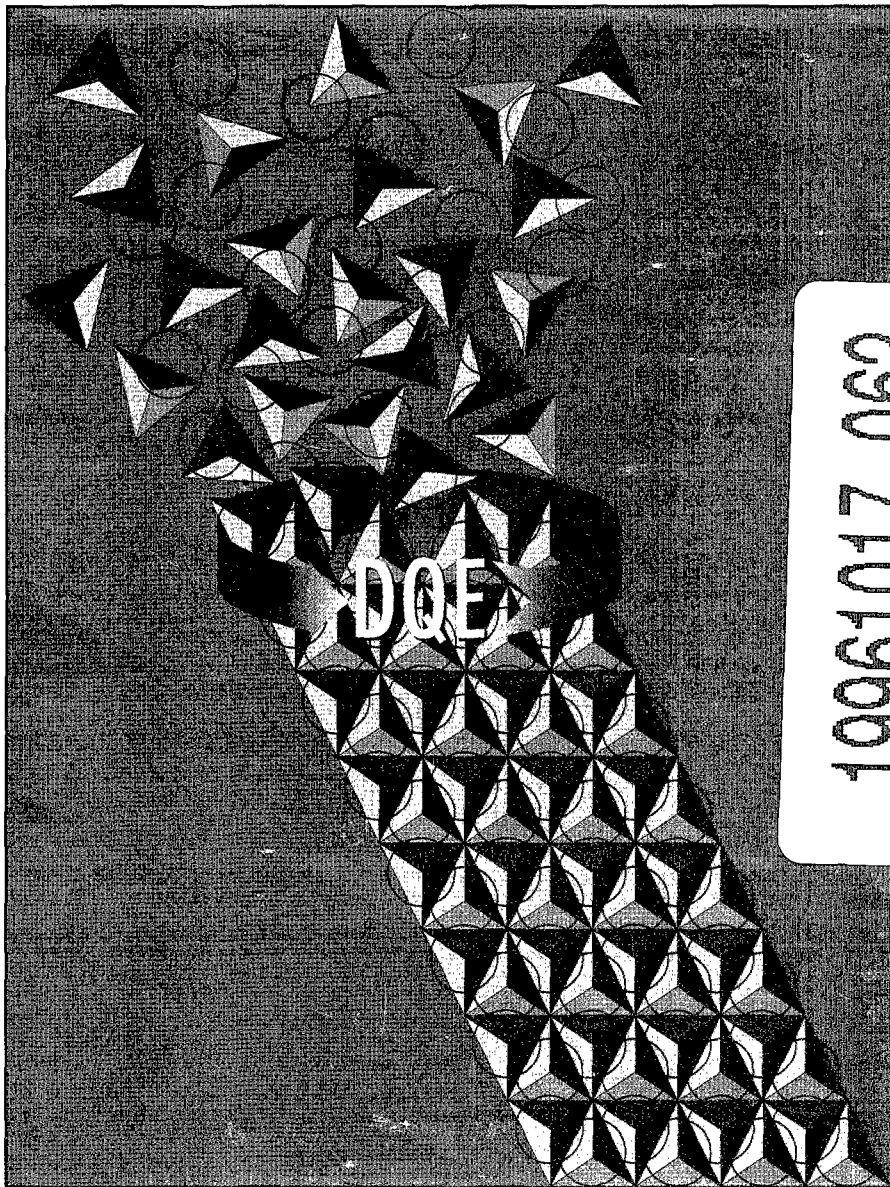


Data Quality Engineering Handbook



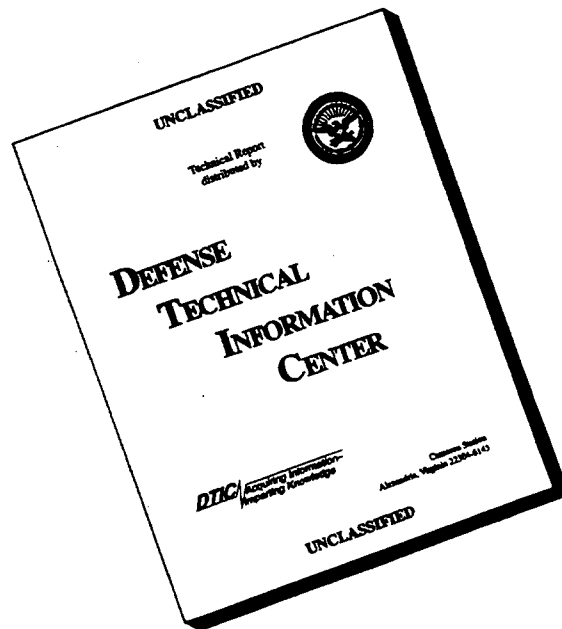
19961017 062

Defense Logistics Agency

RESTRICTION STATEMENT A

Approved for public release
Distribution Unlimited

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.



DEFENSE LOGISTICS AGENCY
HEADQUARTERS
8725 JOHN J. KINGMAN ROAD, SUITE 2533
FT. BELVOIR, VIRGINIA 22060-6221



IN REPLY
REFER TO

CANE

28 JUN 94

PREFACE

In this period of accelerated technological advancements and information proliferation, we are all producers and consumers of large quantities of data. As we define, create, and use data, we need to be concerned about the importance of having good data and being able to share it across functions and applications. The DLA Data Quality Engineering Handbook has been prepared by my staff to explain the basic principles of good data management, how to identify common sources of errors, and how to find help to fix systemic problems. Comments and suggestions for improvement are welcome from all of you who use and help manage the large volumes of information resources that provide the underpinnings for our support to the warfighter.

Thomas J. Knapp
Chief Information Officer
Defense Logistics Agency

DTIC QUALITY INSPECTED 3

Federal Recycling Program Printed on Recycled Paper

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited





Table of Contents

	Page
Introduction	vii
 Chapter 1—Developing a Plan	
Story Line	1
Introduction	5
Key Concepts	5
Task 1—Review the Team Charter	7
Task 2—Develop and Display the Team's Findings	11
Task 3—Consult with the Process Owner	13
Summary	15
Check on Learning	15
 Chapter 2—Preparing the Team	
Story Line	17
Introduction	21
Key Concepts	21
Task 1—Identify the Characteristics of the Problem Area	23
Task 2—Analyze the Problem Area	26
Task 3—Gather and Review Documentation	30
Summary	32
Check on Learning	33
 Chapter 3—Creating Metadata	
Story Line	35
Introduction	39
Key Concepts	39
Task 1—Define Metadata and Begin a Working Data Dictionary	41
Task 2—Analyze Data Element Names and Definitions	47
Task 3—Determine Data Element Domains and Lengths	50
Task 4—Determine the Format and Observed Range	55
Summary	60
Check on Learning	61
Practical Exercises	61
 Chapter 4—Building Metadata-based Business Rules	
Story Line	63
Introduction	67
Key Concepts	67



	Page
Task 1—Build a Domain-based Business Rule	69
Task 2—Build a Format-based Business Rule	77
Task 3—Build a Business Rule Based on Observed Range	82
Summary	89
Check on Learning	89
Practical Exercises	90

Chapter 5—Capturing Discovery Business Rules

Story Line	91
Introduction	95
Key Concepts	95
Task 1—Discover Policy/Procedure Business Rules	97
Task 2—Discover “If...Then...” Business Rules	100
Task 3—Discover “Data-Generating” Business Rules	103
Summary	108
Check on Learning	109
Practical Exercises	109

Chapter 6—Applying Business Rules to a Large Database

Story Line	111
Introduction	113
Key Concepts	113
Task 1—Obtain Database Records	115
Task 2—Apply the Business Rules	120
Task 3—Organize and Report the Results	124
Summary	127
Check on Learning	128

Chapter 7—Working with Multiple Database Systems

Story Line	129
Introduction	133
Key Concepts	134
Task 1—Develop a Data Flow Diagram	135
Task 2—Establish Data Linkages	138
Task 3—Repeat Chapter 6	142
Summary	145
Check on Learning	146

Chapter 8—Finding and Eliminating Root Causes

Story Line	147
Introduction	151



	Page
Key Concepts	151
Task 1—Establish Data Error Categories	153
Task 2—Develop an Action Plan	156
Task 3—Reevaluate Data Quality	161
Summary	163
Check on Learning	166
Practical Exercises	166
Appendix A - Glossary	A-1
Appendix B - Program Team Charter	B-1
Appendix C - Check on Learning Responses	C-1
Appendix D - Practice Data Value Samples	D-1
Appendix E - Answers to Practical Exercises	E-1
Appendix F - U.S. State Code Abbreviations	F-1
Appendix G - Date Format Business Rule List	G-1
Index	xv
Credits and Contacts	xix



List of Figures

	Page
Figure 1-1. Team Observations	8
Figure 1-2. Team Member Strengths	9
Figure 1-3. Partially Completed Storyboard	12
Figure 2-1. Military Identification Card	26
Figure 2-2. Data Element Diagrams	29
Figure 2-3. Data Value Sample	31
Figure 3-1. A Data Dictionary 5"x7" Card	42
Figure 3-2. Data Element Names and Aliases	44
Figure 3-3. Data Sample	44
Figure 3-4. Data Definition Guidelines	46
Figure 3-5. Data Element Definition Comparison	49
Figure 3-6. Domain Examples	51
Figure 3-7. The SOCIAL SECURITY NUMBER 5"x7" Card	53
Figure 4-1. The Business Rule List	70
Figure 4-2. Data Value Sample	71
Figure 4-3. The Business Rule List (Continued)	73
Figure 4-4. The Data Value Error List	74
Figure 4-5. Highlighted Errors in State Codes	76
Figure 4-6. The Business Rule List (Continued)	78
Figure 4-7. ZIP CODE Data Value Sample	79
Figure 4-8. ZIP CODE Data Value Errors	81
Figure 4-9. BODY WEIGHT Data Value Sample	82
Figure 4-10. Standard Deviation Examples	84
Figure 4-11. Business Rule List (Continued)	86
Figure 4-12. BODY WEIGHT Data Sample with Highlighted Errors	87
Figure 4-13. BODY WEIGHT Suspected Error List	88
Figure 5-1. Data Value Sample	97
Figure 5-2. BODY WEIGHT Data Sample	101
Figure 5-3. Military Equipment Dimensions	104
Figure 5-4. Calculated Data Values	106
Figure 6-1. Possible Record Selection Criteria	118
Figure 6-2. Data Value Report Log	119
Figure 6-3. Business Rule List	121
Figure 6-4. Data Quality Metric Table	125
Figure 6-5. Data Quality Baseline	125
Figure 7-1. Data Sources	135
Figure 7-2. Data Flow Diagram	137
Figure 7-3. ITEM VOLUME Data Dictionary Card	139
Figure 7-5. Data Linkages Report	140
Figure 7-4. CUBE Data Dictionary Card	140
Figure 7-6. The "Vertical Slice"	144
Figure 8-1. Error Category List	155
Figure 8-2. Probable Root Causes	159
Figure 8-3. Data Quality Measurements	162



Introduction

Who Will Use the Handbook?

This Data Quality Engineering (DQE) Handbook describes the procedures adopted by the Defense Logistics Agency (DLA) for improving or restoring the quality of the data in corporate information systems. The handbook is written for team leaders and their team of functional “users” of data and not necessarily a data administrator, database administrator, or other specially trained individual. The handbook is intended to be used by the person who creates or uses data and who has a vested interest in the reliability, validity, and accuracy of the data values stored in DLA database systems.

What is a Legacy System?


The handbook is intended to be used to improve data quality in existing, or in what are commonly called “legacy,” systems. For the purposes of the handbook, legacy systems are defined as those automated, mainframe-based, database systems developed before the existence of standard procedures for either automated system design or database implementation. Typically, legacy systems are the automated system support “workhorses” of organizations like DLA. However, in spite of their importance and utility, legacy systems often lack complete or current documentation. Frequently, legacy system program code is convoluted by change after undocumented change.

The word “stovepipe” has been used to describe the way in which legacy systems were developed. Though this term correctly implies that legacy systems were developed independently, now the data in legacy systems is routinely shared and passed from system to system. Again, however, the lack of a consistently applied standard and a parallel lack of current system documentation combine to obscure data sources, complicate data quality control efforts, and, in general, frustrate the functional user.

How is the Handbook Organized?

The handbook is organized into eight chapters. The chapters each contain three or more tasks. Each task is further broken down into three or more steps. The steps describe the detailed series of actions that, taken together, form the Data Quality Engineering methodology. The text includes an example with most steps. Notes are also provided with many steps to provide additional information or detail.

A novel-like narrative, called a “story line,” begins and ends each chapter. The story line serves as a transition between chapters. However, its more important role is to introduce key concepts in a sometimes lighthearted, always easy-reading format. The story line is about four imaginary DLA employees. John is earning a college degree by taking classes in the evening. At the beginning of the story line, John starts a data quality class. Because he suspects the data quality at his DLA work site is poor, he talks his mentor, Professor Hopkins, into helping him apply his new knowledge at work. Maureen, Joe, and Janine are DLA employees and members of John’s Total Quality Management (TQM) team called the “Data Quality Engineers.” (All the characters in the story line are fictitious. Any similarity between these fictional characters and a DLA employee or college professor is purely coincidental.) An italic font is used to distinguish the story line from the main body of the handbook.

Following the story line, each chapter contains a short introduction and a list of key concepts. A lightning bolt  in the margin indicates that a key concept is being introduced. Each chapter ends with a summary and a “Check on Learning” section. The Check on Learning questions are intended to help you ensure that you understood the key concepts introduced in the chapter. Some chapters also contain a practical exercise section designed to allow you to apply what you have learned.

The handbook includes seven appendixes. Appendix A contains a list of the acronyms and a glossary of terms used throughout the handbook. Appendix B contains a sample Total Quality Management Team Charter. Appendix C contains responses to the Check on Learning questions. Appendix D contains a

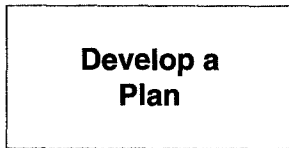


practice data value sample (with embedded errors) for use with the practical exercise questions. Appendix E contains a list of data errors (from the sample in appendix D). Appendix F contains a list of approved U.S. state code abbreviations. Finally, appendix G contains a list of business rules that could be applied to certain calendar date data.

The final section is an index to help readers quickly find a term or concept in the handbook text. Also, to aid readers, chapter numbers are printed on tabs on the edge of each odd-numbered page. Readers can find a certain chapter quickly by bending the handbook and looking for the appropriate tab on the edge opposite the handbook binding.

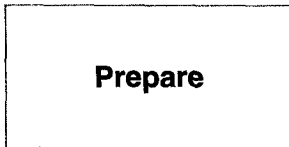
What is Data Quality Engineering?

Data Quality Engineering is a proven process for restoring the validity, accuracy, and reliability of the data maintained in information systems. The DQE methodology uses TQM principles. The methodology consists of the following key activities:



Develop a Plan

A “process owner” initiates the first step in DQE (and TQM) by writing a charter and making an initial team member selection. In chapter 1, the team reviews its charter, assesses its capabilities, and validates its mission. Normally, this step finishes with a joint process owner/team meeting.



Prepare

In chapter 2, the team defines its customer and the problem area. The team breaks the problem area into its component parts, into its subcomponents (if necessary), and then, working to ever greater levels of detail, arrives eventually at the data element level. Once at this level, the team reviews any available documentation on the targeted automated system. The team also works to obtain a good assortment of reports containing sample data values.

**Develop
Metadata**



Next, the team works with its data samples to understand exactly how the data is defined and used within its target system. The information that the team gathers in chapter 3 is called “metadata.” Metadata, really just “data on the data,” includes items like the “data element name,” “definition,” “range,” and others. By capturing and recording metadata, the team begins to understand the meaning and significance of the data values stored in the target system.

**Develop Metadata-
based Business
Rules**



In chapter 4, the team begins to check the quality of the values in its data samples. The team checks data values by building and implementing business rules. A business rule is simply a statement of fact about data. The team uses metadata as the basis for its first set of business rules. For example, if the team determines that a data element has a specific range (“range” is one of the metadata elements), then it can create a metadata-based business rule that flags or identifies all the data values in the sample that do not fall within an acceptable range.

**Develop
“Discovery”
Business Rules**



Business rules are applicable to more than just metadata! In chapter 5, the team learns how to convert written policy and procedure into business rules. A relationship among the data values in a given data set can also be used as a basis for a business rule. Also in this chapter, the team learns that the data values present in a given data value sample can sometimes be used to generate a correct data value in cases where a value is missing or wrong.

**Examine
Full/Large
Database Files**



Up to this point, the team has been developing and learning to use metadata and business rules by working with relatively small data value samples. In chapter 6, the team learns how to work with all or a significantly large portion of

data. This chapter covers techniques for generating, handling, and checking the data quality in a large database report. The number of data values that pass all business rules form a data quality baseline. This measure of data quality, element by element, allows the process owner to establish a clear understanding as to where problems exist, to set priorities, and, if necessary, to manage limited resources. Further, the DQE methodology supports a measurement of data quality (at some later time), performed using exactly the same process as had been used for the baseline. The second and all subsequent measurements allow the process owner to gauge success in achieving data quality.

**Examine
Multiple
Databases**



Chapter 7 describes how the DQE methodology applies to more than one system. As mentioned earlier, data in today's legacy systems is often provided to or received from another (legacy) system. The DQE methodology includes techniques to determine a data source and to restore data quality controls, if necessary, at the source.

**Conduct
Root Cause
Analyses**




The final chapter describes a technique for determining the root cause of data value errors. Typically, errors are caused by defective policy or procedure, poor or missing system documentation, inadequate training, errors in automated processing routines, or, finally, data entry error. The DQE methodology describes techniques for capturing, recording, and then resolving error causes at their source.

Is Data Quality Engineering an Automated Process?

The Data Quality Engineering methodology lends itself to automated support. The handbook, however, was created initially as though no automated support was available. This technique was used to ensure that the reader will be able to understand the process completely.

There is a second benefit to creating the handbook as though no automated support was available. By using this technique, the methodology is not tied to any specific software or hardware product. The methodology can be tailored to use the latest and most innovative automated support techniques known to the user any time a DQE process is initiated.

The fact is, however, that without automated support, the actual execution of the DQE methodology could require an immense commitment of time and resources. Further, the quantity of information used and also generated throughout the methodology could easily overwhelm a team. Therefore, at various points in the methodology, the handbook contains suggestions (shown in the text as  TIP) as to why and how automated tools can be applied to improve consistency and reliability of analyses, reduce the workload, or streamline the record-keeping requirements.

What are the Benefits of Using the DQE Methodology?

Data Quality Engineering results in an improvement in the validity, accuracy, and reliability of the data values stored in a database. It is a process that is repeatable and quantifiable. The improvements that can be attributed to the application of the DQE methodology include

- Fewer data value errors, greater consistency in the definition and use of data, less redundancy, and improved access to data



-
- Clearer definition of data requirements
 - Fewer complaints from functional users, particularly from those who have experienced persistent data error problems
 - Reduced operating costs/improved efficiency.

The DQE methodology applies to a wide variety of data types. It can be used to improve data quality in a single, isolated system, but the most significant improvements occur when DQE is applied to two or more loosely integrated systems. DQE is particularly effective in supporting enterprise integration efforts (developing and implementing plans to move from many independently operated systems to fewer migration systems and, eventually, to a small number of target systems).

Where Can I Find DOD Data Administration Policy?

The Department of Defense (DOD) has an active, ongoing data administration effort, as well as written policy. To the extent possible, the procedures in the handbook comply with published data administration policy and procedures, as defined in the Department of Defense *Data Administration Procedures Manual* (DOD 8320.1-M) and related instructions. In some cases, however, the handbook reflects a concept or a procedure not recognized in the DOD 8320 series. In all cases, these deviations arise from the fact that legacy systems, the subject of DQE, contain design and other elements that predate the current DOD policy. On the few occasions when the handbook introduces a concept or procedure not recognized by DOD, the exception to DOD policy is noted in the text.

.



Chapter 1

Developing a Plan

John, oblivious to all around him, barged past the Department's secretary and opened Professor Hopkins' door.

"I've got nine people to help me do the DQE!" he announced excitedly. John plopped down into the tall, green armchair in front of the professor's desk, quite pleased with the opportunity to improve the data quality at his work site using the techniques he is learning in Professor Hopkins' class.

"Well, John," said the professor. "You certainly didn't waste any time! You wouldn't by any chance be trying to beat the grade submission for the quarter, would you?" Professor Hopkins managed to successfully stifle his laugh, but John noted the twinkle in his eyes.

"Hey, I'm not opposed to doing anything to help you help me—not to mention my DLA depot!" laughed John.

"Good!" responded the professor. "Now let's see who you have to help you."

John handed the list of names to the professor.

"Why these individuals, John?" asked the professor.

"Well, sir, you see," began John. "We're all in the same DLA unit—Joe, he's a data administrator; Darick is new, but he seems to be a great worker; Matt, he's just a friend of mine, and..."

"John," interrupted the professor, "rule number one in DQE is to have the right individuals assembled. I'm not sure this group, however 'great' they may be, represents the range of talent you will need to succeed in a Data Quality Engineering process. Do any of these friends of yours control, or are they affected by, the data that is to be studied?"

"I just thought you needed a few good people! I don't know any real computer experts, but Joe probably has the most to do directly with the data system."



"All you need from the selected members are three attributes: (1) knowledge of the process, (2) a commitment to see the DQE task to its completion, and (3) openness to others' ideas. They don't have to be experts, because the team can always consult with 'the experts.' However, team members do need to represent a particular interest in the process."

"Well, there's Maureen—she handles the data entries from Requisitioning; and there's Colleen who handles the data quality assurance program; and of course, Janine just arrived in our depot, but has a lot of experience in tracking material release orders, and..."

"Sounds like you already have a better idea as to how to pick your team. Don't get more than about seven folks, though. Experience has taught me 'that more than nine and you'll get behind!' Too many people will have you spending more time managing them than the project; on the other hand, too few will result in too little investigation," said the professor. "Now, who's asking you to take on this project?"

"Colonel Mathis," John said. "He's my boss."

"For role clarity sake, let's call Colonel Mathis the 'process owner.' The process owner is the individual who knows and manages the overall process. Colonel Mathis, as the process owner, will draw up a charter to describe the problem as he sees it and request the team to resolve it."

"But if he understands the problem already, why does he need us?" said John.

"He probably doesn't know what causes the problem, John. At this point he may suspect a cause, but he doesn't know for sure. Colonel Mathis will appoint representatives from different parts of the process to determine what causes the problem, how to correct these problem sources, and in the end, decide how to improve the overall system," said the professor.

"You know, this is beginning to sound like TQM to me," said John. "I thought we had finished with that topic last semester!"

"If you think you finished with TQM last semester," said the professor, "then I probably need to reinforce TQM principles even more than I had planned before the close of this semester! John, remember that TQM is a way of managing your business processes. That's why Data Quality Engineering employs TQM—to ensure we continuously improve our process and that we always work with the customer's perspective in mind."

"Sorry," offered John, a little upset with himself for challenging the professor. "So far, we have worked to ensure we have the right players on our team, we know our process owner, and we have a good charter. What do we do next?"

"Well, let's start by reviewing some of those TQM techniques we talked about last semester! They will be very useful as we begin the DQE process," said the professor, heading toward his chalkboard.




Chapter 1

Developing a Plan

Introduction

This chapter describes the Total Quality Management (TQM) techniques that the Data Quality Engineering (DQE) team will be using. In the first task, the team reviews the charter that it has been provided. The charter defines the problem, the process owner's expectations, and an initial team composition. The team will work to ensure it understands the task and that it has the right capabilities in the members of the team. In task 2, the team decides how to retrieve, present, and store all relevant problem-related data and information. In the third task, the team learns how to communicate with the process owner productively. Though largely administrative in nature, these preparatory techniques will enable team members to move quickly and purposefully through the DQE procedures.

Key Concepts

-  Charter
-  Storyboard
-  Process owner

Task One

Review the Team Charter



A charter is a contractual agreement between a manager (process owner) and a team. It is typically written by the process owner and given to the team leader. The charter states a problem, formally establishes the process owner's objectives, defines an initial team composition, and finally, outlines the process owner's expectations for the team. The charter also describes issues or factors that bear on the problem. Such issues may include resource constraints, lack of data, or the need to avoid interfering with regular operations. In addition, the charter should describe anticipated interim meetings with the process owner and establish a completion date for the effort. These milestones enable the team to pace its activities to meet the expectations of the process owner.

