize there are different ways to develop the queries, which has the potential to develop their ability to think critically about problems (Bruffee 1999). In essence, students learn “to discriminate better between facts and conclusions, to draw fewer false conclusions, to consider more than one solution to a problem, and to be less adversely influenced in their approach to a problem by their experience of a preceding one” (Abercrombie 1960, 18).

Although this case has a particular business problem for students to solve, the larger objective is for students to develop query skills that will allow them to develop solution approaches to more challenging problems. The objective is for learners to internalize reasoning strategies that enable them to develop effective approaches to the next new problem...
they encounter (Kozulin 1998; Sfard 1998). Some of these long-lived skills appear to develop in the activity of tearing apart inferior solutions (Kruger 1993). Because they tend to implement the first solution approach that occurs to them, individuals, on their own, are unlikely to generate many opportunities to learn from inferior solutions. For this purpose, collaborating with peers is helpful because peers think of different approaches and push each other to justify their beliefs or abandon them (Abercrombie 1960).

A risk of team completion of Part 1 is freeloading by individuals. To obtain the benefits of team work on the case while minimizing the occurrence of freeloading, we have assigned Part 1’s querying requirements to teams, and afterward given Part 2’s objective questions as an in-class assessment for course credit. Another tactic to deter freeloading is to require student teams to publish reports of their analyses and databases with queries supporting them on the web or in a learning management system 24 hours before a class discussion about analysis results. The visibility of published work is often sufficient impetus to prompt student teams to make good faith efforts to complete the case analysis.

IV. TEACHING NOTES

The Teaching Notes for this case include:

1. The text of:
   a. Part 1 analysis of querying
   b. Part 2 solution for objective questions with question feedback

2. A URL for a zip file containing implementation files for the case:
   a. Access mdb file containing the data for student querying
   b. Access mdb file containing the data and QBE queries for instructor use that supports the part 1 analysis of querying

3. Files containing the part 2 multiple choice questions and question feedback:
   a. Word doc file
   b. WeblCT format txt file

4. HTML file of the case text to enable instructors to stage the case on a website. Any websites used for this purpose should be password-protected, and the passwords should be given only to students enrolled in courses using the case.

V. SUMMARY

This case enables students to experience database querying as a way to investigate a business problem with transaction-level data. Learners develop database queries for assessing the extent of a possible budget overrun for the call center warranty service for a company that makes PCs. The queries are developed in Microsoft Access®, but any database system supporting QBE or SQL could be used. Query-based approaches to analyzing transaction data are likely to become more common as businesses take advantage of the wealth of data available to them for solving business problems and exploiting emerging opportunities. This case is appropriate for students that have attained rudimentary database querying proficiency.

Teaching Notes are available to full-member subscribers to the Journal of Information Systems through the American Accounting Association’s electronic publication system at http://aaahq.org/ic/browse.htm. Full-member subscribers should use their usernames and passwords for entry into the system where the Teaching Notes are available.

If you are a full member of AAA with a subscription to the Journal of Information Systems and have any trouble accessing this material, please contact the AAA headquarters office at office@aaahq.org or (941) 921-7747.
REFERENCES