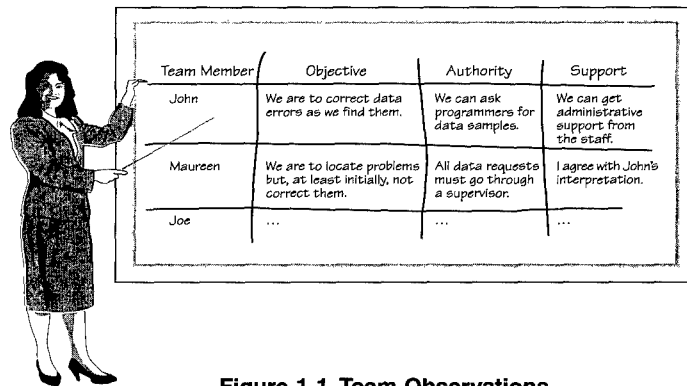
To ensure the team's ability to meet the charter objective, the appointed team leader discusses the charter contents with the rest of the assigned team members. Team members determine whether the charter is comprehensive and explicit enough to support the activities they will be undertaking. If the team determines that the charter falls short of specifying the authority and required resource support it needs, the team leader should arrange to discuss this shortfall with the process owner. As an official document, signed by all involved parties, the charter ultimately empowers the team to do what is necessary, within the noted limitations specified in the document. From time to time, a change in the charter may be required to meet the process owner's or the team's evolving perspective on the problem.

The team should follow a four-step process to ensure it understands the charter.

Step 1 Read the charter

Allow each member to read the charter (appendix B contains a sample charter). Ask each team member to record his or her understanding of the stated or implied meanings of the charter's objectives, team authority, and team support on a white board or flip chart.

Example: The team leader distributes the charter to all team members and calls for an initial meeting. On a white board in a conference room that has been reserved for team use, team members take turns jotting down their understanding of their objective, authority, and their support requirements. Figure 1-1 shows the results of this effort.



Team Member	Objective	Authority	Support
John	We are to correct data errors as we find them.	We can ask programmers for data samples.	We can get administrative support from the staff.
Maureen	We are to locate problems but, at least initially, not correct them.	All data requests must go through a supervisor.	I agree with John's interpretation.
Joe

Figure 1-1. Team Observations

Step 2 Assess the charter contents

Have all the team members introduce themselves and briefly discuss their own understanding of the charter's objective and tasks as displayed on the white board. Where the views expressed on the white board differ, work toward building a consensus. Record the consensus or note where opinions differ.

Step 3 Assess team members' talents and skills

Ask each member to explain his or her interest in the problem and describe any personal strengths that will contribute to the team's effort. Write these ideas down and post them. Review the team member strengths and assess whether a needed area of expertise is missing from the contribution list. If so, take action as specified in the charter to add or change the team composition. Consult with the process owner if additional guidance is needed.

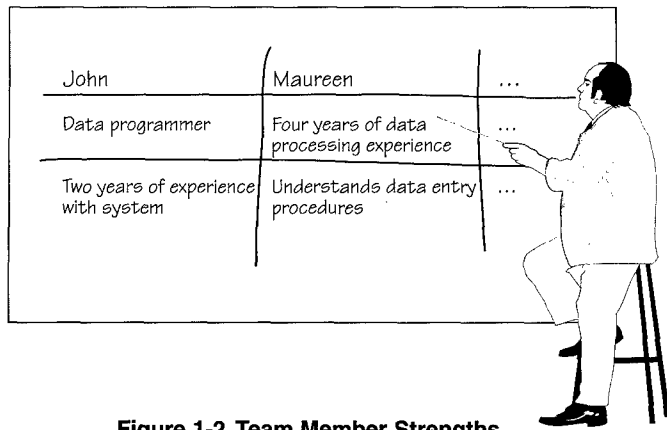


Figure 1-2. Team Member Strengths

Step 4 Build team cohesion and identity

Use team-building exercises to establish and maintain team cohesion (identity) and work on building a consensus for using the DQE methodology (strategy).

Develop a team interpretation of the charter. Present the results to the process owner. Include the team's self-assessment of its ability to accomplish its goals. Include a discussion of issues and preferred solutions. Show a strategy for resolving problems and improving the process. Offer a modified charter if needed. Request the process owner's permission to proceed.

Example: The items marked with an asterisk (*) are recommended activities at each team meeting. This list is not intended to be all-inclusive. A number of other useful TQM team-building activities can be found in current literature on the TQM process.

1. Establish team rules (e.g., “Team members will attend all meetings,” “Criticize ideas—not people,” “Keep an open mind,” “Share responsibility,” “Manage by fact,” “Question and participate,” “Arrive on time”).
2. Agree on team roles (e.g., appoint a recorder, facilitator, team leader, scribe, etc.).
3. Share and record team expectations (e.g., have each member respond to this question: “What do you personally hope to get out of this team effort?”).
4. Build team cohesion using an exercise (e.g., have team members respond to this question on paper, first alone and then with a group of three: “If you were lost in the desert, what 10 tools would bring you to safety?” Compare responses of the individual with the response from the individual working within the context of a group. Ask team members to discuss the difference in outcomes. Which type of response (individual or group) had more productive outcomes? Why? Which type of response is more commonly used? Why? When is one type of response preferred to another? Why?
- 5.* Plan next meeting’s agenda. Decide on a purpose, supporting topics, responsible parties, time limitations, etc.
- 6.* Working as a team, list the pros and cons at the end of each meeting. Ask questions like “What can we do better next time?”, “How can we achieve a more productive outcome?”, “What kinds of things worked at this meeting?”
- 7.* Build and maintain the team strategy and schedule by asking and recording the team response to questions like: “What are the milestones for the team?” “When should each milestone be accomplished and who is responsible?”, “What process will be used to reach each milestone?”

Task Two

Develop and Display the Team's Findings

Throughout the DQE process, the team will be creating reports, lists, and other important documents. To help keep track of its progress, the team should record the purpose, the process used, the results, and any issues relating to each document. These procedures will be used throughout this handbook. They are noted here in general terms. More specific activities will be detailed later.

Step 1 Illustrate key team activities

Using flip charts or other large presentation media, display the purpose, the process used, the results, and the issues related to the documents created or requested by the team.



Note: If possible, maintain the displays on a wall in a dedicated conference room for the duration of the team's activities.

Step 2 Record all team findings



Document all relevant team activities and team findings as they are reported. Maintain these recordings in a storyboard format or in a file that is readily available to the team.

Example: The team elects to use a storyboard technique. It develops a graphic illustration of the steps it will follow to achieve its goal. Figure 1-3 shows the steps the team has outlined, showing some partial entries to date (through step 3).

(Task Two continued)

Step 1: Team Identity	Step 2: Team Objectives
Name: "The Data Quality Engineers" Team Members: John Smithfield, Maureen Adams, Joe Marshall and Janine Wright.	To use a DQE process to determine root causes of problems with data in the personnel information database system
Step 3: Data Process Status	Step 4: Customer Requirements
Customer complaints have risen by 50%. Problems noted with names, birth dates, SSNs ...	
Step 5: Deltas between Customer Requirements and Process Status	Step 6: Internal System Data Element Problem
Step 7: Multisystem Data Element Problems	Step 8: Root Cause Analysis of Data Element Problems
Step 9: Countermeasures to Root Causes	Step 10 : Implement and Measure

Figure 1-3. Partially Completed Storyboard

Step 3 Provide agendas for follow-on meetings

Develop agendas to keep the team focused on what should be accomplished next. Show what part of the story board the planned activities will support. Be sure that individual team members understand their responsibilities. An agenda can be modified, as needed, during a meeting, but the focus should not be altered if the team is to remain on track.

Task Three

Consult with the Process Owner



The process owner is the manager with decision authority over the area in question. The process owner's role is to establish an objective, to select an initial team composition, and to be the team's advocate and supporter. The success of the team depends on the process owner's ability to define a task and to sustain the team by providing the resources needed to complete the task. For success to occur, the process owner and the team require frequent, open, and honest communication.

The team's relationship with the process owner begins with the review of the charter and continues throughout the duration of the project. The team should have periodic meetings with the process owner. For example, if using the storyboard technique, the team should meet with the process owner after each phase. In all discussions with the process owner, the team should be clear and precise in presenting its result and listen closely to feedback. Ask questions to understand the process owner's comments. Keep the process owner informed of all significant developments.

Step 1 **Prepare to brief the process owner**

Choose and publish the issues/objectives for a meeting. Schedule a meeting. Arrange to meet in the team's work area if possible.

Step 2 **Meet with the process owner**

Conduct the meeting by briefing the process owner on the team's issues, findings, and conclusions. Solicit feedback. Make one or more team members responsible for recording the process owner's responses and guidance. Ask questions when necessary to clarify the process owner's comments and suggestions. Schedule or discuss requirements for the next meeting. Close your meeting with a note of appreciation for the process owner's time and support.

Step 3 Document process owner's comments in team log

After the meeting, review the process owner's comments and suggestions with the team to determine if changes in approach are necessary.



Note: Check with team members frequently to be sure they remain focused on the problem and are working cohesively toward their objectives. A team that is not functioning well is usually not focused on the tasks. One common problem is too many changes in the way the team agrees to work. Try to minimize changes in the work process. Routinely give the team a break in the task assignments and an opportunity to identify and resolve team process problems (if they exist).

Summary

This chapter briefly describes the key features of the time-tested TQM process. Although emphasized in this first chapter, the TQM process, in fact, applies to every chapter in the handbook. Each significant step along the Data Quality Engineering path embodies one or more of the TQM characteristics of thorough preparation, open and productive dialogue, focused team activity, and continuous process improvement.

“So, John,” said the professor as he closed his textbook on TQM, “do you feel you remember enough about TQM to support your DQE efforts for DLA?”

“Yeah, it’s all coming back to me now. I hope I can remember what to use when,” laughed John.

“You’ll know, John, I promise,” smiled the professor. “Somehow I suspect this DLA experience is going to be better than any course you could have taken from this institution.”

John nodded. “For some reason, I think you’re right!”

Check on Learning

Responses can be found in appendix C.

1. Who develops the charter and what should the charter contain?
2. What role does the process owner serve on a TQM team?
3. Who decides who should be on a TQM team?

Chapter 2

Preparing the Team



"Now, John," Professor Hopkins began, as he moved from his chalkboard to his desk, "once the team is formed, it will define the effects of the data processing problem. Its first objective will be to identify its customer and to specify the customer requirements."

"But, Professor Hopkins," protested John, "we already know who the customer is!"

"Yes, I know you have encountered the customer and I suspect you might even have established a pet name for them by now!" the professor responded with a smile.

"You've got that right!" said John. "They are always complaining about this or that! Sometimes I think some of them come in to the office just to see how far they can push us! They act like we don't know what we're doing and we're out to make their lives miserable!"

The professor leaned against his desk and took a puff from his pipe. "Well, that may very well be what you are doing, John, but of course, not intentionally! What you want to do in DQE is reconnect the customer to your process—find out what they need and expect, and then measure how well your process is meeting their requirements."

"Does this mean the team has to actually go sit down with them or reinvent our process just to make life easier for them and harder for us?" asked John with a slight hint of discontent.

"John, without the customer, you have no reason to go to work in the morning. Each DLA data process is directly tied—or should be tied—to an individual need. If the individual customers aren't satisfied, then your process is not satisfactory. It's really that simple."

"Now wait a moment!" said John. "Isn't this getting a little out of hand! I can't keep doing everything each customer wants everytime. I've got a job to do! You know, I've got to turn in reports the customer never would even care about,

check on things they wouldn't even dream I had to take care of... And that thing you said about 'whoever receives the product or service' from my process is considered a customer, well do you realize how many folks you're talking about? They are not always outside DLA walls, you know! Some of those guys are people like, well, like the colonel and the unit down the hall!"

"Right!" exclaimed the professor. "The customer can be inside or outside DLA. The point is, John, that regardless of whether the receiver of your products is in the next office or the next building, you have to deliver what is required, in the manner specified. If you can't satisfy the requirement, you are not needed."

"And what about having to meet every little need the customer comes up with?" asked John angrily.

"Well, if you don't, what do you suppose will happen?" responded the professor.

John hesitated and then said, "Okay, okay. They will go elsewhere to get what they want. I know that. It's just tha..."

"It's just that your customers have been so dissatisfied for so long that your people don't have a good feeling about them?" asked the professor.

"Yeah, you could say that I guess," mumbled John. "Okay, okay. You made your point. If I can satisfy the customer, then I can be considered valuable to them, right?"

"No," responded the professor. "If your system or process can satisfy the customer, then it can be valuable to the customer. This is not about people doing wrong, John, for very few want to do wrong things for a customer. This is about a process that somehow failed to meet what the customer wanted. Your team must find out why that happened!"

"Where do we start?" said John with a meekness that caused the professor to laugh out loud.

"Why don't you start with the customer?" said the professor as he settled himself back into a more serious state of mind. "After all, it's the customer who let you know that something needed fixing."

"You make it sound so simple, but I have a gut feeling this is going to open a Pandora's box!" said John.

"Well, it might, John, but the alternative is to let Pandora's box blow up in your face! Let's take the customer's complaints, one at a time, so we can get back on

track with satisfying them, shall we? Any one problem solved is a step closer to doing what we were hired to do, right?"

"Right," said John with a quick nod. "I'll go get the team ready for this. Imagine their faces when I tell them we are going to actually invite the customer to complain!"

"No, John, what you're going to do is ask the customer to help you better meet their needs. They can do this by restating their requirements so you can determine your specific strengths and weaknesses in meeting their needs and expectations. When you can do that, you are on your way toward understanding what part of your data processing system requires analytical attention."

"Gee, this is a real concept," John said trying to hide an emerging grin. "Imagine me talking to the customer!"

"Get out of here!" laughed the professor as he reached for some papers on his desk.

John smiled and rose from his seat in front of the professor. Then with a sudden slap on the professor's desk, he laughed too. "Hey, don't push me!" he said. "I'm your customer!"







Chapter 2

Preparing the Team

Introduction

In this chapter, the team will continue to employ time-tested TQM principles. In the first of three tasks, the team applies a TQM technique for defining the characteristics of the problem. The two key characteristics developed in task 1 are the customer and customer's product. In task 2, the team breaks the problem down. The team works down to a component level, a subcomponent level (if necessary), and continues until it reaches the data element level. Once data elements are identified, the team attempts to identify the source of the data element (input) and the uses for the data element (output). The team captures and records this information in the data element diagrams. Finally, in task 3, the team begins to gather and review pertinent documentation. Perhaps the most important source of information for the team will be tables, reports, completed forms, and other sources of actual data values. Collectively, these tables, reports, forms, and other sources are called "data value samples." The study of data values (or the actual "instances" of data) is a fundamental activity in DQE.

Key Concepts

-  TQM customer
-  Customer's product
-  Problem statement
-  Product component
-  Data element
-  Data element diagram

.....

Task One

Identify the Characteristics of the Problem Area

In this task, the team will develop a problem statement. The team will also define its customer, the customer's product, and the characteristics of the customer's product. This task consists of five steps.



Note: In this task, the examples pertain to a hypothetical problem that military family members have had when they request a new identification (ID) card.

Step 1 Identify the customer



A customer is a person (or group of people) who expects a product or service. In other words, the customer places a demand on a system or process. The customer should be identified in the charter.

Example: The customer is a military service person's family member who is 10 years old or older, primarily dependent on the service person's income, and who needs an ID card.

Step 2 Identify the customer's product



The customer's product is simply the item or the information that the customer expects to receive. To be useful to the team, the customer's product should be a written report, a form, a diagram, or other printed material containing data.

Example: The customer's product is a family member ID card.



Note: For the remainder of this task, the term "customer's product" will be simplified to "product."

Step 3 Identify the required characteristics of the product

Characteristics are the criteria used to decide if the product meets the customer's requirement(s) and expectation(s). The characteristics are always viewed from the customer's perspective. The team may find that characteristics have been defined in the charter. The team, however, may want to add or modify characteristics based on the experiences of team members or on interviews with customers. The length of the list of characteristics will vary. A few characteristics pertaining to the ID card example are given below.

Example:

1. The customer expects the ID card information will be correct.
2. The customer expects the photo to present a good image of him- or herself.
3. The customer expects to receive a card within an hour of completing an application.
4. The customer expects...

Step 4 Relate the problem area to the required characteristics



State, in a simple sentence, the problem issue as described in the charter. A good problem statement will describe an undesirable result that occurs when the customer places a demand on a system or process. The reverse of the undesirable result becomes a quality indicator which is used to measure the quality of the product.

Example: The customer routinely experiences delay in obtaining a family member ID card.

- Quality indicator: no delays in obtaining a family member ID card
- Quality measure for indicator: the presence of delays in obtaining a family member ID

Step 5 Obtain feedback from the customer

At this point, the team will discuss the problem statement, product definition, and product characteristics with the customer. The team will solicit feedback from the customer. As a result of the discussions with the customer, the team may need to refine the problem statement, product definition, or product characteristics.



Note: The team may find that the customer has unrealistic expectations. This is a good time to explain the known limitations of the system or process (and to establish reasonable customer expectations).

Here is an example of a team/customer dialogue:

Example: “Why can’t I get my military ID card when I arrive? Is that too much to ask? I’ve had to go to the Military Personnel Office three times and my card is still not right.”

“Well, Mrs. Adams,” said the team leader, “currently our system depends on two things that must take place the day you come in for your ID. The first is the official photograph, taken according to military specifications. The second is verification of the military family member’s information by the military sponsor. Because these two requirements take time, it is impossible for us to deliver an ID card immediately on your arrival. But certainly having to come back repeatedly is not reasonable either. From what you’ve said, I gather that it is the information part of the ID that is causing the delay. Is this right?”

“Yes. Every time I’ve had to renew my ID, somehow the information on it is wrong. My husband has even taken time off from work, but when we come back to pick up the IDs, the information is still not right. This is very annoying.”

“Let’s see if our office can’t find the problem and fix it. We’re not happy that our system fails to provide you a quality product.”

Task Two

Analyze the Problem Area

The word “analyze” means to separate something into parts and then to examine the parts in detail. In this task, the team breaks the customer’s product into components. The team then divides the components into subcomponents, and continues as necessary until reaching the “data element” level. The ID card problem from task 1 is used again as a source for the examples.

Step 1 Break the product into components



Look for natural divisions in the product layout. Draw a circle around the components or items in a product that seem to share similar features.

Example: Refer to figure 2-1. Note that the front and back of the ID card (the product) contain areas for a DOD emblem, a photograph, blanks for specific kinds of information, and two places that seem to contain encoded information. The components could be defined as the

- Emblem component
- Photograph component
- Information component
- Encoded component.



Figure 2-1. Military Identification Card

Step 2 **Select the component(s) that may relate to the problem area**

Review the problem statement from task 1. Eliminate any component that does not appear to relate to the problem. (Remember that DQE is data oriented.) Components that do not appear to contain data can be eliminated. The team may revisit its choice of components and add or delete components whenever new information on a component develops.

Example: The component(s) that relate to the problem area include

- Information component
- Encoded component.



Note: The emblem component was eliminated because it contained no apparent data and no obvious relationship to the problem area. The photograph component contained no defined data values. The encoded component also contains no readily observable data values, but values may be hidden in specialized codes.

Step 3 **Divide each selected component(s) into data elements**

Break the component(s) into subcomponents if necessary and continue until reaching the data element level.



Note: The formal definition of a data element is “a named identifier of each of the entities and their attributes that are represented in a database” (Reference: *Data Element Standardization Procedures*, DOD 8320.1-M-1). In simpler terms, a data element is the lowest level at which meaningful pieces of information are stored.

Example: The information component of a military ID card (see figure 2-1) contains the data elements:

- Name
- Date of birth
- Social security number
- Sponsor SSN
- Relationship
- Signature
- Hair color
- Height
- ...



Note: The team may be uncertain as to when it reaches the data element level. Because there is no hard and fast rule, the team will have to experiment by attempting to reach the data element level and then proceeding. The team can return to this step and redefine its data if it has difficulty in later steps.

The key is to remember that the data element level is the level at which discrete but uniquely meaningful pieces of information are stored (and usually named). For instance, a data element called AGE could be broken down into its component, single-digit numbers. However, at this level, the meaning and significance of an individual's age is lost.

Step 4 Select the data elements related to the problem area

Review the problem statement from task 1. Compare the problem statement to the list of data elements. Put an asterisk by "suspect data elements" (the ones that seem to be related to the problem statement).



Note: The purpose of this step is to prevent a large number of data elements from diluting the effort to solve the problem. The number of data elements to be considered will vary by problem, but as a rule, the team should use this step to reduce the number of data elements to be considered to 50 or fewer.

At this point, the selection criterion is based as much on the comparison of the data element name to the problem statement as on gut instinct or a feeling.

Step 5 Develop data element diagrams



Create a table with three columns. Label the columns as shown in figure 2-2. List the suspect data element names down the center column, one per row in the table. In the left column, identify the source (input) for the data related to the suspect data element. In the right column, identify the result (output) related to this data element.

Data Element Diagrams		
Source (Input)	Data Element Name	Component (Output)
Applicant	Name	Information Component
Applicant	Date of Birth	Information Component
	Sponsor SSN	Information Component

Figure 2-2. Data Element Diagrams



Note: The team will apply its collective experience and knowledge to complete an initial effort on the data element diagrams. If the team cannot identify the required input or output, leave a blank space. The blank spaces in the table will define the information requirements for the next task.

Task Three

Gather and Review Documentation

In this task, the team will fill in the blank spaces on the data element diagrams. Its search for information begins with a review of documentation that relates to the problem area. It will complement this initial review by interviewing subject matter experts (SMEs). Finally, the team will gather a data value sample. Samples of actual data values play a key role in Data Quality Engineering.

Step 1 Review relevant policy and procedure

Gather written policy and procedure pertaining to the problem area. Look for any information that will help fill in the blanks on the data element diagrams. Begin filling the blanks on the table with the new information. Write a summary paragraph on each document the team reviews and file for later reference.

Step 2 Interview subject matter experts

Discuss the data element diagrams and ask the SMEs to help provide missing information.



Note: SMEs are also good sources for leads on additional documentation. They may also suggest changes (additions or deletions) to the data element list.

Step 3 Obtain data value samples

Request copies of printed reports, printouts, and other automated machine-generated material used routinely by the functional community. Look for samples that list actual data values. Attempt to find at least one sample for each data element in the data element diagrams. Ask the SMEs to search for and highlight data errors in the samples, but do not ask the SME for any explanation at this point. Assign the SMEs a certain

highlighter color. (In this handbook, all SMEs will use gray. Another highlighter color will be needed later.)

Example: A functional user routinely receives a report listing the data used to prepare new ID cards. (See figure 2-3.) The report is organized in columns and rows. Column headings correspond to data element names; in this case, NAME, BIRTH STATE, DATE OF BIRTH, DATE OF ISSUE, and EXPIRATION DATE. Each row in this report forms a record (a series of related data values). SMEs have highlighted two values without explanation.

Data Value Report				
Name	Birth State	DOB	Date of Issue	Expiration Date
Burke, T.	VA	02/22/32	03/02/42	03/01/58
Jarvis, F.W.	MA	10/30/35	03/02/42	03/01/58
Hahn, A.	VA	04/13/43	03/02/42	03/01/52
Walker, R.	VA	03/16/51	03/03/55	03/02/59
Craig, R.	VA	04/28/58	03/03/65	03/02/70
Paddock, C.	MA	04/11/67	03/03/70	03/02/75
Abrahams, H.	SC	03/15/67	03/03/70	03/02/75
Williams, P.	NY	12/05/82	03/03/83	03/02/87
Tolan, E.	VA	02/09/73	03/03/75	03/02/81
		

Figure 2-3. Data Value Sample



Note: Data value samples are important DQE resources. The team should attempt to gather as many samples as time and resources permit. Good data value samples will

- Contain data values for a minimum of five related data elements.
- Contain at least 100 data records. (Examples in the text of the handbook are shorter to save space. Appendix D contains a set of practice data value samples).
- Identify the automated system used to generate the printed report.
- Identify the time frame covered by the report.

Summary

The team formally identified its customer and its customer's product. Next, the team worked on defining the problem area. It subdivided the problem area into components, and then subcomponents (as necessary), continuing until it reached the data element level. Finally, each data element was linked to its source and its output.

"I tell you," smiled John, "this all seems so logical! I don't get why I ever thought DQE was so difficult!"

Joe slapped John on the back. "I agree that when you break a problem area down into components and then elements, it seems to be a lot easier than when you and I used to go in and try to fix things on a whim and a wish!"

"Oh, is that right?" said Janine with a hint of sarcasm. "Maybe, I didn't hear you right, Joe, but I don't ever think I had to guess what was wrong with our program! I always knew! You messed with it!"

"Okay, Janine," said Joe raising his left eyebrow, "You're asking for a real comeback on that comment!"

Janine smiled. "Look, you're the one who made our customers believe you were doing everything you could to correct their problems. Little did they know that we had very little control over our program!"

John winced. "Don't say that so loud, Janine! Besides, I'm beginning to believe that we are getting some control over this thing! Focusing our effort on satisfying our customers was a good start as far as I'm concerned! Now breaking the problem into visible parts helps me to link the requirements to our production. DQE might save our jobs yet!"

"Jobs? What do you mean 'might save our jobs'?" asked Maureen with sudden interest in the dialogue.

“Just kidding,” said John forming his lips as if to whistle. “Look if we don’t get busy around here, we just might be in job jeopardy! Why don’t we call the professor and ask what’s next?”

“I’m for that!” said Janine. “He seems to always get us back on track, doesn’t he?”

Check on Learning

Responses can be found in appendix C.

1. Why is the customer a critical element in the DQE process?
2. Explain how the DQE team breaks down each problem area.
3. How does the team decide which data elements are related to the problem area?

Chapter 3

Creating Metadata

The team sat in the large, nearly empty room at the end of the hall. The movement of various customers and supervisors through John's office had interrupted the team's activities on a fairly regular basis. In fact, at one point, Maureen had thrown up her hands, saying she couldn't participate on a team that had to operate in such a chaotic environment. John had suggested that getting away from his busy work site might help facilitate communication and analysis. The team had agreed immediately.

Janine stopped reading her DQE handbook long enough to say "I have to admit that I am confused about what the term 'metadata' means. In chapter 3 of the handbook, I read that metadata is really just 'data about the data,' but I don't understand what that means. Does anyone here really understand the concept?"

"Maybe I can help you," said Maureen. "I'm not an expert, but based on what I read last night, I think I could give you at least an example of what 'metadata' means."

"Shoot!" said Janine.

"Okay." Turning toward Joe, Maureen asked, "What would you expect to find in the DLA phone directory?"

"Names, office codes, and telephone numbers, I guess," said Joe, a little irritated to find himself in the middle of Maureen's example.

"Great! Now, suppose you wanted to automate that book," said Maureen. "You know, create a database. Now what sort of information would you want to store in the database?"

Joe raised his eyebrow. "I think I've already said what was needed, Maureen—names, office codes, and telephone numbers!"

Maureen smiled. "Joe, you're really good at this! You just created three new data element names for our database!"

"But what does that have to do with metadata?" asked Janine.

"Well, a data element name is metadata," said Maureen. "In this case, Joe created three data elements named NAME, OFFICE CODE, and TELEPHONE NUMBER. Next, we would want to develop definitions for each of these elements. A data element definition is another example of metadata. In fact, chapter 3 contains several

other examples of metadata . . . I remember reading about 'domain,' 'observed range,' 'format,' and 'length.' Again, I am not an expert, but I could tell that each metadata element helped to define or limit the kind of data value that could be stored in a database."

"Whoa," said Joe. "There is a concept that still confuses me a little. What is a data value?"

Maureen had to think a minute, but soon responded with a question. "Joe, are you in the DLA telephone directory?"

"You know I am," said Joe, still looking a little peeved.

Maureen continued, "Then, your name would be just one of the data values in our new database. Your telephone number and office code would also be data values. In fact, if we just tore a page out of the directory, that page would probably be a good 'data value sample.'"

Joe's irritation started to diminish as he listened to Maureen. "She really seems to know her stuff..." he thought to himself. "Hey Maureen," blurted Joe, "you are really good at explaining these concepts. Can we go back to the metadata element called 'domain' for a minute? I have to admit that I still don't understand exactly what 'domain' means."

Maureen strained to think of a good way to describe the term. Finally she said "To me, 'domain' defines the raw materials or the building blocks that can be used to create data values. For instance, all the telephone numbers in the DLA directory are made up of some combination of the digits between 0 and 9. If that is true, the numbers between 0 and 9 are like the raw materials that can be used to 'build' data values for the data element called TELEPHONE NUMBER. In fact, I remember reading that the digits 0 through 9 form what is called the 'numeric' domain. TELEPHONE NUMBER in our new database would have a 'numeric' domain."

"This is scary!" laughed Janine. "I am starting to catch on!"

"Wait a minute," said Joe. "What about 'length'? That is the metadata element that confused me the most."

"Let me think of an example." Maureen paused, "I know! How about one called DAY OF THE WEEK? The data values for this data element are 'Sunday,' 'Monday,' 'Tuesday,' 'Wednesday,' 'Thursday,' 'Friday,' and 'Saturday.' What is the length for this element?"

"Seven," chimed Joe excitedly.

"Nine," said John, more resolutely.



"Why did you say 'nine,' John?" asked Joe. His excitement had now given way to curiosity.

"Well, I remembered that 'length' had to do with the length of the longest data value. In this case, I counted the number of letters in 'Wednesday' because it was the longest data value. It has nine letters so I figured the length was nine."

"John, I would say you're right" Maureen said with some assurance. Joe looked a little befuddled, but before he could ask another question, the professor entered the room.

"Well, well," said the professor. "What do we have here? It looks as if a serious discussion has been going on!"

"Well, sir, we were just reviewing some of the definitions of metadata. We're ready now to move ahead, though," said John.

"I'm pleased that you're reviewing and reenforcing ideas with each other. That shows team cooperation and it helps build a better understanding of the process!" said the professor. "Now, shall we begin?"

"I think we're ready, aren't we, team?" asked Maureen as she stood up to present her findings.

Each team member smiled as they waited for Maureen to begin her report.

Chapter 3

Creating Metadata

Introduction


In this chapter, the team continues to apply the principles of TQM. However, the actual tasks (and the steps within those tasks) now begin to take on the unique characteristics of Data Quality Engineering. In the first two tasks of this chapter, the team works to develop and analyze data element names, aliases, and definitions. In task 3, the team expands its knowledge about data by using concepts called “domain” and “length.” In the final task, the team adds the optional data elements called “format” and “observed range.” Collectively, the new information—data element name, alias, definition, domain, length, format, and observed range—is known as “metadata.” Metadata is simply data about the data.

Key Concepts

- ⚡ Metadata
- ⚡ Data element name
- ⚡ Data element alias
- ⚡ Data element definition
- ⚡ Data dictionary
- ⚡ General domain
- ⚡ Specific domain
- ⚡ Data element length
- ⚡ Data element format
- ⚡ Data element observed range




Task One


Define Metadata and Begin a Working Data Dictionary

 The DOD definition for metadata is:

“Information describing the characteristics of data; data or information about data; descriptive information about an organization’s data, data activities, systems, and holdings.”

This definition was originally developed by the National Institute of Standards and Technology (NIST) formerly known as the National Bureau of Standards (NBS Special Publication 500-152). Though descriptive information can include many different elements, task 1 will focus on just three metadata elements:

-  • Data element name – the data element name as listed in the data relationship table from chapter 2.
-  • Data element alias – an optional second or additional name for a data element. An alias is useful when the data element name provides little insight into its real meaning or when the meaning is obscured because of a limitation in the programming language or operating system. (Note: DOD does not recognize this metadata element).
-  • Definition – a short, clear phrase that describes the concept embodied in the data element name.

 In its original definition, NIST noted that metadata is stored in a data dictionary. At Defense Logistics Agency (DLA), the data dictionary will contain large amounts of metadata and will require a specialized, automated data dictionary software program. Within DLA, the data dictionary will be part of the DLA Corporate Repository. At this point, the team does not need specialized automated support. However, the team will want to adopt a standard technique for recording its metadata. One simple technique for recording metadata is to use 5”x7” cards.

(Task One continued)



Tip: Though the handbook assumes that the team is using the 5"x7" card technique, the actual technique used by the team to manage its metadata should depend on the amount of data to be collected, the talent of team members, and the availability of automated support. Automated support for this task varies from simple to sophisticated. Today's word processing software programs offer relatively simple-to-use, yet powerful data entry and management features. A spreadsheet program may also provide the capability to store, view, and organize metadata files. The team may also want to consider the design and use of dedicated database management software.

Step 1 Begin a data dictionary card file

Obtain a supply of blank 5"x7" cards. Create a format for each card as shown.

Example: Create a format for use on 5"x7" cards as shown in figure 3-1.

The image shows a stack of several 5x7 inch cards. The top card is a data dictionary card with the following fields and options:

- Data Element Name _____
- Data Element Alias _____
- Data Element Definition _____
- _____
- Domain (check either general or specific)
 - General
 - Alpha
 - Numeric
 - Alphanumeric
 - Other _____
 - Specific
 - List is located at/in _____
- Length _____
- Format _____
- Observed Range _____

Figure 3-1. A Data Dictionary 5"x7" Card



Note: This format also contains information that will be explained in later tasks. For now, the team should ignore all the metadata except the data element name, alias, and definition.

Step 2 **Note and record data element names and develop aliases**

Begin by writing one of the data element names from the data element diagrams on the top line of a 5"x7" data dictionary card. From the reports and other data sources gathered in chapter 2, find a data value sample where the data element name is used as a column header or other label for a set of data. If this situation exists, scan the column of data values. Does the data element name seem to describe the observed data values? If so, an alias (an optional second data element name) is not required. However, if the data element name does not describe the data values, choose a more descriptive name and record this name on the second line of the 5"x7" data dictionary card.

Example 1: The data element diagrams contain the `APPLICANT NAME` data element. The same data element name appears in a column heading in one of the data value samples (see figure 3-2). The team scans the data values and notes that they appear to be the names of applicants. In this case, the data element name describes the observed data values and an alias is not required. The team simply writes "applicant name" on the data element name line of a 5"x7" card.

Example 2: The team notes that the data element diagrams contain the `DATE OF BIRTH` data element. In the data value sample, the team notes the column header "DOB." A quick scan of the data values in the sample seems to support the idea that `DOB` and `DATE OF BIRTH` may be associated. In this case, the team would use `DATE OF BIRTH` as the data element name and `DOB` as an alias.

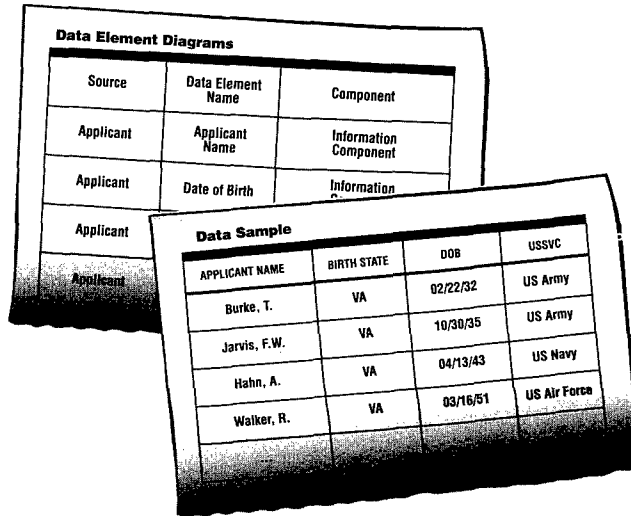


Figure 3-2. Data Element Names and Aliases



Note: Not all comparisons are this straightforward. Particularly when working with older (legacy) database systems, the team is likely to encounter data element names that convey very little meaning, like USSVC, FACTOR, or X135B. In these instances, actual data values may be the only clues the team has to establish a meaningful alias.

Example 3: The team reviews a data sample containing the information shown in figure 3-3.

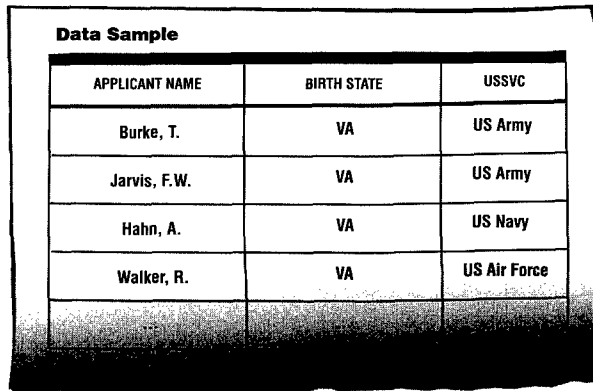


Figure 3-3. Data Sample

The team notes that the data values in the column labeled ussvc look like the names of the Services in the U.S. Armed Forces. Based on this observation, the team selects the more descriptive term BRANCH OF SERVICE as the alias for the ussvc data element.



Note: The team should not be overly concerned about making an error in deciding on a name. An error in a data element name or alias, if one occurs, will be discovered and corrected easily in later steps.

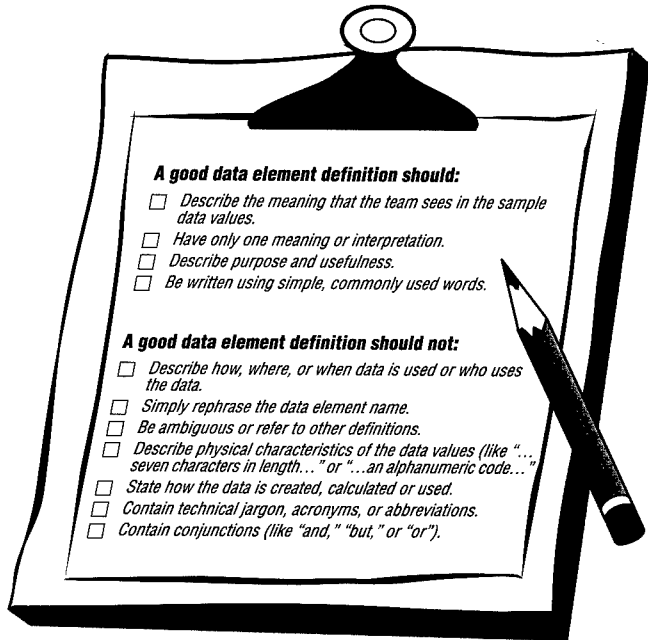
The team is likely to find that it has no (or poor) data value samples for some of the data element names that appear in the data element diagrams. In this case, the team returns to chapter 2, task 3, and renews its effort to obtain data value samples.

Step 3 Develop data element definitions

Look for a suitable definition in the written policy and procedure documentation that has been gathered so far. If this fails, interview SMEs. With their help, record a definition in the “data element definition” section of the 5”x7” card.



Note: Figure 3-4 contains a list of guidelines that will help the team to develop good data element definitions. The team may also want to refer to *Data Element Standardization Procedures* (DOD 8320.1-M-1) whose chapter 3 contains a comprehensive set of rules relating to the creation of data element names and definitions.



A good data element definition should:

- Describe the meaning that the team sees in the sample data values.
- Have only one meaning or interpretation.
- Describe purpose and usefulness.
- Be written using simple, commonly used words.

A good data element definition should not:

- Describe how, where, or when data is used or who uses the data.
- Simply rephrase the data element name.
- Be ambiguous or refer to other definitions.
- Describe physical characteristics of the data values (like "...seven characters in length..." or "...an alphanumeric code...").
- State how the data is created, calculated or used.
- Contain technical jargon, acronyms, or abbreviations.
- Contain conjunctions (like "and," "but," or "or").

Figure 3-4. Data Definition Guidelines

Task Two

Analyze Data Element Names and Definitions

In this task, the team will again focus on just the data element names and definitions recorded on the data dictionary 5"x7" cards. The team activities will consist of multiple searches of the cards. With each search, the team will be looking for duplication in either the data element names or their definitions.

In well-designed and managed database systems, data element names and definitions are not duplicated. However, in legacy systems, duplications can occur frequently. The instances of duplication (or redundancy) in data element names or definitions may cause data quality problems. Even if they are not a root cause of data quality problems at the moment, duplications will almost certainly cause problems in a future database integration effort.

Step 1 Find duplications in data element names

Begin by putting the data dictionary cards in alphabetical order by the data element name. Look for instances where two or more cards bear the same data element name but different definitions. Repeat this process using the data element alias. Repeat until every name has been compared with every other name and alias in the card stack. Make a list of all data elements that were identified as having duplicate names.



Note: Cards with identical data element names and definitions probably indicate that the team simply found the same element in two or more sources. Remove the duplicate cards, mark them as suspected duplicates, and store them in a separate location. The team should review the duplicate stack when it has completed the metadata development process, discarding confirmed duplicates and reevaluating cards that no longer have an exact match to the refined metadata card stack.

This effort is more than just a simple search for exact matches. For instance, the data element names BIRTH DATE, DATE OF BIRTH, and DOB, though all structured differently, may be redundant.

Step 2 Find duplications in data element definitions

Pull out the first card and study the data element definition. Read the data element definition on every other card and look for duplication. If the team suspects, but cannot be certain that duplication exists, assume that the data element definitions are redundant. Make a list of all data elements that were found or assumed to have redundant definitions. Repeat until every card has been compared with every other card in the stack.



Tip: If the team has been using an automated tool to store its metadata, it may want to experiment with the search and retrieval capabilities resident in the software program. However, for steps 1 and 2 in this task, there is no substitute for the intellect and reasoning power of the team members.



Note: This search may require considerable time and energy because the team is looking for duplication in the concept described in the data element definition (and not just similarities in word patterns). Consider the two data dictionary cards pictured in figure 3-5. Here, though little similarity exists in the data element names or even the words or the construction of the definitions, a careful review of the concepts being defined reveals strong similarities.

Data Element Name Capacity

Data Element Definition The petroleum, oil, or lubricant storage volume, measured in gallons

Data Element Name Storage Limit

Data Element Definition The total quantity of POL in gallons that can be stored

Domain (check either general or specific)

General

Alpha

Numeric

Alphanumeric

Other _____

Specific

 List is located at/in _____

Length _____

Format _____

Observed Range _____

Figure 3-5. Data Element Definition Comparison

Step 3 Discuss the results with subject matter experts

Create a data element duplication table. Label the first of two columns “Data Element Name” and the second column “Data Element Definition.” List the pairs of data elements that are suspected to have duplicate names (along with their definitions). Create a second table using the same format. On this table, list the data element names and definitions for the pairs that were suspected of having duplicate definitions. Review each pair with SMEs.



Note: Discussions with SMEs should help the team achieve a better understanding of the data. As a result of the discussion, the team may

- Conclude that duplication exists
- Revise the data element name on the data dictionary card to eliminate duplication
- Revise the data element definitions to clarify the meaning and eliminate duplication.

Task Three

Determine Data Element Domains and Lengths

So far, the team has captured data element names, aliases, and definitions. In this task, the team will add to its metadata by adding the elements called “domain” and “length.” Because these may be unfamiliar terms, a set of definitions and examples is provided below.

The domain for a data element defines the kind of written character that can be used to form a valid data value. DOD 8320.1-M-1 describes two types of domains—general and specific. Here is the DOD definition of a general domain, followed by a few examples to help clarify this concept.



Definition: General Domain – the set of permissible data values from which actual values are taken.

Example: A general domain of “A to Z” means that data values can only be constructed using the letters of the alphabet (the shorthand term for this domain is “alpha characters” or just “alpha”).

A general domain of “0 to 9” (called the “numeric” domain) means the data values can only be constructed using the digits 0 through 9.

An “alphanumeric” domain allows both numbers and letters of the alphabet (but may not allow symbols like parentheses, slashes, etc.). Figure 3-6 contains examples of both valid and invalid data values for various general domains.



Note: The rules regarding the validity of symbols like dashes, parentheses, slashes, punctuation, and other symbols will vary by system.

DOD offers the following as a definition of “specific domain.”



Definition: Specific domain – an enumerated set of data values allowed in representations of a data element.

		Type of Domain		
		Alpha	Numeric	Alphanumeric
Sample Data Values	A	Valid	Invalid	Valid
	4	Invalid	Valid	Valid
	Smith	Valid	Invalid	Valid
	Smith 3rd	Invalid	Invalid	Valid
	Sm1th	Invalid	Invalid	Valid
	(703) 555-1234	Invalid	Invalid	Valid

Figure 3-6. Domain Examples

Example: A specific domain usually consists of a list of acceptable data values. For instance, the specific domain for a data element used to store the answer to a yes or no question might be listed as simply “Y” or “N” (no other values are valid). The specific domain for a data element called MONTH may include “January,” “February,” and on through “December.” A specific domain list is sometimes referred to as a “look-up” table.

The general and specific domain definitions can be confusing. One way to decide if a domain is general or specific is to think about how the data values are chosen. Could each value be picked from a limited set of choices or was each value created by combining certain raw materials? If each value could have been picked from a limited set of choices (in other words, from a list), then the specific domain definition applies. Is it likely that each data value was built from “raw materials” using any of the letters in the alphabet and/or the digits between 0 and 9? If so, then the general domain applies.

In addition to the domain metadata, DOD provides the following definition for the metadata element called “maximum character count quantity.”

Definition: Maximum character count quantity – the maximum quantity of characters that can be used to describe a data value.



For the purposes of this handbook, the term “length” will be used in place of “maximum character count quantity.”

Example: In the yes or no example above, the stored data value can only be the character “Y” or “N.” The length, in this case, would be one. For the MONTH data element in the example above, the length would be nine (because September, the longest name for a calendar month, contains nine letters).

Step 1 Gather a data sample for each data element

Find a data value sample for each data element in the data dictionary card file. If a data value sample cannot be located, describe the data element to a functional expert and request assistance in generating a data value sample.



Note: Data value samples may contain data values on more than one data element. In fact, as the team will see later, the most useful samples contain values from a number of data elements.

Step 2 Study pertinent documentation

Search for domain and length information in the system operating manuals, standard operating procedures, or other documentation already gathered by the team. Validate domain and length information by comparing the documented domain and length to the data values in the sample.

Example 1: In its review of existing system documentation, the team finds a paragraph that describes the SOCIAL SECURITY NUMBER data element as “...nine characters in length and consists of just the numbers 0 through 9.” Based on this information, the team decides that SOCIAL SECURITY NUMBER has a general (numeric) domain. A quick scan of data values in the sample confirms the team’s decision. On the data dictionary card, under domain, the team checks the boxes marked “general” and “numeric.” The team decides that the length for this data element is nine. The card now looks like figure 3-7.

Data Element Name Social Security Number

Data Element Alias SSN

Data Element Definition a number, unique to an individual, assigned by the government, originally intended to identify each individual's contribution to a savings fund that can be accessed after reaching a certain age.

Domain (check either general or specific)

General

Alpha

Numeric

Alphanumeric

Other

Specific

List is located at/in _____

Length 9

Format 999-99-9999

Observed Range None

Figure 3-7. The SOCIAL SECURITY NUMBER 5"x7" Card

Example 2: As a result of its search for domain and length for the EYE COLOR data element, the team finds the following phrase in an operational procedure manual "...select eye color from the following list: blue, brown, black, gray, green, or hazel." Based on this information, the team concludes that EYE COLOR has a specific domain (a list of only six possible values). Based on a review of this list, the team assigns this data element the length of five (the longest data element value is five characters long). The team scans the data values in its sample and notes that all the values appear to have come from this list. The team decides to store the data value list on the back of the data dictionary card.

Step 3 Create domain and length metadata

If documentation is missing or incomplete, create the domain and length metadata by conferring with SMEs. Validate the SME recommendations by comparing their response to the data values in the samples. Record the domain and length metadata on the data dictionary 5"x7" cards.

Example: The team notes a data element called CCC. The team's review of documentation produces nothing relating to domain and length. In earlier discussions with SMEs, the team developed an alias for this data element called CARGO CATEGORY CODE. The team also developed this definition: "a code used to describe the different ways in which military equipment or supplies can be packaged for shipment." The team studies the data value sample and notes that all the data values are three characters in length and that they all begin with two letters of the alphabet and end with a one-digit number between 0 and 9. Based on its observation of the data value samples, the team decides that the CCC data element has a general, alphanumeric domain and a length of three.

Task Four

Determine the Format and Observed Range

Up to this point, the team has been busy gathering and recording the data element name, domain, and length metadata elements. These three metadata elements are recognized by DOD and defined in DOD 8320.1-M-1. The team has also gathered a metadata element called the alias. As explained earlier, the alias metadata element is not recognized by DOD. It is included here because the data element name in legacy systems sometimes provides little insight into the meaning or use for the data. In this task, the team will add two more metadata elements that are useful, but not recognized by DOD. These are “format” and “observed range.” Here is a definition for format.



Definition: Format (called “picture” in some legacy database systems) defines a certain convention for storing or displaying data values.

Example: Consider the following number:

7035551234

Now consider the same number, but with formatting conventions:

(703) 555-1234

The addition of a format convention helps the reader interpret and understand the data value. This particular format has been used effectively for displaying telephone numbers.

In legacy systems, format is usually left to the person making the initial data entry. Over a period of time, data values are entered without applying a consistent format. The result is that format becomes a source of data quality problems, particularly when merging data from a legacy system into another database system.

The observed range, as the name implies, is established by the team by simply observing the highest and lowest data value in a given data value sample. Observed range only applies to the data elements used to store numeric values. Here is a definition for the observed range metadata element.



Definition: The observed range is the complete set of values between the lowest (or smallest) and the highest (or largest) value in a given set of data values.

Like format, observed range can be a powerful tool for evaluating data quality. The procedure for generating format and observed range metadata consists of simply obtaining and studying data value samples. Both format and observed range are optional metadata elements. They may not apply to every data element.

Step 1 Gather a data sample for each data element

Find a data value sample for each data element in the data dictionary card file. If a data value sample cannot be located, describe the data element to a functional expert and request assistance in generating a data value sample.



Note: Data value samples may contain data values on more than one data element. In fact, as the team will see later, the most useful samples contain values from a number of data elements.

Step 2 Study the data values and look for format

Study the list of data values carefully and look for any commonly used techniques for displaying or recording the data values. Record the most common technique on the data dictionary card on the line called "Format."



Note: The team will need to establish a shorthand method for recording format. The method that the team uses will depend on the personal preferences of the team members and on the nature of the data being studied. A few methods are embedded in the following examples.

Example 1: The team looks over a data value sample for the data element called EXPIRATION DATE. Here are a few of the data values noted:

04/12/96
05/31/97
01/01/99
12/30/95

The team observes a strong pattern in this sample. Based on its observation, the team records the EXPIRATION DATE format using this notation:

MM/DD/YY

where MM, DD, and YY (representing numeric data) are separated by the slash symbol. MM is a number between 01 and 12, DD is a number between 01 and 31, and YY is a number between 95 and 99.

Example 2: The team looks over a data value sample for the data element called SSN. Here are a few of the data values noted:

123-45-6789
234-56-7890
345678901
456-78-9012
567-89-0123

The team notes that with one exception, all the data values are displayed with a dash between the third and fourth characters and between the fifth and sixth characters. On the format line of the data dictionary card, the team writes the format for the data element called SSN using this notation:

999-99-9999

where a nine represents numeric data (any digit between 0 and 9) and dashes separate the third and fourth and the fifth and sixth digits.



Note: The team selects a format based on its observation of that format in a significant majority of the data values in the sample. At this point, the team will not be able to determine whether an exception represents a valid new data value or a failure to comply with a format convention. After working with other samples or with all the data, the team may elect to modify its format metadata.

Example 3: The team looks over a data value sample for the data element called ADDRESS. Here are a few of the data values noted:

123 Oak Drive
P.O. Box 123
Route 1, Box 2
234 Main Street

The team decides that this data value sample contains no commonly used technique for representing the data values. The format line is left blank.

Step 3 Study the data values and look for observed range

Go over the list of data values again. This time, study just the data value samples that contain numeric data. For each sample, try to find a lowest and a highest value. Record the lowest and highest values on the data dictionary card on the line called “Observed Range.”



Tip: A number of software programs, notably spreadsheet software programs, support the requirement to identify the low and high values in a range of values.

Example: The team looks over a data value sample for the data element called `VEHICLE WEIGHT`. Here are a few of the data values noted:

3,000

5,150

6,001

2,995

3,125

In this example, the smallest data value is 2,995 and the largest is 6,001. The team records “2,995 to 6,001” on the observed range line of the data dictionary card.



Note: The observed range is initially established by observing a sample of data values. Later, when working with all or different sets of data, the team may find that this initial effort to establish the observed range was not accurate. At that point, the team will simply revise the observed range metadata.



Note: Observed range does not apply to all numeric data elements. For instance, the observed range would be meaningless for the `SSN` data element used as an example in step 2.

Summary

The team should now understand the concept of metadata. It should be able to discuss the various kinds of metadata and be able to explain how metadata defines or establishes limits on the data values stored in the database.

The team sat around the cafeteria table munching on a 'Mama Pizzeria' special. John lifted his mug and said, "A toast to my hard-working, dedicated team!" Others raised their drinks. "Cheers!" they said in unison.

The professor laughed. He had been invited to join them after their long days of sorting through the reams of data samples provided them. It had not been easy to build up their metadata resources, but in the end, the team believed it had captured the information it needed. The professor was proud of the initial DQE work.

"Well, team," said the professor. "Are you ready for the next set of DQE tasks?"

"You mean there's more?" asked Janine with some disbelief. "I thought this would wrap it up." Grabbing a slice of pizza, she continued, "Now that we know what should and should not be present in the data, can we just tell the programmer not to allow the entry of the unacceptable?"

"I wish it were that simple, Janine," responded the professor. "Unfortunately, the unacceptable data is only a symptom of the existence of a deeper rooted problem. But we can't get at the real problem until we establish some rules."

"What rules?" asked Joe.

"Meet me at 0700 hours tomorrow in your DLA office and I'll show you," said the professor.

John laughed. "I think he really is beginning to sound like the military, don't you, gang?" The team laughed out loud.

Check on Learning

Responses can be found in appendix C.

1. List some good characteristics of a data element definition.
2. Distinguish between the concepts of “specific” and “general” domain.
3. Write a format specification for a data element called ZIP CODE. Explain the conventions you used.

Practical Exercises

Use the sample data values provided in appendix D. Answers are provided in appendix E.

1. Create the definition and domain metadata for the data element called NAME:

Definition:

Domain:

2. Create the length and format metadata for the data element called ZIP CODE:

Length:

Format:

3. Create the observed range metadata for the data element called WT:

Observed range:

Chapter 4

Building Metadata-based Business Rules

I have one question,” began the professor as he swung his jacket over a nearby chair. “How do you live with all that traffic? Cars coming off two highways, merging, and then pouring into your base here! What a mess! I felt like a data element, caught in a merger between gigantic automated systems! At one point I wasn’t sure who I was, what I supposed to do, and whether I would be useful at all this morning!”

“I guess we forgot to tell you, sir.” Joe laughed. “On military installations, like this one, most people arrive between 0700 and 0730 hours. In fact, by eight o’clock, our customers are usually waiting impatiently at our service desk!”

“Well, then, let’s not waste time,” said the professor as he rolled up his sleeves. “We are here to meet their needs, right?”

“Right!” said John.

“The order of the day is for us to build business rules based on the metadata we gathered earlier. Now, let me ask this question first. What did we gain by working with data samples to develop metadata?”

Maureen looked around at her peers before raising her hand. “Well, for one thing, I learned what the terms ‘alias,’ ‘general domain,’ ‘specific domain,’ ‘observed range,’ and ‘length’ really mean. But more than that, by working so intently on the metadata, I found that I gradually got a better and better picture of the how data has been defined and how it is used here at DLA!”



“Terrific!” exclaimed the professor.

Maureen’s eyes lit with excitement. Before she could offer more explanation, Janine interrupted. “Let me tell you what I discovered through my work with metadata,” she exclaimed. “Would you believe, I found about forty data value errors in one relatively small sample?”

The professor responded with a broad smile. “Sounds as if there was some success with the use of metadata! Let me ask you all something, though. How could

Janine find so many data value errors by simply knowing something about the metadata for the data sample?"

"I'll answer that one!" interjected John. "I bet Janine learned how to apply the domain and range metadata... well ... I guess all the metadata in a specific way." Looking around, John found only silence and a few expressions of confusion.

"What I mean," he began again, "is that she used metadata to construct business rules. She then simply used the rules to test the data values in her sample."

"Excellent!" exclaimed the professor. "That's probably just what happened. In fact, John, the process you just identified is the process by which we get our metadata-based business rules!"

"Our what?" asked John, squinting his eyes.

"Our metadata-based business rules," repeated the professor. "For us, metadata serves two key purposes. First, it helps us understand how DLA has defined and how they use their data. But perhaps equally important, we have developed a way to use our new insight to actually check the quality of stored data values. To do that, we developed metadata-based business rules. Maybe we ought to try to define a business rule. Would anyone like to take a shot at a definition?"

Joe raised an uncertain hand. "To me, a business rule is some statement of fact about the data."

"Very good, Joe," stated Professor Hopkins, adding, "A business rule is simply a written expression of fact about the data. Each rule is written in a way that allows a team of analysts, like ours, to check to see if the data values in the database agree (or comply) with the rule. We will eventually learn to develop business rules from several sources, but for now, our source is metadata."

"Hey, come to think of it, that's exactly what I did," added Janine.

"Why don't you tell us about it, Janine?" asked the professor.

Janine replied, "I was working with a file that contained data on the date that requisitions had been processed. According to our metadata, the format for the data was DD/MM/YY, where DD was a number representing the calendar day, MM was a number between 01 and 12 (representing the calendar month), and YY was a number that represented the last two digits of the year. I was able to create a whole series of business rules based on this information. For instance, I created a rule that said 'The value for MM had to be a whole number between 01 and 12.' When I checked my data values, I found two instances where the MM was '00' and three others where the value was larger than '12.' All together, I

created ten business rules for this one data element and that is how I found forty suspected data value errors!”

“That is the concept, Janine” exclaimed the professor once again. Then, with a twinkle in his eye, he added, “Okay, troops, you know the rules. Let’s stick to them!”

Chapter 4

Building Metadata-based Business Rules

Introduction

In this chapter, the team builds and uses business rules. For Data Quality Engineering, a business rule is defined as "...a statement of fact about the data that an organization creates, maintains, or uses." Business rules can be found in written policy statements, procedures manuals, and many other sources. However, in this chapter, the source for business rules is limited to metadata (specifically, the data the team has created and stored on its 5"x7" cards).

By using metadata to build business rules, the team benefits in two ways. First, it learns to use metadata to identify data value errors. Second, the analysis of the data values, coupled with focused discussions with SMEs, helps the team build on or "fine-tune" metadata resources.

Key Concepts

- ⚡ Business rule
- ⚡ Domain-based business rule
- ⚡ Data value error list
- ⚡ Format-based business rule
- ⚡ Mean
- ⚡ Standard deviation
- ⚡ 95 percent interval
- ⚡ Observed range business rule

.....

Task One

Build a Domain-based Business Rule



A business rule is a statement of fact about the data that an organization creates, maintains, or uses. To build its first set of business rules, the team focuses on the metadata element called “domain.”

Step 1 Search the data dictionary card file

Search through the data dictionary card file and find all the cards with the alpha domain.



Tip: If the team has been storing its metadata in an automated file, this search can be done using “find” or “search” capabilities, if resident in the software program.

Example: The team finds that the alpha domain applies to the data elements LAST NAME and FIRST NAME.

Step 2 Develop a business rule statement



Write a statement that expresses the limits imposed by the alpha domain. Use the data element names in the statement. The statement that results from this step is a domain-based business rule.

Example: For the data elements called LAST NAME and FIRST NAME, each character used to form a valid value must be a letter between A and Z.

Step 3 Begin a business rule list

On a separate piece of paper, begin a numbered list of business rules (see figure 4-1). The statement from step 2 is business rule #1.

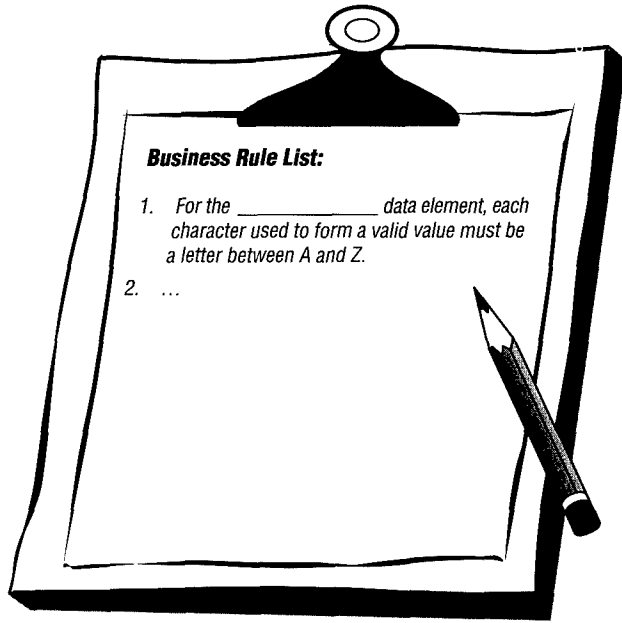


Figure 4-1. The Business Rule List




Tip: The team could potentially create many business rule lists, each containing a significant number of rules. The team may want to consider storing its business rules using word processing or other automated support tools.




Note: This particular rule applies to two data elements (in actual practice, a single rule may apply to many data elements). To avoid writing a long list of nearly identical rules (changing only the data element name), the team may elect to leave a blank space in place of the data element name. Using this technique, business rule #1 can be applied to LAST NAME, FIRST NAME, OR any other data element with an alpha domain.

Step 4 Apply the business rule to a data sample

Find the data value sample that contains the data element(s) identified in step 1. Examine each data value and highlight those that are constructed using anything but the letters between A and Z.


 **Note:** Recall that SMEs have already reviewed some data value samples and have highlighted errors using a gray maker. To help keep track of the source of its marks, the team will use a different color. For this handbook, the team will use blue.

 **Tip:** The process of applying a business rule to a long list of data values can be tedious. Experience indicates that for all but the shortest lists, the team will benefit if it can make use of automated support. Given access to automated data value samples, the team may find that some business rules can be implemented using word processing programs. Other rules can be implemented simply using spreadsheet or database software programs. Some computer-aided software engineering (CASE) and a limited number of other specialized support tools offer a much broader capability. Some will accept database files as well as store, organize, and even apply business rules automatically.

Example: The team finds a data sample like the one shown in figure 4-2. It examines each LAST NAME and FIRST NAME data value on the table and highlights (using a blue highlighter) the entries that do not comply with the business rule.

Data Value Report			
Last Name	First Name	Birth Town	Birth St.
Adams	John	Quincy	MA
JeffersOn	Thomas	Shadwill	Vir
Burr	Aaron	Newark	NJ
Clinton	George	Ulster	NY
Gerry	Elbridge	Marblehead	MA
Tompkins	Daniel D.	Scarsdale	NY
Calhoun	John C.	Abbeville	SC
Van Buren	Martin	Kinderhook	NO
Johnson	Richard M.	Louisville	KY

Figure 4-2. Data Value Sample

 **Note:** “JeffersOn” was highlighted because the eighth character is not a letter between A and Z.

(Task One continued)

“Van Buren” was highlighted because the space (written as <space> in this handbook) between words is considered a character in automated systems, but <space> is not a permitted character in the domain for this data element. The other three data values were highlighted because they contained both a <space> and punctuation (a period). Like <space>, a period is not a character between the letters A to Z.

In this example, the domain does not include a restriction on the use of uppercase (capital) and lowercase letters. In practice, however, the team will find that the significance of case will vary. Some automated systems “ignore” case (treat upper- and lowercase letters as equivalent characters). Other “case-sensitive” systems uniformly store and treat upper- and lowercase letters as different characters. Further, some systems may impose case restrictions on an element by element basis. The team should discuss case restrictions with SMEs and include case limits in the domain definitions and in the business rules any time that they apply.

Step 5 Consult with subject matter experts

Discuss the highlighted data values with SMEs. Ask them for an opinion whether the highlighted values are errors.

Step 6 Revise metadata if necessary

Based on the new insight provided by SMEs, review the metadata and revise it if necessary.

Example: The team groups all the data values for LAST NAME in which a <space> appears. After review by SMEs and discussions with the team, all reach a conclusion that the data values are correct as written. Based on this information, the team modifies the domain metadata for LAST NAME to include alpha and <space> characters.

The team groups all the data value errors for the `FIRST NAME` data element that contain a `<space>` followed by a single alpha character. After review, the SMEs and team agree that the `<space>` followed by a single character (with or without a period) is probably an individual's middle initial. A SME points out that, although not shown on this report, a data element called `MI` has already been created to store an individual's middle initial data. Based on this information, the team keeps the `FIRST NAME` metadata as written.



Note: When a SME refers to another data element, the team should review its 5"x7" cards to ensure it has a card on the element. If not, the team creates a new card using the procedures described in chapter 2.

Step 7 **Revise the business rule if necessary**

Based on the new or revised metadata, change or add business rules as necessary.

Example: The new business rule list looks like figure 4-3.

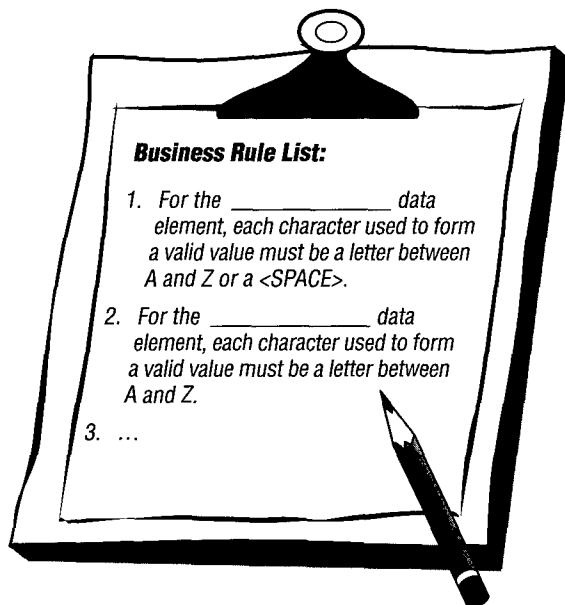


Figure 4-3. The Business Rule List (continued)

Step 8 Record the business rule/data element relationship

On the back of each card, write the business rule number(s) that apply to the data element.

Example: On the back of the LAST NAME card, write “Business rule #1.” On the back of the FIRST NAME card, write “Business rule #2.”

Step 9 Start/continue a data value error list

⚡ Begin (or when one already exists, add to) the list of records containing data value errors (with the errant data values highlighted). In a new column on the report, write the number of the business rule that was used to identify the data value error.

Data Value Error List

Last Name	First Name	Birth Town	Birth State	
Jefferson	Thomas	Shadwill	Vir	1
Tompkins	Daniel D.	Scarsdale	NY	2
Calhoun	John C.	Abbeville	SC	2
Johnson	Richard W.	Louisville	KY	2

Figure 4-4. The Data Value Error List



Tip: The creation of the data value error list can be a tedious process. If error lists and data samples are large, the team should consider working with automated data value samples and creating its error lists using the capabilities of word processing, spreadsheet, database, or CASE tool program support.



Note: At this point, the team should not attempt on-the-spot corrections. Because the team is still working with samples and because a root cause has not been determined, attempts to make data value corrections are likely to be counterproductive.

Step 10 Repeat for the numeric and alphanumeric domains

Repeat steps 1 through 9, using data elements with a numeric and then an alphanumeric domain. Add the new business rules to the list of rules. Add records containing suspected data value errors to the suspected error list. Add a new page to the data error list each time the team uses a new data sample.

Step 11 Repeat for the specific domain

Repeat steps 1 through 9, using data elements with a specific domain. Add the new business rules to the list of rules. Add records containing suspected data value errors to the suspected error list. Add a new page each time the team uses a new data sample.



Note: In step 4, in addition to finding the data value sample, the team will also need to find the specific domain list associated with this data element (recall that this list was also called a “look-up” table). With the data sample and the specific domain list side by side, the team examines each data value in the sample and compares it to (or “looks it up” in) the list of permitted values in the specific domain.

Example: The data value sample from step 4 contains the BIRTH STATE data element. This data element has a specific domain consisting of a list of 54 possible values (a set of codes representing the names of the 50 U.S. states, the nation’s capital, Guam, Puerto Rico, and the Virgin Islands). That list can be found in appendix F. After repeating step 4, the team’s sample data value list now looks like figure 4-5.

Data Value Report

Last Name	First Name	Birth Town	
Adams	John	Quincy	
Adams	Thomas	Shadwill	
Burr	Aaron	Newark	NJ
Clinton	George	Ulster	NY
Gerry	Elbridge	Marblehead	MA
Tompkins	Albany	Scarsdale	NY
Calhoun	Abbeville	Abbeville	SC
Martin	Kinderhook	Kinderhook	
Johnson	Louisville	Louisville	KY
	

Figure 4-5. Highlighted Errors in State Codes

During its discussions with SMEs, the group concludes that all the highlighted values are data errors because they were not values from the specific domain list for this data element.



Note: Steps 1 through 9 represent a logical start-to-finish approach that will be most useful to the team performing DQE for the first time. An experienced team may elect to delay step 5 (the meeting with SMEs) until it has finished all the domain-based analyses.

Task Two

Build a Format-based Business Rule

In this task, the team will be building business rules based on “format.”

Step 1 Search the data dictionary card file

Search the data dictionary cards and find all the cards with a format specification.

Example: The team finds that format applies to the ZIP CODE data element. The format looks like this:

99999-9999

where a dash separates the first five characters (here the number nine represents any digit between 0 and 9) from the last four characters. The first five characters are required (cannot be blank).



Note: Format is an optional metadata element. Not all data elements will have a specified format.

Step 2 Develop a business rule statement



Rewrite the limits imposed by format specifications as a statement. Use data element names in the statement. The statement that results from this step is a format-based business rule.

Example: The ZIP CODE data element consists of five characters (mandatory) followed by an optional dash and four characters.



Note: A domain-based business rule would be applied to ensure that the characters in ZIP CODE are numeric.

Step 3 Add the new rule(s) to the business rule list

Rewrite the limits imposed by the format statement as a rule or a set of rules (see figure 4-6).

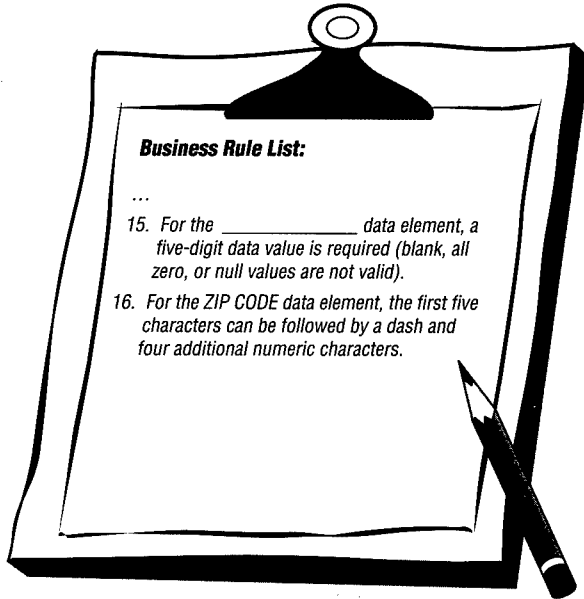


Figure 4-6. The Business Rule List (continued)



Note: In this case, two business rules were written because a single rule would have had several parts and might have been hard to understand or apply.

Step 4 Apply the business rule to a data sample

Find the data value sample that contained the data elements identified in step 1. Examine each data value and highlight those that do not comply with the new business rule(s).

Example: The team examines the ZIP CODE data values in the following sample and highlights (using a blue marker) the entries that do not comply with the business rules shown above (see figure 4-7).

Zip Code Data Value Sample

City	State Code	Zip Code
Baltimore	MD	21202
Baton Rouge	LA	70821 - 3217
Birmingham	AL	34202-2027
Boston	MA	
Buffalo	NY	14202
Charlotte	NC	28232
...		

Figure 4-7. ZIP CODE Data Value Sample



Note: The data value 70821-3217 was highlighted because a <space> appears before and after the dash. This difference may seem trivial. However, this situation can have significant repercussions. Consider the instance where this data is used to generate addresses. A legacy system used to generate addresses and print envelopes will probably print the last line of the address like this:

Baton Rouge, LA 70821 - 32

This situation occurs because the system allocated room for only 9 characters (10, counting the dash) and 2 of those characters in this data sample are <space> characters. The system then truncates or “cuts off” the remaining data.



Note: The ZIP CODE data value for Boston, MA, is blank. Because a value is required, the empty space is highlighted.

Step 5 Consult with subject matter experts

Discuss the highlighted data values with SMEs. Review highlighted data values, grouped by type of discrepancy, but also individually with SMEs. Ask them for an opinion whether the highlighted values are errors.

Step 6 Revise metadata if necessary

Based on the new insight provided by SMEs, review the metadata and revise it if necessary.

Example: The team groups all the records in which the data value for ZIP CODE is blank. SMEs review the group quickly and concur that data values are mandatory. The team decides to keep the metadata as written.

The team groups all the records in which the data value for ZIP CODE contains the <space> character. Again, SMEs concur that highlighted data values are probably errors and the team keeps the metadata as written.

Step 7 Revise the business rule if necessary

Based on the discussions with SMEs, modify the business rule developed in step 3.

Example: In this instance, no changes are required.

Step 8 Record the business rule/data element relationship

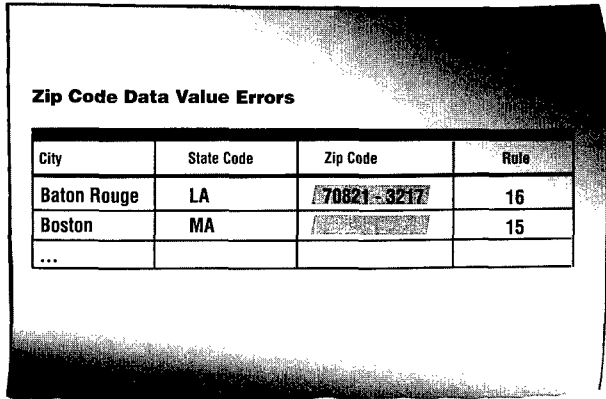
On the back of the 5"x7" cards found in step 1, write the number of the applicable business rule(s).

Example: On the back of the ZIP CODE 5"x7" card, the team writes "Business rules #15 and #16."

Step 9 Continue the data value error list

If the team located any new errors, it starts or continues the data error list.

Example: The new page might look like figure 4-8.



The image shows a screenshot of a software interface with a table titled "Zip Code Data Value Errors". The table has four columns: "City", "State Code", "Zip Code", and "Rule". The first row shows "Baton Rouge" for the city, "LA" for the state code, "70821-3217" for the zip code, and "16" for the rule. The second row shows "Boston" for the city, "MA" for the state code, a blurred zip code, and "15" for the rule. A third row contains three dots "...".

City	State Code	Zip Code	Rule
Baton Rouge	LA	70821-3217	16
Boston	MA		15
...			

Figure 4-8. ZIP CODE Data Value



Note: Certain business rules based on format tend to apply regardless of the database system. For instance, a nearly universal set of business rules exists to check the data value for the calendar date, stored in the DD/MM/YY format (described in chapter 3). The set of rules that could be applied to a data element with this format have been developed and are recorded for the team's use in appendix G.

Task Three

Build a Business Rule Based on Observed Range

In this task, the team will be building business rules based on “observed range.”

Step 1 Search the data dictionary card file

Search the data dictionary cards and find all the cards with a specified observed range.

Example: The team finds that observed range applies to the BODY WEIGHT data element.

Step 2 Find a data value sample

Find the data value samples for the observed range data elements.

Example: The team finds the BODY WEIGHT data sample (see figure 4-9).

Data Value Report			
Name	Body Weight	Name	
Burke, T.	150	Owens, J.	
Jarvis, F.W.	141	Harrison, D.	
Hahn, A.	115	Remigino, L.	139
Walker, R.	205	Morrow, B.	144
Craig, R.	266	Hary, A.	181
Paddock, C.	155	Hayes, B.	301
Abrahams, H.	10	Hines, J.	150
Williams, P.	160	Borozov, V.	137
Tolan, E.	175	Crawford, H.	201
		Lewis, C.	143

Figure 4-9. BODY WEIGHT Data Value Sample



Note: The sample size in this example was limited to 20 records to focus attention on the steps in this process (and not on the review of large amounts of data). In actual practice, however, the sample size of a hundred or more records should be used for this task.

Step 3 Calculate the mean and the standard deviation

Use a specialized hand-held calculator or a spreadsheet software program, calculate the mean and the standard deviation of the data values for BODY WEIGHT in the sample.

Example: The team enters the data values into a specialized hand-held calculator one at a time and, following the instructions for the device, calculates the mean and the standard deviation. For this sample, the mean is 160.85 and the standard deviation is 58.11.



Note: A technically precise definition and a detailed description of the calculation techniques for the mean and standard deviation are beyond the scope of this handbook. Additional information on these two concepts can be found in the DLA handbook *It's a Statistics Jungle Out There*. For this handbook, the following descriptions of these two terms will suffice:



Definition: The mean (sometimes called the “average”) is a measure of a central point in a set of evenly distributed data values. The terms “batting average” in sports or “grade point average” from school are familiar examples of a mean.



Definition: The standard deviation is a measure of how the data values are spread out around the mean. If the standard deviation is small, then most or all of the data values are clustered around the mean. If the standard deviation is large, then the data values vary widely. Figure 4-10 compares the number of times that each data value occurs against the value of each occurrence. The graph on the left shows data with a small standard deviation. The graph on the right shows data with a larger standard

deviation. In both cases, the data is distributed evenly around the mean.

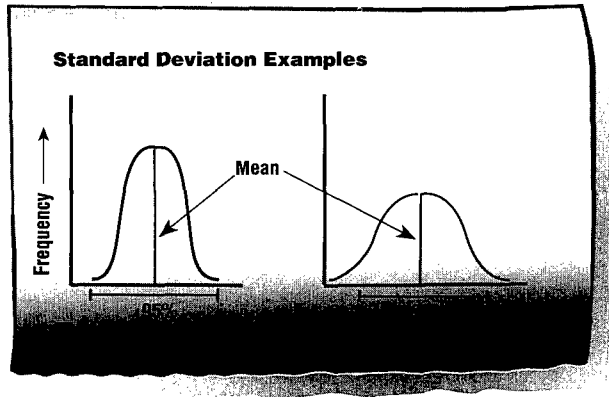


Figure 4-10. Standard Deviation Examples

Step 4 Calculate the 95 percent interval

Calculate the value for two standard deviations by multiplying the standard deviation by 2. Subtract the value of two standard deviations from the mean (to find the low end of the 95 percent interval). Next, add the value of two standard deviations to the mean (to find the high end of the 95 percent interval).

Example: The standard deviation is 58.1. Multiplied by 2, the value for two standard deviations is:

$$58.1 \times 2 = 116.2$$

The low end of the 95 percent interval is found by subtracting two standard deviations from the mean. The mean is 160.8.

$$160.8 - 116.2 = 44.6$$

The high end of the 95 percent interval is found by adding two standard deviations to the mean.

$$160.8 + 116.2 = 277.0$$

The 95 percent interval is 44.6 to 277.0 for this data sample.



Note: The mathematical principles that underlie the 95 percent interval calculation go beyond the scope of this handbook. The key is to understand what the 95 percent interval means. Simply stated, approximately 95 percent of the data values in this sample will be between the values 44.6 and 277.0 or, in other words, approximately 5 percent of the values will either be smaller than 44.6 or larger than 277.0.

Step 5 Develop a business rule

Write a statement that identifies data values that deviate from the mean by two standard deviations. The statement that results from this step is a business rule based on observed range.



Example: Approximately 5 percent of the data values in the BODY WEIGHT sample will be smaller than 44.6 or larger than 277.0.



Note: The fact that a data value lies outside this interval does not mean it is an error. However, any value outside this interval is one that deviates significantly from the majority of other values in the sample. This deviation may be legitimate, but may also be the result of an error. Because the team cannot be certain, the values at this point are “suspect.”

Step 6 Add the new rule(s) to the business rule list

Rewrite the limits imposed by the format statement as a rule or a set of rules (see figure 4-11).

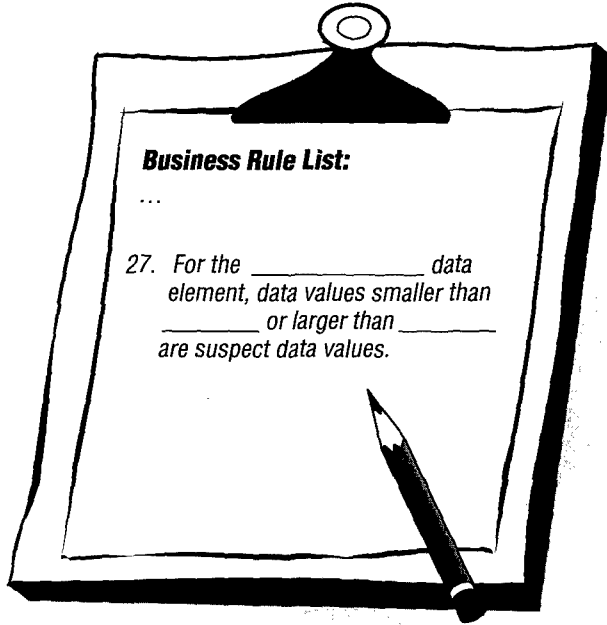


Figure 4-11. Business Rule List (continued)



Note: Like the data element name, the range limits for the business rule are left blank because they will vary by sample. For this sample, the range was 44.6 to 277.0.

Step 7 Apply the business rule to a data sample

Examine each data value and highlight those that do not fall within the 95 percent interval.

Example: The team examines the data value sample (from step 2) and highlights the entries as shown in figure 4-12.

Body Weight Data Sample			
Name	Body Weight	Name	Body Weight
Burke, T.	150	Owens, J.	110
Jarvis, F.W.	141	Harrison, D.	165
Hahn, A.	115	Remigino, L.	139
Walker, R.	205	Morrow, B.	144
Craig, R.	266	Hary, A.	181
Paddock, C.	266	Hyes, B.	301
Abrahams, H.	10	Hines, J.	150
Williams, P.	160	Borzov, V.	137
Tolan, E.	175	Crawford, H.	201
		Lewis, C.	143

Figure 4-12. BODY WEIGHT Data Sample with Highlighted Errors



Note: In this particular sample of 20 values, 2 (or 10 percent) of the values were outside the range of 44.6 and 277.0. Had this been a larger sample, the number of suspect data values would be at or closer to 5 percent.

Step 8 Consult with subject matter experts

Discuss the highlighted data values with SMEs. Ask them for an opinion whether the highlighted values are errors.

Example: The SMEs review this data and note that this particular report was taken from a list of ID card applicants, all 10 years of age or older. The SMEs and the team concluded that in this age group a body weight of 301 was possible but a weight of 10 was probably an error.



Note: This particular technique is based on the assumption that the data is distributed “normally.” A normal distribution means that the data values vary evenly around some central point. The examples in figure 4-10 show such normal distributions. If the data is not distributed normally (for instance, if the data is distributed around several points or if the data values vary linearly), this rule will not yield consistent results.

(Task Three continued)

Step 9 Record the business rule/data element relationship

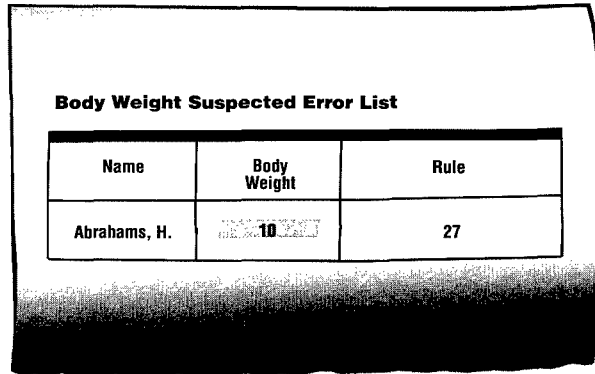
On the back of the 5"x7" card(s) found in step 1, write the number of the applicable business rule.

Example: On the back of the BODY WEIGHT 5"x7" card, the team writes "Business rule #27."

Step 10 Continue the data value error list

If the team located any new errors, it starts or continues the data error list.

Example: The new page might look like figure 4-13.



Body Weight Suspected Error List

Name	Body Weight	Rule
Abrahams, H.	10	27

Figure 4-13. BODY WEIGHT Suspected Error List

Summary

In this chapter, the team learned how to apply the knowledge it gained by creating metadata. Specifically, the team learned how to build metadata-based business rules. It developed techniques to apply the new business rules to check the quality of the data in data samples.

“I never knew there was so much I could learn from one data sample,” remarked Janine. She handed her index card file to John. “If nothing else, I have discovered how bad our data quality can be!”

“Yeah, I agree,” responded John. “Who would have ever thought we could get so far off the track from giving the customers what they wanted!”

“Don’t get down on yourselves,” said the professor. “All data systems require purging from time to time. Remember the TQM principle of ‘continuous improvement.’ We must periodically measure the goodness of our process in order to ensure the quality of our products!”

“So what’s next?” asked Joe. “More rules?”

“You’ve got it, Joe,” smiled the professor. “There’s a whole new set of business rules we haven’t dealt with yet!”

Check on Learning

Responses can be found in appendix C.

1. Distinguish between the concepts of “metadata” and “business rule.”
2. Describe how the team records a relationship between a data element and a business rule.
3. In a sample of 200 data values, how many “suspect” values would you expect to identify if you applied the “95 percent interval observed range” business rule?

Practical Exercises

Use the sample data values provided in appendix D. Answers are provided in appendix E.

1. Write a domain-based business rule for the data element called `NAME`. Highlight the data values that fail to comply with your rule.
2. Write a domain-based business rule for the data element called `ST`. (Note that appendix G may be referenced in your response.) Highlight the data fields that fail to comply with your rule.
3. A subject matter expert mentions that U.S. ZIP Codes consist of five (mandatory) digits followed by a dash and then four (optional) digits. Write one or more format-based business rules for the data element called `ZIP CODE`.
4. Calculate the 95 percent interval for the data element called `WT`. (The mean value is 168.73 and the standard deviation is 23.83.) Highlight the values that fall outside the 95 percent interval.

Chapter 5

Capturing Discovery Business Rules

A regularly scheduled team meeting was just about to kick off. Because the team had taken to the idea of displaying the products of its labor, the faded yellow walls of the conference room were now barely visible. The team had one full wall devoted to data error reports. The total number of errors now exceeded 5,000! Another wall contained lists of business rules. All together, the team had identified more than 70 rules.



"Maybe this would be a good time to just review our efforts to date," began the professor. "I am pleased to see so much enthusiasm for the DQE process. As I look around the room, I can see the 'fruits of your labor.' Clearly, this team has had a lot of success so far, but we do not want to risk losing our focus, so perhaps we ought to take a moment to review."

The professor looked to the back wall and there, now partly covered by data element diagrams, hung the team's problem statement and the definition of the customer. As the professor read the statement and definition, each team member reflected on their recent activities. A few team members realized they had wondered off the target, and while the professor read aloud, they quietly resolved to refocus their efforts.

Next, a stack of 5"x7" cards on the conference room table caught the professor's eye. He reached for the deck and holding them high, said "As we all know, this deck of cards represents a significant achievement. We have all come to know the information on these cards as 'metadata' and, further, we have all come to realize how important this thing called 'metadata' can be. Maureen, would you say the effort we expended to generate this metadata was worth it?"

Maureen responded without hesitation. "Yes. By going through the process of creating metadata, I learned a lot about how we have structured and how we use data here at DLA. I was even more impressed, though, when I learned how to use the metadata to check our data quality."

Joe could not resist interrupting Maureen. "At first I thought we were wasting a lot of time on metadata, but I have to admit that I could see the benefit once we started building and using metadata-based business rules."

The professor responded, "Okay, Joe, I am glad you are starting to see some benefit in all the work we have done so far. I think you are REALLY going to enjoy our next effort."

"Eliminating root causes, right?" blurted Joe.

"Not quite yet," cautioned the professor. "We are going to do more work with business rules."

Joe could not conceal his disappointment. His sad demeanor made Janine laugh, and that caused Joe to shoot her a stern look. "What more can we possibly do with business rules?" Joe asked with frustration in his voice.

"You will be impressed, I bet," said the professor. "Our next step is to work on 'discovery' rules."

This time, Maureen could not resist interrupting. "Why are they called 'discovery' rules?"

The professor responded "They originate as part of our work with our sample data values, combined with discussions with subject matter experts. Really, these are rules that we 'discover' along the way."

"Can you give us a quick example?" piped Janine.

"Sure," responded the professor. "Remember that we gave our data value samples to SMEs and asked them to highlight suspected data value errors?" The team members nodded. "And remember that we did not ask the SMEs to explain why they thought a data value might be an error?" Again the team nodded. "Well now, based at least in part on the knowledge we have acquired about the data because of our work on the metadata, we are ready to discuss suspected errors with SMEs. Further, we will build business rules based on what the SMEs tell us."

"I am not sure I understand yet," stated John.

"Well," continued Professor Hopkins, "suppose we are working on a sample containing data on the date that individuals applied for an identification card. Further suppose we are asking a SME to explain why a certain issue date has been highlighted. The SME explains that he highlighted that particular date because he noticed that the individual's date of birth recorded in another column

in the same record, occurred after the date of issue. In other words, the SME suspects an error because he knows that ID card issue dates always occur after birth dates. As soon as we hear that piece of news, we can write a 'discovery' business rule! There are several types of discovery rules. In this case, I think we would write a 'policy/procedure' rule. The rule might look like this." The professor turned toward the white board and wrote:

The date of issue for an ID card must occur on or after the date of birth of the individual applicant.

"I get it!" volunteered Joe. "That sounds like a powerful new capability. When can I get started?"

"Right away, Joe." The professor added, "If you think that rule is powerful, you are really going to appreciate the 'data-generating' rule."

"What's that?" Joe said, already heading for the door.

"Well, I have to run myself, but I have time to tell you that one type of discovery rule can actually be used to generate missing data values. Even when no data is missing, this rule can be used to generate a data value that will help us check the quality of other data values in the record."

"That sounds complicated," mused Janine.

"It may sound complicated, but I think you will find that discovery rules, in general, are easy to create and use. I apologize, team, but I have a class to teach this afternoon. Good luck with 'discovery' rules. I will check with you later in the week." Professor Hopkins was collecting his coat and scarf as he spoke. He was not the first one out the door however. Joe was already part way down the hall. The professor could hear him muttering "Rules, rules, rules."

Chapter 5

Capturing Discovery Business Rules





Introduction

In this chapter, the team adds a powerful new set of business rule development techniques.

As in past chapters, the team continues to use metadata and data value errors as aids in identifying business rules. The team will look at each of the errors it identified using metadata (highlighted in blue in this handbook) and will also consider the data errors highlighted by SMEs (using a gray highlighter). Highlighted values will again be the subject of discussions with SMEs, but this time, the discussions lead to the *discovery* of new business rules. Though three different kinds of discovery rules are presented as tasks 1 through 3, the rules are not intended to be worked in this or any other particular sequence. Rather, the team will let its discussions with the SMEs determine which of the discovery rules to apply.

Task 1 describes a procedure for developing business rules based on policies and procedures. Task 2 describes a powerful “if ...then...” business rule. In task 3, the team learns how to develop and use a “data-generating” business rule. The new rules tend to be exceptionally useful tools for refining data value error lists. Occasionally, the rules even help identify the correct value!

Key Concepts

-  Discovery business rule
-  Policy/procedure business rule
-  If...then...business rule
-  Data-generating business rule

.....

Task One

Discover Policy/Procedure Business Rules

The team uses a relationship between data elements mentioned by SMEs to help identify additional data quality problems.

Step 1 Consult with subject matter experts

Begin with a data sample in which a SME had highlighted a data value error. (Recall that, in chapter 2, SMEs had highlighted some errors using a gray highlighter.) Ask them to explain why the highlighted value is (or is suspected to be) an error.

Example: The team is discussing the following data value sample with SMEs (see figure 5-1). The sample is taken from a report listing all the ID cards issued over a period of time. The report is organized in columns and rows. Column headings correspond to data element names; in this case, NAME, BIRTH STATE, DATE OF BIRTH, DATE OF ISSUE, and EXPIRATION DATE. Each row in this report forms a record (a series of related data values). SMEs have highlighted two values.

Name	Birth State	DOB	Date of Issue	Expiration
Burke, T.	VA	02/22/32	03/02/42	03/01/48
Jarvis, F.W.	MA	10/30/35	03/02/42	08/01/58
Hahn, A.	VA	04/13/43	03/02/42	03/01/52
Walker, R.	VA	03/16/51	03/03/55	03/02/59
Craig, R.	VA	04/28/58	03/03/65	03/02/70
Paddock, C.	MA	04/11/67	03/03/70	03/02/75
Abrahams, H.	SC	03/15/67	03/03/70	03/02/75
Williams, P.	NY	12/05/82	03/03/83	03/02/87
Tolan, E.	VA	02/09/73	03/03/75	03/02/81
		

Figure 5-1. Data Value Sample

(Task One continued)

When asked why the data values are highlighted, the SMEs explain that in the case of Jarvis, the expiration date is probably an error because, by DOD policy, ID cards must be renewed every 6 years. The SMEs point out that the period between the date of issue and the expiration date in this record is 16 years. (This statement of policy is only intended to serve as an example of how to use this technique. This is not intended to be a statement of current or past DOD policy. All actual statements of DOD policy in this handbook will be followed immediately with a specific reference to an applicable written directive.)

When asked about the highlighted value in the Hahn record, the SMEs note that the issue date occurs before the birth date. The SMEs explain that, by procedure, an ID card is never issued before the birth of a child. This discussion with SMEs results in a class of business rules called discovery business rules.



Step 2 **Build a policy/procedure business rule**

Write a statement that expresses the observation(s) of the SME. Include the names of the data elements that the SME considered. The statement that results from this step is a policy/procedure business rule.



Example: The team notes that the SME revealed two business rules. Using data element names, the team turns the SME comment that an ID card must expire within 6 years of the date of issue into the following business rule:

The EXPIRATION DATE must be less than the ISSUE DATE plus 6 years.

The team turns the fact that ID cards are not issued before an individual is born into the following business rule:

An ISSUE DATE must be on or after a BIRTH DATE.



Note: The team should ask the SMEs to verify the statements on policy and procedure by producing written documentation.

If produced, the team should attach a substantiating statement from an official document to each policy or procedure business rule. Further, the team should review such documents, looking for other statements that can be used as a basis for business rules.

Step 3 Add the business rule to the list

Repeat the steps defined in chapter 3 for adding business rules to the team's list.

Step 4 Apply the rule to the data sample

Check all the values in the data value sample using this new business rule.



Note: The SMEs may not have noticed other instances where this type of error occurred in the data sample. Methodically apply the rule to each record and highlight (in blue) additional instances of errors.

Step 5 Add to or refine the data value error list

Start or continue the error list, adding the errors found by applying the policy/procedure rule(s).

Task Two

Discover “If...Then...” Business Rules

In reality, the activities described in this task are a continuation of the work done in chapter 4. In fact, the first four steps of this task are already complete. They are summarized here just to help explain how the team arrived at its meeting with the SMEs in step 5.

Step 1 Search the data dictionary card file

Search the data dictionary card file for selected metadata.

Step 2 Develop a business rule

Develop a metadata-based business rule.

Step 3 Add the rule to the list

Add the business rule to the team’s list.

Step 4 Apply the rule to the data sample

Apply the business rule to the data value sample and highlight all values that do not comply with the rule.

Step 5 Consult with subject matter experts

Discuss the data values that the team has highlighted with SMEs. Ask them for an opinion whether the data values are errors. After they determine whether a highlighted value is or is not an error, ask the SMEs to explain how they made their determination.

Example: The team is discussing the following data value sample with a SME (see figure 5-2). A few BODY WEIGHT data values have been highlighted using the techniques described in task 3, chapter 4. For the purposes of this example, the team is reviewing data from a medical file (not identification card data as in earlier examples).

In this instance, the SME reports that the highlighted body weight is probably not a data value error. By asking the SME to explain, the team learns that the SME referred to the data value for age and noted a value of 0. The SME explains that this value indicates that the individual is less than 1 year old and that she believes that for ages 5 years and under, a body weight of less than 44.6 pounds is reasonable.

Name	Body Weight	Age
Burke, T.	150	55
Jarvis, F.W.	141	23
Hahn, A.	115	14
Walker, R.	205	32
Craig, R.	266	44
Paddock, C.	155	62
Abrahams, H.	10	0
Williams, P.	160	40
Tolan, E.	175	39
	164	29

Figure 5-2. BODY WEIGHT Data Sample

Step 6 Build an “if...then...” business rule



Restate the SME’s observation as an “if...then...” rule, citing the data elements that the SMEs used to reach their conclusion.

Example: If AGE is less than 5, then BODY WEIGHT may be less than 44.6 pounds.

Step 7 Add the business rule to the list

Repeat the steps defined in chapter 3 for adding a business rule to the team's list.

Step 8 Apply the rule to the data sample

Check all the values in the data value sample using this new business rule.



Note: The SME may not have noticed other instances where this situation occurred in the data value sample. Methodically apply the rule to each record and note additional instances of this situation.



Tip: Generally, the team will require specialized programming skills if it decides to automate the process of checking a data sample using an “if...then...” business rule. However, a number of specialized CASE tools have been developed to simplify the process of automating this kind of business rule.

Step 9 Add to or refine the data value error list

Start or continue the error list, adding the errors found/validated by applying the “if...then...” business rule(s).



Note: In this particular example, the business rule helped the team to *reduce* the number of suspected errors on its list.

Task Three

Discover “Data-Generating” Business Rules

In the discussions with SMEs, the team discovers that, in certain situations, a SME can predict a data value by looking at other data values in a record. This fact can be turned into a data-generating business rule. This new family of rules is unique in that it serves two purposes. First, the rule can be used on records where data is missing to help SMEs generate missing information. Second, the rule can be applied where all data values are present to validate existing data (or to discover new errors).

In reality, the activities described in this task are a continuation of the work done in chapter 4. In fact, the first four steps of this task are already complete. They are summarized here just to help explain how the team arrived at its meeting with the SMEs in step 5.

Step 1 Search the data dictionary card file

Search the data dictionary card file for selected metadata.

Step 2 Develop a business rule

Develop a metadata-based business rule.

Step 3 Add the rule to the list

Add the business rule to the team’s list.

Step 4 Apply the rule to the data sample

Apply the business rule to the data value sample and highlight all values that do not comply with the rule.

Step 5 Consult with subject matter experts

Discuss the highlighted data values with SMEs. Ask them for an opinion whether the data values are errors. After they determine whether a highlighted value is or is not an error, ask the SMEs to explain how they made their determination.

Example: The team is discussing the following data value sample with SMEs (see figure 5-3). This is a sample from a file containing military equipment dimensions. Because all pieces of equipment are required to have dimensions, the team has already applied the business rule requiring data values (blank, zero, or null values are not valid). The sample includes an equipment type column and columns for equipment length, width, and height (all in inches) and cubic volume (in cubic feet).

Type	Length	Width	Height	
Truck	360		120	21
Jeep	96	70	55	214
Trailer	75	65	70	1975
Tank	180	110	80	9170
Forklift	84	64		243
Box	24		18	3
Crate	48	24	18	12

Figure 5-3. Military Equipment Dimensions

The SMEs report that all the highlighted values are errors. By asking the SMEs to explain, the team learns that all equipment on this list has a measurable length, width, and height. Further, the SMEs point out that the missing dimensions must exist (and they could be determined using just the data in this report)! The SMEs explain that data on the cubic measurement stored in cubic feet, while length, width, and height are stored in inches. The cubic measurement should equal the length multiplied by the width and the height and then divided by

1,728 (the number of cubic inches in a cubic foot). Using this relationship, a missing data value in a record can be calculated by using the other values in the record.

Step 6 Build a data-generating rule



Capture the relationship that the SMEs expressed in the form of a business rule. Include the data element names in the rule. The statement that results from this step is a data-generating business rule.

Example: The data value interrelationship expressed by the SMEs could be written:

$$(\text{LENGTH} \times \text{WIDTH} \times \text{HEIGHT})/1728 = \text{CUBE}$$

Step 7 Apply the rule to re-create missing data values

For records with one data value missing, apply the rule by substituting known data values for LENGTH, WIDTH, HEIGHT, and CUBE. Use algebra to solve this equation for the missing value.

Example: A truck WIDTH data value is missing. The team generates a probable value using the data-generating rule.

$$(\text{LENGTH} \times \text{WIDTH} \times \text{HEIGHT})/1728 = \text{CUBE}$$

$$(360 \times \text{WIDTH} \times 120)/1728 = 2100$$

$$\text{WIDTH} = 84$$

Step 8 Apply the rule to the data sample

For the records where all the data values are present, use this rule to calculate a data value for the right side of the equation using the data values for the elements on the left side of the

(Task Three continued)

equation. Create a new column called "Calculated Cube" in the sample data value. Record the newly calculated data in the new column.



Tip: Generally, the team will require specialized programming skills if it decides to automate the process of checking a data sample using a data-generating business rule. However, a number of automated tools have been developed to simplify the process of automating this kind of business rule.

Example: The team applies the rule from step 6 to all the records containing a value for LENGTH, WIDTH, HEIGHT, and CUBE in this sample. The team calculates a new value for cube for each record by multiplying the LENGTH, WIDTH, and HEIGHT values and dividing by 1728. The team writes the newly calculated value for CUBE in a new column on the sample data value report (see figure 5-4).

Calculated Data Values					
Type	Length	Width	Height	Cube	
Truck	360		120	2100	
Jeep	96	70	55	214	214
Trailer	75	65	70	1975	198
Tank	180	110	80	9170	917
Forklift	84	64		243	
Box	24		18	3	
Crate	48	24	18	12	12
...	

Figure 5-4. Calculated Data Values

Step 9 Highlight the newly identified discrepancies

Compare the calculated value to the value in the record. Highlight the value in the record when it does not agree with the calculated value (the team may choose to ignore small differences).

Example: Here, the cube values for the Trailer and the Tank would be highlighted because they do not agree with the calculated values.



Note: At this point, the team does not attempt to determine the cause or source of the discrepancy.

Step 10 Add the business rule to the list

Repeat the steps defined in chapter 3 for adding a business rule to the team's list.

Step 11 Add to or refine the data value error list

Start or continue the error list, adding the errors found by applying the data-generating business rule(s).

Summary

The team learned how to develop and use three new types of business rules. First it learned how to build and use a rule based on policies and procedures. Next, it discovered how to build and use an “if...then ...” business rule. Finally, the team learned how a data-generating rule can be used to re-create missing or inaccurate data values.

“Okay, okay,” John said with enthusiasm as he typed the last character for his report. “I tell you all, if this list of business rules doesn’t make the colonel’s eyes water, nothing will!”

“I know one thing,” said Joe. “My eyes are watering! That last task was pretty intensive research, wouldn’t you say?”

Maureen rolled her eyes. “If it were up to me I’d say there should be a business rule for selecting the type of person you have to work with around here!”

“Hey, easy now!” laughed the professor. “I’ve got an ego, too, you know!”

“Sorry,” Maureen said.

The professor continued reviewing one of the team’s marked-up data samples. “Hmnn...” he said, more to himself than to the others. “I think we are moving on rather well here!” With a slightly louder voice, he said, “Gather round, DQEs! Let me show you something!”

The team quickly assembled around the professor. They leaned over to see what their mentor had discovered.

“Here,” the professor said pointing to a line entry in the report. “Do you see anything unusual?”

“Hey, it looks as if many of the records that lack dimensional data come from a foreign supplier!” exclaimed Joe. “Am I right?” He looked around anxiously at the group.

“Maybe,” responded the professor. “We’ve got a few more steps before we can be sure, but you do have evidence in that direction. Let’s wait and see if our suspected trend is present in a larger database. If it is, we may have made a key observation!”

“All right!” exclaimed John for the second time that afternoon. “We’re getting there! I love it!”

“Not so fast, John,” cautioned the professor with a smile. “Remember, business rules help us find and sort through our data value errors—but they didn’t often confirm the source of the errors. That challenge still lies ahead!”

“Well then, let’s get on with it!” John motioned to the computer.

Check on Learning

Responses can be found in appendix C.

1. Briefly describe the technique the team uses to create a policy/procedure business rule.
2. Briefly describe two ways to use data-generating rules.
3. Explain how the use of discovery business rules may result in changes to the team’s list of data value errors.

Practical Exercises

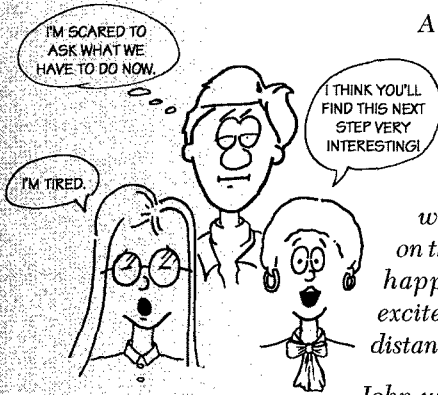
Use the sample data values provided in appendix D. Answers are provided in appendix E.

1. A subject matter expert mentions the mandatory retirement age for the organization is 60. Use this information to build and apply a policy/procedure business rule. Highlight the data values that fail to comply with your rule.

-
2. Create a data-generating business rule for the data elements called AGE and DOB. Highlight the data values that fail to comply with your rule. Use your rule to calculate the missing value for the data element called AGE in record 22. (Assume this data sample was created 1 June 1995.)

Chapter 6

Applying Business Rules to a Large Database



A crowd had formed around the shiny red 4x4 sitting in the DLA employee parking lot. Janine, the proud new owner, stood in the center of a ring of friends and co-workers, trying to answer several questions at once. John was standing in the shade of an awning on this particularly hot summer day. He was happy enough to watch the flurry of excitement over Janine's new jeep from a distance.

John used the moment to reflect on the events of the previous few days. He realized he was particularly pleased to see the team take a break. "They have all contributed 110 percent," he thought to himself, "to the task of reviewing the stacks of database reports that still cover the conference room table." Earlier, John remembered, the team had written out a list of about 90 business rules. Following that, it had issued its database report requests. John grinned as he reflected on the expressions on everyone's face when the database report printouts were first delivered!

Lost in his thoughts, John was startled to find he was now sharing the shade of the awning with Maureen. "Feels good to let your eyes rest," she stated. John recovered his composure, shook his head in agreement. Before he could say anything, Maureen asked, "Mind telling me what is amusing you? I thought you were grinning."

John was a little embarrassed to realize that Maureen was so good at reading his feelings. He answered honestly. "I was just thinking about the 'thunk' noise that the thick database report printouts made when they were dropped onto the conference room table. Did you see the look on Janine's face?"

While both laughed aloud, Joe quietly walked within earshot.

After a few moments, Maureen observed, "I guess we have all had a few surprises. I know I've been surprised twice. The first surprise was over how much work is required for an individual analyst to sit and apply the business rules to the

database. But the second surprise seemed to help justify the first." She asked John, "Were you as shocked as I was at the number of errors we found?"

John certainly shared that sentiment. In fact, he found he was anxious to tell someone about it. He and Maureen spent the next few minutes, comparing results and discussing their new ability to find data value errors and to assess data quality. They discussed the advantages of measuring data quality in a consistent way and how their first measurement, called the "data quality baseline" might be used by the process owner.

Soon, Joe interrupted with one of his own surprise observations. "What amazed me," Joe said, anxious to share his viewpoint with Maureen and John, "was how much time and energy we saved when we switched over to the use of an automated tool."

Maureen's eyes lit up. "I agree," she chimed. "That was amazing. By then, we all understood the process and so we could see clearly what the tool could do for us. What a relief to have an automated tool apply each rule to the 'soft-copy' of our database."

"Automated tools offer several advantages," agreed John. "You know, the tool had to apply the rules more consistently than we did, but what really impressed me was the tool's ability to generate the printed error list. After manually checking a database and then writing out an error list by hand, I really was excited to see how a tool could be used to create and print the error list for me."

Just then, the professor drove up and started to get out of his car. The "Data Quality Engineers" reacted like children in a school yard at the end of recess. Noting the presence of the professor, the team gradually broke off all discussions and quietly moved toward the door to the DLA building.

As they filed through the doors leading toward their conference room, Joe mumbled to no one in particular. "Back to the sweat shop." John overheard him and whispered, "Back to the 'no-sweat' shop I'd say. I can't wait to see how many data value errors we can capture and document combining our process with the use of automated tools."

Joe couldn't help but agree.

Chapter 6

Applying Business Rules to a Large Database





Introduction



Up to this point, the team has been working with relatively small data samples from a single database. It has used the samples to develop and also to refine its metadata. Using its newly acquired knowledge about metadata, the team then created and tested a list of business rules, again using data value samples.

In this chapter, the team applies its business rules to much larger quantities of data. Ideally, the team will apply the business rules to the entire database. However, time and the available resources may limit the team's capability to achieve this goal, particularly if the team does not have access to automated support. Should it need to limit the quantity of data that it reviews, the team should follow the guidelines on selecting limiting criteria that are contained in this chapter. The chapter is written as though no automated support is available because with or without support, the procedure is essentially the same.

In chapter 6, the team obtains the full database file in printed form or, if that is not practical, a significantly large portion of the data (task 1); applies the business rules (task 2); and organizes the results by creating a data quality baseline (task 3).

Key Concepts

-  Database value report
-  Limiting criteria
-  Database report log
-  Specific business rule list

-
-  Data quality metric
 -  Data quality baseline

Task One

Obtain Database Records

In this task, the team seeks access to all, or a significantly large portion of the records in the database. The team will use its knowledge of the data and its exposure to the data quality in the sample it has seen to date as a basis for making its data request.

Step 1 Prepare a database report request

Review the data error list and select from one to five “target” data element(s). Initially, select the data element(s) that appears most often on the error list. Next, select additional data elements as needed to ensure that the records in the printed report on the target data element can be identified. Finally, pull the 5”x7” card on the target data element and review the applicable business rules (recorded by number on the back of the card). Based on the data element relationships discovered by the team as it developed business rules, select additional data elements to be included in this report. Do not pick more data elements than will fit across a single line of a printed text in a report. Leave some space on the paper for making notes beside each record in the file. (Check with SMEs to determine the limits on the length of a printed line.)

Example: The team reviews its data value error list and notes that LENGTH, WIDTH, and HEIGHT data values appear frequently. It decides to construct a “Dimensional Data” report and to make these data elements the target. The team decides to include EQUIPMENT NAME and providing organization (PROV ORG) so that it can identify the records in its report (in this case, to associate the dimensional data values in the report with a given piece of equipment). Next, the team pulls its 5”x7” cards for LENGTH, WIDTH, and HEIGHT. It notes that one business rule compares CUBE to a calculation using all three of these elements. The team decides to include CUBE in the report. Finally, the team checks with SMEs and finds that the data it will be requesting will fit on a printed page, with room to spare for notes.



Note: This data value report request does not need to be designed from scratch. In fact, the team is likely to find that the use of existing reports, with small modifications if required, will reduce confusion and improve response time by SMEs. In fact, the team may elect to use one or all of the report formats it has already evaluated as data value samples.

Step 2 Repeat for other target data elements

Repeat step 1, selecting new target data elements. Repeat until every data element on the error list has been included in at least one database report request.



Note: A separate database report request for each data element on the data error list may place unrealistic demands on SMEs and on system capabilities. To minimize delays and to keep new workloads on SMEs to a minimum, the team should strive to construct multipurpose database report requests (a multipurpose report is one that targets two or more data elements, as described in the example for step 1). As already mentioned, the team should always consider using a preexisting report format.

Step 3 Review database report requests with subject matter experts

Provide a written data value report request(s) to SMEs. Solicit their opinion as to the time and costs required to produce the report(s). For a large database, consider limiting the number of requested records using some criterion.



Example: The SMEs review the team's Dimensional Data report request and note that the database files contain dimensional data on more than 2 million line items. A printed report will take days to produce. The SME suggests that the report be limited to the line items selected several months earlier for an inspection team. The SME notes that this list provides over

60,000 line items from a good cross-section of the inventory. Further, this item list contains the data records most often accessed by the functional community. This choice of records is fixed (the same set of records can be retrieved from the database at any future date, as required). The team concurs with the SMEs' suggestion and decides to limit its request.



Note: The choice of a limiting selection criterion varies with the nature of the data and the size of the database. The choice, however, always requires careful consideration. To stimulate discussions with SMEs, a few possible selection criteria are described in figure 6-1.

Criterion	Example	Cross Section	Yield	Consistency	Comments
Interval	Every fourth record in a file	Probably good	Predictable (25%). Can be tailored (every 100th record yields 1%, etc.)	A small change in the file changes the resulting record selection.	May require special programming to generate.
Data value	LAST NAME starting with "S"	Probably poor	Not easily predicted. Varies with choice of letter.	Repeatable with same letter selection.	Not recommended.
Data value	STOCK NUMBER ending in "3"	Probably good	Predictable (10%). Can be tailored (stock numbers ending in "3" and "8" yields 20%, etc.)	Repeatable. Yields an identical or nearly identical record set.	May require special programming to generate.
Time frame	ISSUE DATE between 1-31 May 95	Probably good	Predictable. Varies by months stored in database file.	Yields identical record set for same month.	Usually easy to implement.
Random	Use random number generator to select records	Excellent	Predictable. Can be tailored.	Yields consistent results but not identical record sets.	Specialized skills needed to develop random selection technique and to apply results to full database.
User-defined	Existing inspection report	Varies	Varies	Varies	Readily available Easy to implement. SME analyses usually already complete. May not be useful if looking at two or more systems.

Figure 6-1. Possible Record Selection Criteria

Step 4 Create a database report log



Record the time that the data values were generated, the originating database, the data limiting criterion (if applicable), and the period covered by the data (if applicable).

Example: The team begins a log of data value reports as shown in figure 6-2.



Note: The data quality in the reports contained in this log forms a baseline. Later, the team will use the information in the log to make identically configured requests for new reports (in other words, after a period of time, the team will request exactly the same set of records, in exactly the same format). By repeating the business rule checks on the second (and subsequent) reports and then comparing the quality in each report, the team will be able to assess its progress toward achieving its data quality goals.

Log#	Report Title	Source System	Data Limiting Criteria	Creation Date	Time Frame
001	Dimensional Data	Inventory Management System	Inspection Team Report	03/02/95	Not applicable
002					
003					

Figure 6-2. Database Report Log

Task Two

Apply the Business Rules

In this task, the team carries out the often tedious task of applying business rules to a large number of records in one or more database reports. (If more than one report applies, simply repeat this task for each report.)

Step 1 **Select the applicable business rules**

Review the 5"x7" card on each of the selected data elements in the database report. List the business rules that apply to these elements by referring to the numbers on the back of the applicable 5"x7" cards.

Example: The team begins to work on its Dimensional Data report. It reviews the 5"x7" cards on the EQUIPMENT NAME, PROV ORG (providing organization), LENGTH, WIDTH, HEIGHT, and CUBE data elements and notes that business rules #15 and #44 apply.

Step 2 **Write out a specific business rule list**



Write a specific list of business rules to be applied to the database report. Note the log number for the database report (created in task 1) on the business rule list.

Example: Some business rules were created using a "fill-in-the-blank" technique. To avoid confusion now, the team writes out the complete list of business rules for each data element in this data value report, noting the report log number and filling in all the blank spaces the business rules with data element names. The first time a rule is used, add a "-1," the next time, use a "-2," etc. as shown in figure 6-3.

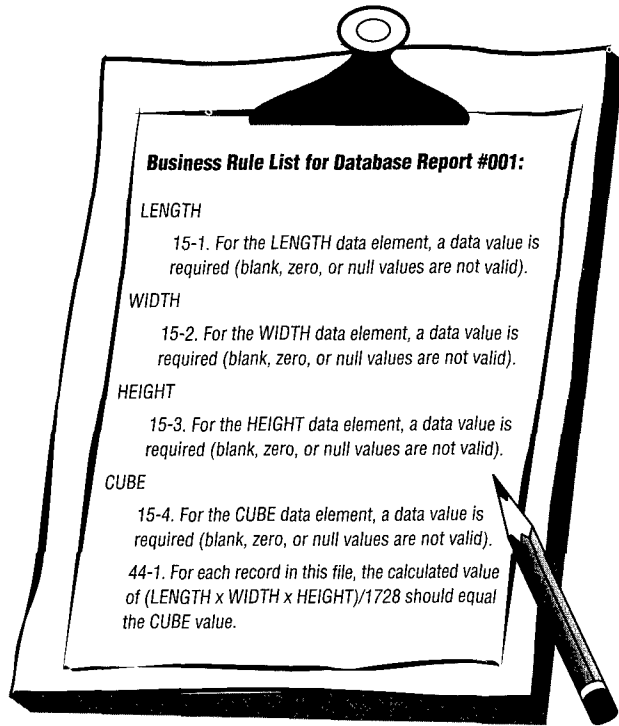


Figure 6-3. Business Rule List



Note: Writing a specific business rule list and associating that list with a database report request serves two purposes. First, in a typical project, the team will be handling large amounts of data and hundreds of rules. The use of lists and logs will help the team stay focused and organized. The second purpose is to set the stage for later data quality checks. As noted in task 1, the team will be repeating the process of using the rules to check data value quality at some time in the future. The team's goal will be to repeat all future data quality checks using exactly the same database report request *and* exactly the same business rules set.

Step 3 Apply the business rules to the data value report

Work with one rule at a time and work through the records one line at a time (to reduce the potential for human error). Highlight each data value that fails to comply with the rule (recall that team members use blue markers). Each time a value is highlighted, write the business rule number at the end of the record.

Example: One team member selects rule #15 for the LENGTH data element and, one record at a time, highlights the LENGTH field each time that field is empty (that is, each time the field contains no value or the value "0"). Each time the LENGTH field is highlighted, the team member writes "#15-1" at the end of the record.

A second team member selects rule #15 for the CUBE data element and, one record at a time, highlights the CUBE field each time that the field is empty (that is, when the field contains no value). Each time the CUBE field is highlighted, the team member writes "#15-4" at the end of the record.

A third team member selects rule #44 and, for every record in the file containing a data value for LENGTH, WIDTH, HEIGHT, and CUBE data elements, calculates a cube value and compares it to the stored value. The team member highlights the stored cube value when it does not agree with the calculated value. Each time a value is highlighted, the team member writes "#44-1" at the end of the record containing the value.



Tip: Automated support can reduce the workload involved in this step dramatically. In addition, an automated tool will almost certainly improve the consistency and reliability of the output. Automated tools vary in sophistication from tailored database queries and other database software programs to dedicated data quality software products. The choice of tools depends on the availability of software support and the training/experience of team members.



Note: It is possible for a single data value to fail to pass two or more rules. When this occurs, the value is highlighted normally (when it first fails a business rule). When a team member using a different business rule notes a problem with a value that has already been highlighted, the team member simply writes the applicable business rule number after the number of the business rule already recorded at the end of the record (a data value error is highlighted only once, but a record may violate any number of business rules).

Task Three

Organize and Report the Results

Having completed the process of applying the business rules to the database, the team begins to organize the results of its labor.

Step 1 Group results by data element



For each data element in the database report, count the times that a data value was *not* highlighted. Convert this “raw score” into a data quality metric using the following formula:

$$\left(\frac{\text{Number of times a value was not highlighted}}{\text{total number of records in the report}} \right) \times 100$$

Create a table showing the data element name, the number of times the data element was not highlighted (the raw score), and the data quality metric.

Example: The team reviews the Dimensional Data report and, working with one column at a time, counts the nonhighlighted data values for PROV ORG, LENGTH, WIDTH, HEIGHT, and CUBE. The count becomes the raw score for each data element. The team then calculates the data quality metric based on its report of 60,000 records, using the above formula. The team creates the table shown in figure 6-4.




Tip: Dedicated, data quality software programs are available on the commercial market. These software programs provide the capability to accept large database files, apply a wide variety of business rules, and then sort, print, summarize, and even graph the resulting output.

Data Quality Metric Table

Data Element	Raw Score	Data Quality Metric
PROV ORG	55,200	92%
LENGTH	45,603	76%
WIDTH	45,244	75%
HEIGHT	45,988	77%
CUBE	37,051	62%

Figure 6-4. Data Quality Metric Table

Step 2 Display the data quality baseline

 For each database report, create a bar graph. Create a bar for each data element in the report using the data quality metric (that is, showing the percent of the data values that complied with all the business rules).



Tip: See the tip for task 3, step 1.

Example: The team creates the bar graph shown in figure 6-5 for the Dimensional Data report using the data from figure 6-4.

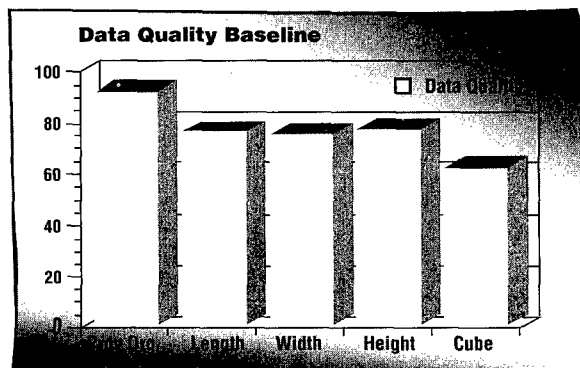


Figure 6-5. Data Quality Baseline

Step 3 Present the baseline to the process owner

Show the data quality baseline to the process owner. Solicit additional guidance.

Example: The team develops a data quality baseline for each database report and briefs the process owner. The process owner sees the benefit in having a quantified data quality assessment. At the same time, the process owner estimates the resources required to resolve all of the data quality problems immediately and realizes that this approach is not feasible. However, the process owner also realizes that by combining the new knowledge about data quality with the functional importance of the data, he or she can establish achievable priorities. Further, if this measurement of data quality is repeatable, then it can serve again later as a measure the team's progress toward achieving data quality objectives. After considering the team's report on this and dozens of other data elements, the process owner tells the team that dimensional data is critically important to the organization. The process owner establishes a data quality goal of 95 percent for dimensional data and requests a second data quality assessment in this area in 2 months.

Summary

Having previously tested the business rules on a relatively small sample of data, the team expanded its assessment of data quality into large quantities of data. The team first prepared a database report request and then applied a list of prewritten business rules to it. Subsequently, the team organized and reported its results.

“Honestly, “Maureen said as she smoothed down the pleat that had formed in her skirt fabric, “I sometimes think that the more we look at this data the worse it becomes! You don’t think we might be seeing things that aren’t there, do you?”

She turned around to look at John who was absently brushing away chalk dust from the nearby chalk tray.

“Well, I sure hope not!” he responded looking pointedly at her. “We have enough trouble in this program without borrowing more!”

The professor, who had been studying their database report walked over and joined Joe and Maureen. “You only can see what is there, trust me. Some data files are so badly flawed that there is no hope of ever cleaning them up. In this case, however, I think there are some trends that can be reversed.”

“Really?” asked Joe with noted enthusiasm. “Do you really think we basically have a sound program?”

“I don’t think you’re in bad shape,” said the professor. “Look here, for example.” The professor raised the data quality baseline report. “With high raw scores in the larger data sample, I have an idea that you’ll not have too big of a problem in finding the root causes.”

“You really mean it?” asked John as he looked at the recorded scores. “We really aren’t so bad off? Wait, though, look at this.” He moved his finger from the where the professor had originally pointed to another place at the bottom of the page. “Now, this score is pretty low, wouldn’t you say?”

“Yes, it is, John,” said the professor with a nod. “And so are these two. Oops, here’s another low score! We better move to the next step in the process and see why these are showing up as they do!”

“What step is that?” Janine asked.

Crouching over like an ogre, Joe gestured for Janine to come closer. “Step into my computer room and let me show you... heh, heh!”

Check on Learning

Responses can be found in appendix C.

1. How does a team prepare a database report request?
2. Why is the data quality baseline an important tool in DQE?
3. What sort of guidance might the process owner be expected to give as a result of reviewing the quality baseline?

Chapter 7

Working with Multiple Database Systems

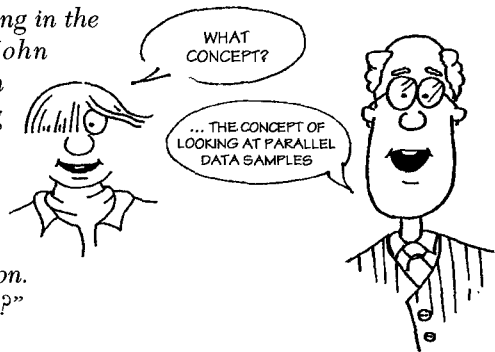
It was drizzling as Joe hustled across the DLA parking lot toward the brick complex housing the DLA workforce. He moved quickly for two reasons. For one thing, the dark, Monday morning sky seemed ready to let go with a downpour. The other reason, though, was that Joe felt particularly anxious to resume the DQE effort the team started during the past week. He could not believe that they were already working with a second automated system. Joe was reasonably confident that, with a little determination, they would be able to complete the process in another week or two.

A heavy rain began just as he came within a few paces of the double glass doors marking the entrance to DLA. In spite of his dart for the door, Joe ended up with a damp coat and wet shoes. Just inside the door, he bumped into John.

While they worked to dry off, John asked Joe, "How are we doing at developing metadata for the Executive Information System?"

Joe, busy stomping his feet to dry his shoes, replied, "I think we are on track. We were able to gather the metadata quickly because the team has become... well, I guess you could say...skilled data quality engineers. Maybe because we just finished our work with the source-level system, we all knew just what to do." Joe continued as they walked down the hall toward the now all-too-familiar conference room. "Maybe even more important, we knew exactly what kind of data sample to request because we were already familiar with the source system's data." Joe was finishing his thought just as the two entered the room. "In fact, we structured our requests for data from the new system to match the data we had already analyzed in the source system."

The professor happened to be sitting in the conference room as Joe and John entered. He too had been caught in the morning shower. He was wiping a few remaining rain drops off his forehead with a handkerchief. He overheard Joe's comment and could not resist using that information to pose a "quiz" question. "What is the term for that concept?"



"What concept?" asked Joe. He really understood the question; he had fired off his own question in order to stall for time to think.

"The concept of looking at parallel or matching data samples when conducting DQE on two or more systems," the professor shot back.

Maureen, who had been sitting quietly at the conference room table, reviewing a database report, looked up and waited a moment for Joe to answer. When she noticed that he had developed a sudden, apparent long-term fascination with the tile design on the ceiling, she decided to risk a response herself. "I think the concept is called the 'vertical slice.'" As the professor turned to face her, she continued confidently, "For instance, a source system usually stores and manipulates data at a detailed level. Usually, that data is either aggregated or changed in some other way when it moves from the source to an executive-level system. In my case, when I requested source system data samples, I happened to have been given a sample containing all the transactions from the month of May of last year. Later, when I formulated my data sample request for the executive-level system, I asked for a sample for the month of May. If this works, I will be able to compare my analysis from the source system data to the analysis of the executive-level system. In other words, I will be working with a 'vertical slice' of data."

Joe was relieved to see that Maureen had diverted the attention away from him. He found an empty chair, sat down, and tried to look inconspicuous.

The professor agreed with Maureen, adding, "I think you described the concept of the 'vertical slice' very well. I might add that the concept applies equally to data samples and, later, to work with the larger database reports."

Rather than risk getting another tough question to answer, Joe decided to give the professor a question of his own. "Since we are talking about concepts, can you explain the concept of the 'data linkage?'"

"Sure, Joe," the professor answered. "A linkage, or specifically a 'data linkage report,' establishes a relationship between data elements in different systems. For instance, suppose we develop a data element definition in our source system for an element named DOB. Let's say the definition is 'the calendar date on which a person was born.' Now, suppose that later, we find a data element named BIRTH DATE in our executive-level system. Though the data element names are different, our methodology calls for us to compare definitions. In this case, we will probably find the definitions are nearly identical. After we compare data values and then confirm this finding with subject matter experts, we would record

the association of DOB in the source system to the BIRTH DATE data element in the executive-level system by adding this information to our data linkage..."

Just then, Janine burst into the conference room. She was soaked to the skin. "What a storm!" she exclaimed. "I ran from the car to the building and still got drenched! Oh, and while I was driving, I hit some high water on the road near my house. I could not believe how well my new 4x4 handles water on the road!" Janine suddenly realized she had interrupted the professor in mid-sentence. She mumbled an apology.

"That's okay," offered Joe. "The professor was explaining the data linkage report. I think I understand the concept now, but, Janine, I might like to know more about how to handle a 4x4 in high water!"

To a chorus of laughter, Janine answered "Joe, that may be a concept that is just 'too deep' for you!"

Chapter 7

Working with Multiple Database Systems

Introduction

In chapter 6, the team presented the results of its efforts, in the form of a data quality baseline, to the process owner. The process owner considered the team's report and estimated the resource commitment needed to solve all the problems. Usually, the amount of work that is required will exceed the available resources. Before the use of DQE, the process owner had no consistent measure of either the raw number of errors or of the relative data quality (error counts by data element). This time, that kind of information is available. The process owner can now weigh the functional importance of the data against relative data quality. Based on these criteria, he or she can establish priorities. The process owner may want to revise the team's charter to reflect the new priorities and then request another data quality assessment after a reasonable period.




Given a list of priority data elements, the team's first action is to determine where or how the data values originated. The team makes this determination by referring to the data element diagrams created in chapter 2. In general, data values are created in one of three ways. They can be entered by an individual using a keyboard, mouse, or other "input" device. They can be derived by manipulating other data. Derived data results when automated processes (like counting, adding, subtracting, multiplying, etc.) are applied to existing data values to generate new data (derived data is sometimes called "roll-up" or "summary" data). Finally, data values may have been created in some external system and then transferred, electronically or manually, into the current system.

If the data originated through either of the first two techniques, then the team is working with a "source" system for the data elements in question. The team should skip to chapter 8 and begin work on finding and eliminating root causes. If the team concludes that an external system is the source for its data elements, then it begins a two-step process. First, the team

repeats the DQE procedures described in chapters 2 through 6 for the external system. The team activities include building metadata files, creating business rules, applying business rules to database values, capturing data errors, and measuring data quality. The second step is to build data element linkages. A linkage establishes a relationship between a data element in one database system and a data element in another system.

The key to conducting the Data Quality Engineering process on two or more systems is to use synchronized data value sets. In other words, the database value reports that the team used for the work described in chapter 6 for the source system should match, or be synchronized with, the requested data value samples for the destination system. This concept will be explained in more detail in task 3. First, however, the team develops data flow diagrams (task 1) and develops data linkages (task 2).

Key Concepts

-  Data Flow Diagram
-  Data Linkages Report
-  Vertical Slice

Task One

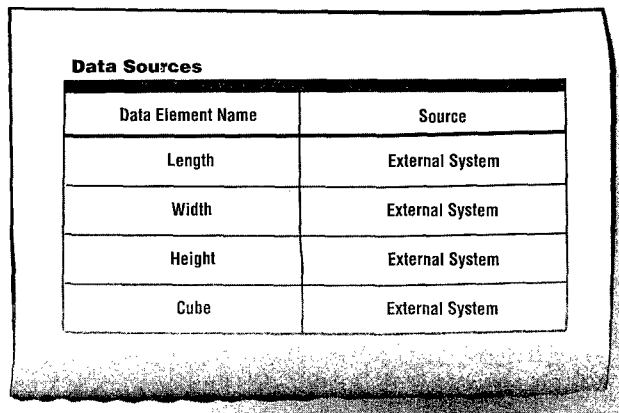
Develop a Data Flow Diagram

In this task, the team will build a data flow diagram for its priority data elements by combining its knowledge of the data in the destination system with the knowledge of SMEs.

Step 1 List the data element sources

List the priority data elements by data element name. Refer to the data element diagrams created in chapter 2 and, for each priority data element, determine the data source. If the source is “data entry” or “derived,” skip to chapter 8. If the source is “external system,” go to step 2.

Example: As shown in figure 7-1, the team determined that “dimensional data” included the LENGTH, WIDTH, HEIGHT, and CUBE data elements. It listed the elements by name and then, referring to the data element diagrams created for these elements, noted that the source was listed as “external system.”



Data Sources	
Data Element Name	Source
Length	External System
Width	External System
Height	External System
Cube	External System

Figure 7-1. Data Sources

Step 2 Determine the source system name

Meet with SMEs and ask them to identify the external system used as a source for each data element on the list. Review data samples if necessary to stimulate thought and discussion. If SMEs identify more than one source system (or file) for a data element, ask them to explain how they can determine which source was used for a given data value.

Example: The team meets with SMEs and asks them to identify the external system(s) used as a source for the data elements on the list. A SME notes that the dimensional data source varies with each providing organization. In fact, the SME continues, the data values stored in this file come from the Medical Battalion, the Fuels Unit, and the Motor Transport Company. The team asks the SME to look at a data value sample and to help identify those sources. The team notes that the SME uses the PROV ORG codes to identify data from the Medical Battalion Inventory Management System (code MB), Fuels Unit Equipment System (code FU), and Motor Transport Company Equipment Control System (code MT) respectively.

Step 3 Create a data flow diagram



Draw a data flow diagram by using boxes to represent source system(s) and the destination system. Label the boxes with the system names. Use arrows to show how the data moves from the source systems to the destination system. Write the data element names on or near the arrows.

Example: A team member draws a box for the Equipment Description System on the right side of the white board in the team conference room. Along the left side, the team member draws three boxes and labels them as shown in figure 7-2. The team member uses arrows to show the data flowing from the three boxes on the left to the single box on the right. Near the arrow, the team member writes the data element names.

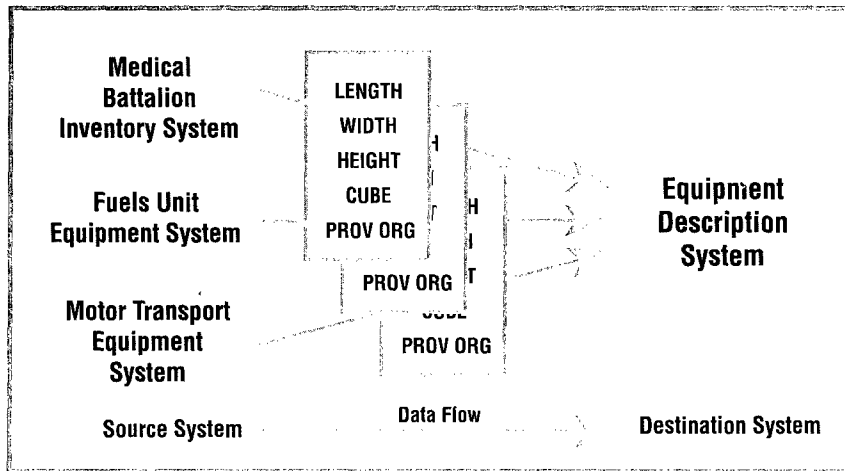


Figure 7-2. Data Flow Diagram

Task Two

Establish Data Linkages

In this task, the team's focus shifts from the database system on which it recently completed the DQE process (this system is now called the "destination" system). The new focus will be the system(s) identified as data sources. For the newly designated source system(s), the team simply repeats the DQE procedures outlined in chapters 2 through 5. Now, however, the priority data elements from the destination system are regarded as the "problem area."

Step 1 **Build a data element diagram**

Using the procedures from chapter 2, build a data element diagram for each priority data element. For the "Component" (or "Output") column, use the name of the destination system. Refer to chapter 2 for additional detail and examples.



Note: The team is likely to discover that more than one data element in the source system(s) seems to relate in some way to the data element name in the destination system (that is, the team does not have enough information about the source system, at this point, to pin down data element relationships). Until it can build metadata files for the source system, the team will consider any apparently related data element as part of the "problem area."

Step 2 **Create data dictionary cards on source system data elements**

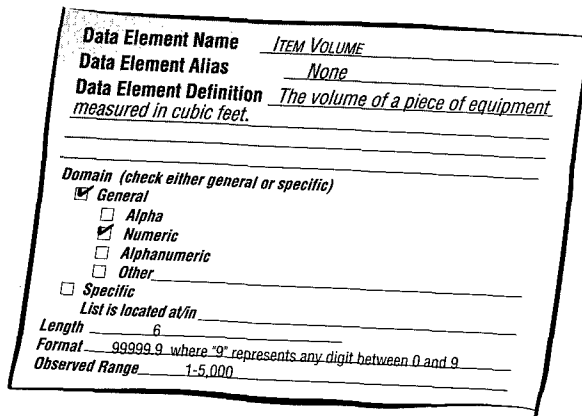
Use the procedures described in chapters 3, 4, and 5 to develop and refine metadata for all the data elements in the problem area. (Refer to chapters 3, 4, and 5 for additional details and examples.)

Step 3

Compare the source and destination system data dictionary cards

Compare the source system data dictionary cards to those from the destination system. Using the data element definitions as a key indicator, but also referring to the data element name and the other metadata, identify similarities in the data element descriptions. Create a report called "Data Element Linkages" and record the names and the system of data elements that are related.

Example: While working with the Medical Battalion Inventory System (one of the source systems), the team develops a data dictionary card for a data element called ITEM VOLUME (see figure 7-3).



Data Element Name ITEM VOLUME
Data Element Alias None
Data Element Definition The volume of a piece of equipment measured in cubic feet.

Domain (check either general or specific)
 General
 Alpha
 Numeric
 Alphanumeric
 Other

Specific
List is located at/in _____

Length 6

Format 99999.9 where "9" represents any digit between 0 and 9

Observed Range 1-5,000

Figure 7-3. ITEM VOLUME Data Dictionary Card

(Task Two continued)

The team compares this card to the CUBE data dictionary card from the Equipment Description System, shown in figure 7-4.

Data Element Name CUBE

Data Element Alias Item Volume CUFT

Data Element Definition The measurement of an item's volume calculated by multiplying the width by the length by the height (all in inches) and then dividing by 1,728.

Domain (check either general or specific)

General

Alpha

Numeric

Alphanumeric

Other

Specific

List is located at/in _____

Length 6

Format 99999.9 where '9' represents any digit between 0 and 9

Observed Range 1-6,000

Figure 7-4. CUBE Data Dictionary Card

Based on a careful review of the data definitions on these two cards, and considering other similarities in the metadata (note the similarities in the alias for cube and the definition for item volume), the team concludes that these two data elements describe identical, or nearly identical, concepts. This relationship is recorded in the team's Data Linkages report (see figure 7-5). The team completes this process for all the data elements on the data flow diagram.

Data Linkages Report

System	Data Element Name	Is Linked To	Data Element Name	System
Medical Battalion Inventory	ITEM VOLUME	→→→	CUBE	Equipment Description
Medical Battalion Inventory	LENGTH	→→→	LENGTH	Equipment Description
...	...	→→→

Figure 7-5. Data Linkages Report



Note: The team does not compare data dictionary cards until it has completed the procedures in chapters 3, 4, and 5 for the source system(s). This is because the business rules that are developed and used in chapters 4 and 5 almost always help to improve or refine the metadata on the source system data dictionary cards created in chapter 3.

Task Three

Repeat Chapter 6

In chapter 6, the team worked with data from the *destination* system. In this task, the team applies business rules to the full (or to a significantly large portion of the) *source* system database files.

Step 1 Obtain source system database records

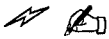
Repeat the procedures defined in chapter 6. Structure the request for source system data to parallel (or to relate to) the data in the destination database report.

Example 1: A team has been working with a fiscal year database with massive amounts of data. To save time and resources, the team decided to select a “time frame” limiting criterion for its database report. After some deliberation, SMEs suggested that data from a randomly selected 30-day month could be assumed to be a good indicator of overall data quality. The team then arbitrarily selected June 1995 as its limiting criterion for the destination database system. Next the team structured a source system database request to cover the same period of time. SMEs reported that the retrieval of all the data from June 1995 from the source system would be easy to implement because an existing program could be tailored easily. In its two database requests, the team achieved an exact “parallel;” that is, the team could compare the data quality in the source data with the quality of the data used to generate the values in the destination system.

Example 2: The team working on the dimensional data issue (used as an example in chapter 6) is structuring its source system database request. The dimensional data values it originally requested from the Equipment Description File had been limited. It had considered only the data provided to the inspection team. Now, for its work with the source system, the DQE team seeks a parallel set of data. In other words, if possible, the team will structure its source system database report request so as to generate data on just those records that

were included in the inspection team report. SMEs report that this type of request involves a labor-intensive and potentially costly selection process. In this instance, the team weighs the costs of a specific selection against the relative merits of

- Using a less precise database report selection criterion and estimating the data effects of the observed data quality
- Restructuring the original (destination system) database request in a way that permits a parallel request in the source systems. This option, of course, requires the team to reestablish a data quality baseline in the destination system.



Note: The term “vertical slice” has been used to describe the concept of selecting parallel or “related” data values in two or more database systems. Since data quantities tend to decrease (data values tend to be aggregated or “rolled-up” in higher level systems), related systems could be visualized as the “layers of stone” in a pyramid. (See figure 7-6.) The foundation of the pyramid could be viewed as “System C”—a source system with large quantities of highly detailed data values. Above that sits “System B” with fewer, more aggregated data values. At the top of the pyramid, typically, sits “System A”—a decision support or management-oriented system containing highly aggregated or summary-level data. A DQE team tasked to evaluate data quality in System A would gradually work “down” the pyramid, considering related data in the lower, more detailed system levels. Conceptually, matching data sets in each successively lower layer of the pyramid form a “vertical slice” of data. The idea is to ensure that the data values that have been aggregated at a higher level system are included in the database report from the lower level system.

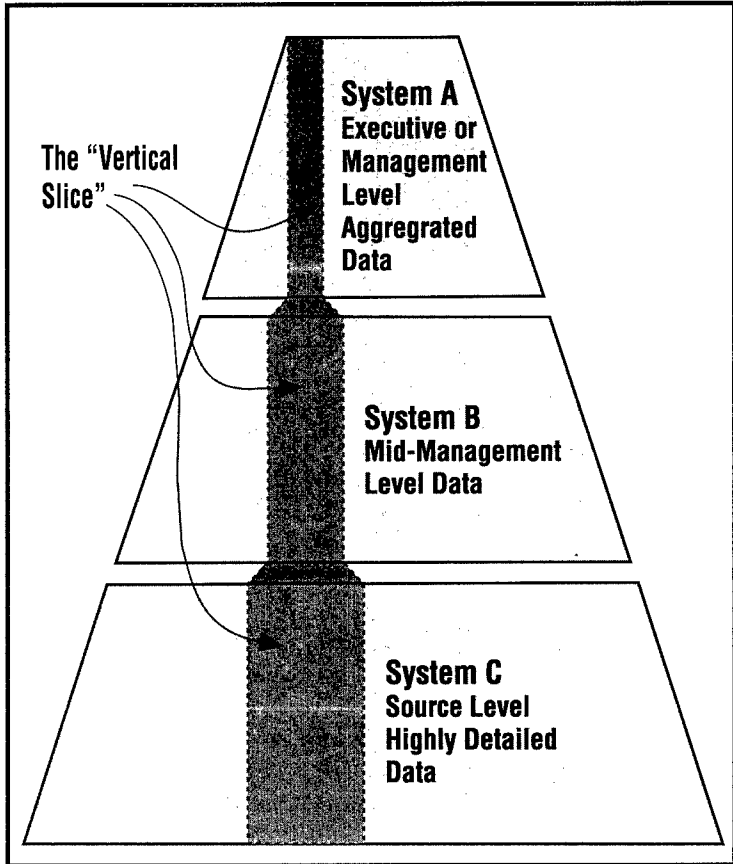


Figure 7-6. The "Vertical Slice"

Step 2 Apply the business rules

Apply the business rules to the records obtained in step 1. See chapter 6, task 2 for details on the development and application of business rules.

Step 3 Organize and report the results

Gather, organize, and report the results from step 2. See chapter 6, task 3 for more detail.

Summary

The team located where and determined how its data values originated. When the data source was determined to be an external system, the team learned how to develop metadata and business rules for the external (source) system. Once constructed, the team was able to apply business rules to the data values in the source system, assess the data quality, and establish a linkage between the data elements in the source system and the elements in the destination system.

John twirled around, simulated a dribble, then moved into position for a game-winning basketball shot. He then arched his back and sent a wadded sheet of paper arching gracefully through the air and into the green trash can. "Two points!" he cheered as his peers looked on.

The team members clapped. Joe picked up a nearby umbrella and, using it as a simulated microphone, said, "Well, folks, here he is! John, in all his glory, has come to the finale of this DQE event. And folks, quite frankly, he's never been happier! But wait, folks! What's happening here? Could it be? ... It is! ... Yes folks, it's..." Joe reached over and picked up a torn computer sheet of paper that had been caught under his desk chair. "It's a 'Data Quality Improvement' award, honoring 'DQE John'!"

The team members roared with laughter as Joe mockingly gestured towards John to come and receive his award. John laughed too, as he quickly picked up another wad of paper and targeted it at Joe.

"Hey, team," interjected the professor as he entered the room and pulled up a chair close to John. "We're not quite through yet! Just because we now have established a relationship between the data in the source and destination systems doesn't mean that we have a root cause yet!"

"What?" asked John with sudden seriousness.

“Of course not,” said the professor. “Recall that when we originally identified a problem, we claimed that it had to have a root cause. We have not found the root cause yet!”

“We haven’t?” asked Maureen with her mouth remaining slightly open.

The team stood dumbfounded. The professor sat quietly.

“I knew this happy moment would be too good to enjoy for long!” sighed Maureen.

“Come on, Maureen,” said the professor. “We’re almost there! A few more actions and we’ll have caught the culprit to your problems!”

“Well, then, lead on, master, lead on...” said Maureen pulling up a chair for the professor.

The team drew around its mentor.

Check on Learning

Responses can be found in appendix C.

1. How does the DQE team develop a data flow diagram?
2. What does the team look for when it compares the source and destination systems’ data dictionary cards?
3. Explain the “vertical slice” concept?

Chapter 8

Finding and Eliminating Root Causes

With his first cup of coffee in-hand, Joe joined John in the now all-too-familiar DQE conference room. John had not yet looked at the data error lists that the team had started to develop. Instead, he was poring over the DLA office floor plan. Noticing Joe, he asked, "How do you like the new office plans?"

Like many others in DLA, Joe had already reviewed the plan for his office. He told John that he was generally pleased with the new arrangement, adding, "I will definitely appreciate a new desk and chair." At the same time, Joe took a moment to lower himself gently into one of the conference room chairs. He was careful not to spill his steaming coffee.

Well," John said pushing the office plans to the side, "I have to admit that on an occasion or two in the past, I have been slow to grasp a few of the DQE concepts. However, our work so far with error lists has not been difficult for me."

"Were you really confused, or were you just testing us?" Joe asked the question to give John the benefit of a doubt, but John's attention had already shifted. He was looking at a data error report that had been lying on the conference room table.

"Joe, we are making some key discoveries here." John was reviewing a list of data errors. "I did not realize how evident a root cause can be when you simply group errors by some common feature."

Joe slid his chair in John's direction to have a better view of the report that John was studying. "What do you mean?"

John replied. "Well, look at this list of the records that we grouped together because the ZIP code in each record violated the business rule that we developed earlier. Remember that we decided that all U.S. ZIP codes had to be at least five characters in length?"

Joe nodded and sipped his coffee.



“Well, do you notice any other trends or common features on this particular list of errors?” John paused to look for a reaction from Joe.

Joe scanned the report with little apparent interest, looking first down each column and then working his way across the pages. A smile started to form at the corners of his mouth as he came to the column of state codes. With only two exceptions in pages of data, the state code was “AK.” “They are all from Alaska,” Joe stated emphatically.

“I noticed that too. That fact does not suggest a root cause to me yet, but I bet it is pertinent. I am also willing to bet that it will help our subject matter expert narrow his or her search for a root cause. What amazes me,” continued John, “is how obvious this became after we grouped our errors. Even though this particular observation does not yet point us to a root cause, I worked up a grouping yesterday that immediately led us to a probable root cause.”

“Really! When did that happen?” Joe had forgotten, for a moment, that he left early the day before for a doctor’s appointment.

“We were working on a list of errors that resulted from a rule that compared data for cubic volume to a value calculated by multiplying length by width by height. Do you remember that business rule?”

“Sure do.” Joe remembered that he had helped to create this particular business rule.

“Well, the whole team was scanning our error list for common features when someone... Maureen, I think... noticed that a significant number of the data values differed by exactly a factor of ten. We recreated a list showing just those records with a ‘factor of ten’ error, and guess what?”

Joe was pleased to see how excited John had become. “What?” he asked.

“Virtually all the ‘factor of ten’ errors originated in our motor transportation unit. After a quick discussion our subject matter expert, we concluded that the form used to record the initial measurements had a design flaw that resulted in this error.”

Just then Maureen and Janine entered the conference room. John asked, “Janine, do you recall our discussion about the form used by motor transportation personnel?”

“I remember that and I recall listing a requirement to review that form on our action item list.” Janine scanned the conference room table, hoping to find a copy of the action item list among the collection of reports and error lists. Instead,

her eyes fell on the new office floor plans. "Whoa," she interrupted herself. "How long have we known about these plans? Where will I sit? How big are our new offices? What floor are we on?"

Maureen walked toward Janine and fired off a few questions of her own. Like Janine's, her questions were asked of no one in particular.

John looked over toward Joe and said, "I think we lost them for a while."

"I think so too," agreed Joe, adding "...but their timing is just about perfect because I need to warm my coffee." Joe made his way toward the door.

"I'll join you," John said. As he headed down the hall, he was pleased to note the sound of not one, but now two sets of rapid-fire questions fading quickly.

"I wonder where our other division's offices are located? Who has an office with a window? Where is our new conference room? Where are the..."





Chapter 8

Finding and Eliminating Root Causes

Introduction

In this chapter, the team returns to its database report. (If the team has been working with more than one system, then it will look at the database reports for each system identified in the data flow diagram developed in chapter 7.) The team will again focus on individual data value errors. This time, however, it will establish data error categories by grouping the data errors according to any notable common trends (task 1). Task 1 results in an error category description and develop likely cause-and-effect relationships. The team will discuss its findings with SMEs and, with confirmation, initiate action to correct both the faulty process and the data value errors (task 2). Finally, it will measure data quality using exactly the same technique used to establish the baseline (task 3).

Key Concepts

-  Data error category
-  Error category list
-  Root cause statement
-  Action item list

.....

Task One

Establish Data Error Categories

In this task, the team will be reviewing all highlighted data errors in a database report and looking for common trends. Any notable common trends will be used to develop data error categories.

Step 1 Look for common trends

Closely examine all the records in the database report that failed a given business rule. Consider all the related data elements and look for common trends. Look for common features in the degree to which (or in the way that) the data value failed the rule.

Example: The team member working on rule 15-4 found that 11,299 of 13,401 errors (about 84 percent) had a “Med Battalion” providing organization code. For the remaining 2,102 errors, no common trends were noted.

The team member working on rule 44-1 found that 8,015 of the 9,548 records containing errors had a “Med Battalion” providing organization code. This team member also noted a set of 1,060 records in which the calculated cube value and the recorded value differed by a factor of exactly 10. The team member also noted a third trend. In 550 records, all with a “Fuels Unit” providing organization code, the calculated and the recorded cube value differed by a factor of 0.028. For 150 records, the team member could find no common trends.

Step 2 Create error categories



Create a category for the data errors that all share a common trend. Give each category a descriptive name. Put each data value error into at least one category (even if that category is called “unexplained error”).

(Task One continued)

Example: The team member working on rule 15-4 created a “Med Battalion” category containing 11,299 records. The remaining 2,102 records were put in an “unexplained error” category.

The team member working on rule 44-1 grouped errors into the following four categories, assigned a descriptive name to each category and generated an error count as shown:

Category	Error Count
Med Battalion	8,015
Factor of 10	1,060
Factor of 0.028	550
Unexplained errors	150
Total	9,775



Note: The total number of errors in categories may exceed the total number of errors in the original group. This is because a given error may fall into more than one category (here, 227 “Factor of 10” errors also fell into the “Med Battalion” subgroup).

Step 3 **Compile an error category list**



Assemble the team and compare results. Note instances where data error categories overlap and combine those data error categories. Write a description for each error category, capturing the general nature of the errors in the group. Develop a list of the error categories.

Example: The team meets and compiles the results of its analyses. The team creates the combined error category list as shown in figure 8-1.

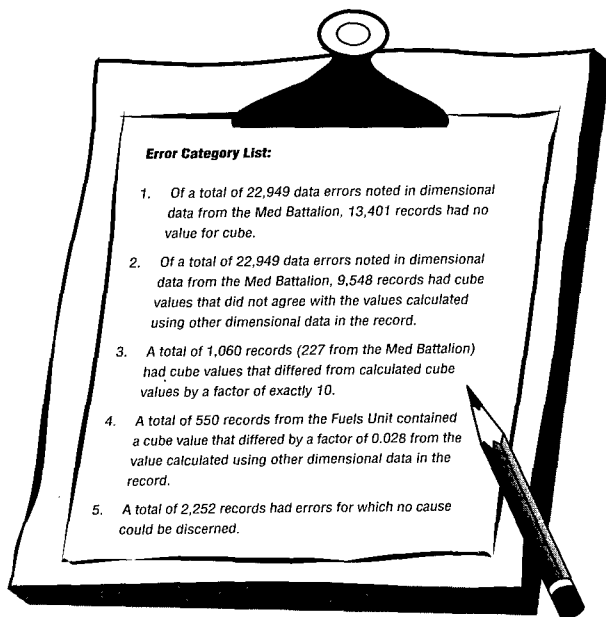


Figure 8-1. Error Category List



Note: In this set of examples, only about 2 percent of the data values fall into an unexplained category. This percentage parallels the results that have been obtained consistently by other teams practicing the procedures described in this handbook. Typically, unexplained errors account for between 2 and 5 percent of the total.

Task Two

Develop an Action Plan

In this task, the team will be reviewing each data error category and identifying the root cause of the data errors. The team then follows through with a plan to correct both the root cause and the data value errors.

Step 1 Interview subject matter experts

Discuss each error category description with SMEs. Ask for an opinion as to the probable cause. If no opinion is offered, review individual data value errors in the category with SMEs. Ask the SMEs to look for and comment on the errors and to look for common trends.

Example: The team meets with the SMEs and asks them for comments on the error category list, one item at a time. The SMEs offer the following observations (the error categories are restated here for clarity):

1. Of a total of 22,949 data errors noted in dimensional data from the Med Battalion, 13,401 records had no value for CUBE.

A SME comments that the automated system used as a source for this data is used daily by the medical community to maintain stock inventories. However, because dimensional data is not a factor in maintaining medical inventories, the SME is aware that data entry personnel frequently leave these fields blank when creating or updating records. The SME was not aware that this data was being transferred to and used by another automated system.

2. Of a total of 22,949 data errors noted in dimensional data from the Med Battalion, 9,548 records had CUBE values that did not agree with the values calculated using other dimensional data in the record.

Here, the SME notes that the comment on category 1 (above) also applies. The SME adds, however, that on the occasion when

dimensional data is entered, the quality control process does not include a check on dimensional data.

3. A total of 1,060 records (227 from the Med Battalion) had CUBE values that differed from calculated cube values by a factor of exactly 10.

The SME suspects that a decimal placement error has occurred. The SME arbitrarily picks a record with this error. Because the equipment name happens to be “truck,” the SME calls the motor transportation unit and requests a physical measurement. Within a half-hour, the SME confirms the error, but also hears the clerk comment that this problem occurs frequently because the blank space for cubic volume on the data entry form is too small to allow a legible entry.

4. A total of 550 records from the Fuels Unit contained a CUBE value that differed by a factor of 0.028 from the value calculated using other dimensional data in the record.

Initially, the SME is unable to explain the error. As the team begins to review individual data errors with the SME, a trend emerges. Based on the past experience, the SME realizes that each record with this particular error originates with a European supplier. By working with metric conversion factors, the SME discovers the dimensional data had been converted to inches, but the cubic measurement had been entered based on a cubic meter unit of measure.

5. A total of 2,252 records had errors for which no cause could be discerned.

The SME notes that this number of errors, while of concern, represents only 2 percent of the total. The SME requests a listing of each record and agrees to review each individually.

Step 2 Establish the root cause



Develop a root cause statement for each item on the error category list. Create the statement by selecting one or more causes from one of the options on the following list:

1. **Policy problem.** The root cause relates directly to a failure on the part of workers or managers to comply with one or more policies. A policy problem also occurs when procedures have been established with no supporting policy. The typical reasons that people fail to comply with policy include: policy is out of date, poorly worded, not available at the work site (in written form) for review, no longer applicable/current, or does not exist.
2. **Procedure problem.** The root cause relates to a failure on the part of workers to comply with written or implied DLA procedures. Procedures are defined in DLA instructions, operating system manuals, local "standard operating procedures" publications, and related written and implied activities. The typical reasons why people fail to comply with procedures include: out-of-date instructions, the practice of unpublished "work-around" procedures, poorly designed or worded instructions, documentation not available at the work site for review, or instructions that are no longer applicable/current.
3. **Training problem.** The root cause relates directly to the opportunity that DLA personnel have had to understand what they are expected to do. Failure on the part of managers or workers to enforce policy or to comply with procedures can be a result of poor training practices. Often, a training problem can be traced to a lack of opportunity for training, long lapses between training occasions, or training course materials that are out of date.
4. **Internal system error.** Typically, in legacy systems, only a small number of data value problems can be traced to errors in the automated program code within a single system. Normally, such errors will occur in an infrequent, but consistent pattern.

5. **Interface system error.** An “interface” problem can be a significant cause of data value errors. Interface errors occur when two or more legacy systems share data values. The root cause of the error is poorly designed or documented metadata.
6. **Unassigned error.** This is the category for all errors for which no other cause can be determined.

Example: The team assigns probable root causes as shown in figure 8-2. The error categories are again restated for clarity.

Error Category	Probable Root Cause(s)
Of a total of 22,949 data errors noted in dimensional data from the Med Battalion, 13,401 records had no value for cube.	<p><u>Policy problem:</u> The team reviewed medical battalion policy and noted it is silent on the importance of acquiring, entering accurately, and maintaining dimensional data values.</p> <p><u>Procedure problem:</u> The team could find no written instructions for medical battalion data entry personnel for entering dimensional data.</p> <p><u>Interface problem:</u> The team notes that the data element name for cubic volume data is CUBE in the Medical Battalion Inventory System and ITEM VOLUME in the Equipment Description File. This relationship had not been uniformly noted in data file transfers in the past. The result was that cubic volume data in the medical battalion system has not been uniformly transferred into the Equipment Description File.</p>
Of a total of 22,949 data errors noted in dimensional data from the Med Battalion, 9,548 records had cube values that did not agree with the values calculated using other dimensional data in the record.	<p><u>Procedure problem:</u> The team could find no written instructions for medical battalion data entry personnel for entering dimensional data.</p> <p><u>Training problem:</u> Medical personnel did not realize they could validate cubic volume data by using a formula and the length, width, and height data.</p>
A total of 1,060 records (227 from the Med Battalion) had cube values that differed from calculated cube values by a factor of exactly 10.	<p><u>Procedure error:</u> The form used to record data values (at the motor transportation unit) is poorly designed. The space for noting cubic measurements is too small. Often, data entry clerks, straining to read the value, fail to see and enter the decimal point.</p>
A total of 2,102 records from the Med Battalion had no discernable cause.	<p><u>Unassigned error:</u> A probable root cause is not known at this time. These data value errors have to be corrected one data value at a time.</p>

Figure 8-2. Probable Root Causes

Step 3 **Develop an action item list**



For each item on the error category list, develop an action plan, assign team member responsibilities, and establish reasonable deadlines. Action plans should include activities to resolve the probable cause and to correct data value errors. Each team should be provided with its respective data value error list and the data quality baseline measurements.



Note: One of the techniques that the team may want to consider as an action item is the development of an automated “filter” at the data entry point for dimensional data. Automated filters, as the name suggests, require modification to the program code in the source system. The sophistication of these filters will depend on the desires of the process owner, available resources, and other factors.

One simple filter option is to modify the program code to require data entry personnel to enter length, width, height, and cube data before they are permitted to store the data record. A more sophisticated filter could calculate the cubic volume based on the entered values for length, width, and height, and then alert data entry personnel when the entered and calculated values differ by a preset margin.

Even if the process owner opts to forego the use of automated filters, Data Quality Engineering business rules can be applied in later system redesigns, upgrades, or wholesale replacements.

Task Three

Reevaluate Data Quality

After action has been taken to eliminate both data value errors and the source of data quality problems, the process owner will be interested in measuring the change in data quality. In this task, the team will learn how to reevaluate data quality using exactly the same procedures used to establish the data quality baseline. The team will request a database report using the same procedure used for the baseline and it will apply the same set of business rules. In fact, this procedure can be repeated as often as necessary to monitor progress toward achieving the process owner's data quality goals.

Step 1 Obtain a new database report

Refer to the Database Report Log created in chapter 6, task 1, and initiate an identical database report request. Use the business rule list (created in chapter 6, task 2) to apply exactly the same rules to the data. (See chapter 6 for examples and more detail.)

Step 2 Generate new data quality measurements

Using the procedure defined in chapter 6, task 3, create a second set of data quality measurements. Graph the results using a "stacked bar chart" technique. Label the first series of data on the bar chart as the data quality baseline. Graph the subsequent series to reflect the change in data quality achieved by the team.



Tip: Dedicated, data software programs are available on the commercial market. These software programs provide the capability to accept large database files, apply a wide variety of business rules, and then sort, print, summarize, and even graph the resulting output.

Example: Figure 8-3 provides an example of a baseline established in January and two subsequent data quality

(Task Two continued)

evaluations taken in 2-month intervals. In this example, by May, the team has reached its goal of having 95 percent of all data values compliant with the business rules.

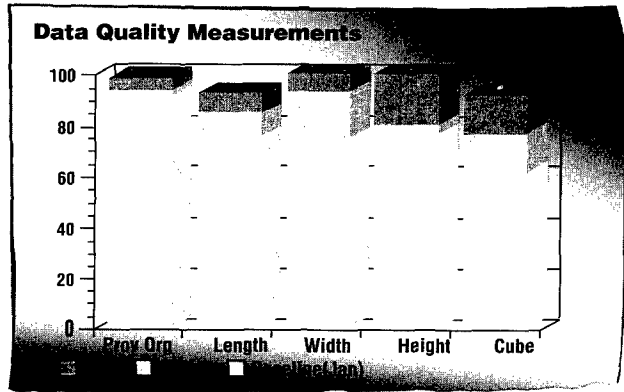


Figure 8-3. Data Quality Measurements



Note: The process of collecting and displaying data quality can be repeated at relatively frequent intervals until the team achieves its goals, and then at longer intervals (or as often as necessary) to ensure that data quality standards are being maintained.

Summary

To pinpoint the root cause, the team identified a common trend among all the data errors. This allowed it to establish a category for each error trend and associate a cause for each error. The assigned causes were verified by SMEs.

The professor stood behind the gathered team as their leader reported on his morning's meeting with Colonel Mathis.

"And so, I am convinced the colonel was extremely pleased with our efforts," John concluded. "He's asked that each of us appear at Monday's Commander's Call, including you, Professor, if possible, to be recognized for not only showing how DLA can save very valuable and needed resources, but employ a reliable assessment methodology to our data programs."

"Good work, team!" yelled Maureen.

Joe shouted out with joy. "I can't believe it! We're finished and everyone likes what we did!"

"You know," interrupted Janine, "if you think about it, we're not really finished, are we? I mean, like we only discovered root causes of our problems and told the colonel about it. Unless I missed something, nothing's been corrected yet, has it?"

The professor stepped forward as John seemingly searched for an answer.

"You are absolutely 100 percent on target, Janine!" the professor said. "You all did intensive research to unravel the mystery of the errors, but now that you know what it is, are you any better off in satisfying your customers?"

"Well, we know the source of the problem and it's pretty obvious that it needs to be eliminated, right?" asked John. "So why don't we go into the data files and just eliminate the errors?"

"Well for one thing, that is not what your charter asked you to do. Remember, Colonel Mathis asked that the team make

recommendations on what would be the best course of action. I assume you did that in your meeting this morning, right?" asked the professor.

"Right!" responded John. "And he said he would implement our recommendations."

"Well, I would suggest that your team work with him on coming up with a viable strategy to do just that. Once the strategy is written down, try it out on a small segment of your operation, see if it works, and if it does, publicize and train others in the new procedures. If it doesn't, then your team should review why and take action accordingly."

"That sounds like a lot of bureaucratic nonsense," said Joe. "Sorry professor, but is that stuff really DQE?"

"This is where the DQE activities back into the more general mainstream of TQM procedures," offered the professor in response to Joe's question. Moving to the board, he drew a large circle. He then cut it into four quadrants. He placed a "P" in the first quadrant, a "D" in the second, a "C" in the third, and an "A" in the last. "Remember this?" he asked after completing his visual.

"Oh, you mean the 'Plan-Do-Check-Act' cycle in TQM?" asked Maureen. "Sure. We've all had to go to TQM classes here on base where they taught that stuff."

Despite an attempt to contain himself, the professor found himself laughing at Maureen's sudden lack of academic sophistication in describing this TQM principle.

Composing himself, the professor continued, "Well, in DQE, we went through the Plan cycle when we localized and defined the problem, as well as learned about its relationships and behaviors. Then when we tested to what extent the errors surfaced in our data program, based on our generated business rules. From that, we discovered root causes. Until we enact countermeasures, we aren't even into the 'Do' portion of the cycle!"

“So if we are following a quality procedure,” said John, “we have to follow through with the ‘Do,’ ‘Check,’ and ‘Act,’ right?”

“Right,” said the professor with emphasis on his selected word response. “You can’t just react to data problems, you must study them! That’s one thing that DQE does for us.”

The professor continued, “You see, this is where I leave and the colonel takes over. As a team, you must decide how to enact countermeasures to eliminate the error sources permanently. Should you put out a policy statement? A directive? Should you start new programs? Begin a training program? There are many options here. Your business culture and allotted resources will dictate which will best resolve the situation. After you decide on which countermeasure is right for this problem, you test it out on a small scale, see if it works, and, if it does, apply the change throughout the system or organization.”

“And then we’re finished?” asked Janine with timidity.

“No,” smiled the professor. “You check data quality again periodically. This happens to be another key feature of DQE. The methodology supports identically configured checks. By checking database reports using exactly the same techniques each time, you or your team can assess the improvement in data quality over time.”

“And then are we through?” asked John winking at Janine.

The professor laughed aloud as he twirled around in John’s desk chair. “You know, I really hate to tell you folks this, but DQE is an everlasting, ongoing effort! If done right, it never ends!”

Check on Learning

Responses can be found in appendix C.

1. How does the team establish data error categories?
2. What sort of guidance can the team get from SMEs in establishing root causes?
3. How can the team measure its success in eliminating the root cause(s) of poor data quality?

Practical Exercises

Use the sample data values provided in appendix D. Answers are provided in appendix E.

1. Apply the following business rule to the data element called ZIP CODE. Highlight the values that fail to comply with your business rule.

Rule 8-1: The data element called ZIP CODE consists of a mandatory five numeric characters, followed by a dash (-) and an optional four additional, numeric characters.

Create an error list containing all the data values that failed to comply with this business rule. Sort the list by type of error or any other common trend. Group the errors by trend and write a brief description of each trend.

Epilogue

A group of subject matter experts was now filing out of the team's conference room. John was aware that he now had a cool feeling of relief in the place of his pre-briefing jitters. He had been the principal speaker at the now completed briefing. He plopped into a swivel chair and let his eyes drift around the familiar conference room walls.

John noted that the team's conference room looked better than it had in weeks. The team had obviously spent some time recently in an effort to reorder and straighten the materials taped to the walls. As now organized, the materials on the walls told a story, probably a little like Egyptian hieroglyphics. The walls contained a record, in sometimes cryptic pictures and key phrases, of the "life and times" of a team called the Data Quality Engineers. While he scanned the walls, John tried to remember what he did for a hobby before DQE. He had a quick flashback to a mountain stream and the flicker of a fly-fishing rod. This particular mental image faded as John focused on the materials of the conference room walls.

John noted that the team charter and the "Quality Improvement Story" now occupied a prominent spot on the wall immediately to the left of the door. Beside that, on the same wall, came a description of the customer, the product, and components of the product.

The wall on the narrow end of the room was reserved for a projection screen, but the wall opposite the door contained highlights of the team's effort to break the components of the product down to the data element level. This wall also contained a large banner that simply said "Metadata!" The banner served two purposes. First, it seemed to shout to anyone entering the room that metadata was a key element of the DQE methodology. Second, it served to remind the team that while the banner could be mounted easily on a wall, the metadata itself could not. Initially the team's considerable investment in the effort to create metadata resulted in annotated 5"x7" cards, stored in a plastic box on the conference room table. Later, as their files grew in size and complexity, the team switched to using word processing software on a notebook-sized computer.

The wall opposite the projection screen contained list after list of business rules. Some rules were based on the metadata. Others were grouped under titles like "Policy," "Procedure," and "If...then...Rules."

Finally, the wall to the right of the door contained several graphs showing a "data quality baseline," with annotations by the process owner, assigning priorities to the data quality improvement effort. Next came data error lists,

categorized by type of error. Those reports had been the subject of the briefing prepared by the team and delivered, for the most part, by John.

John's mental tour of the conference room was interrupted as the other members of the team returned after walking with SMEs part way toward their own offices.

"I think we impressed them," Joe stated with pride.

Maureen agreed "I think they were impressed on at least two counts," she said. "First, they were amazed at the sheer number of data value errors that we were able to find and identify for them. But, they were even more impressed at our ability to sort the errors by error type. Isn't it amazing how sorting data errors by type or category tends to reveal the root cause of the error?"

Though Maureen's question was addressed to no one in particular, Joe responded. "Yes! I was definitely pleased to be able to point to policy problems, procedure shortfalls, training needs, and other causes. My favorite root cause, though, was probably the one related to the foreign supplier."



"That was my favorite too," piped Janine. She had been a key player in this particular root cause determination. "I still remember the feeling of satisfaction I had when I discovered that for a certain, relatively large number of records, the calculated cubic data differed from reported cubic data by a constant factor. The providing organization code gave us the first clue to the source of this error. Each item with this particular error had a foreign manufacturer. Once we knew that, it did not take us long to figure out that the constant error factor related to the use of 'inches' for linear dimensions and 'cubic centimeters' for cubic measurements."

"I was particularly pleased to see the SMEs leaving the room this afternoon with annotated lists of data errors," observed Joe.

"I also liked the idea that in many cases, we not only highlighted an error, we predicted a correct value!" Janine added.

“Personally, I was pleased to be able to measure the data quality in a quantitative way and then to offer to measure it again later, in exactly the same manner,” Maureen observed. She added, “You can be sure the SMEs will be looking for improvements in their next report.”

Maureen turned toward John and noted that he seemed lost in his own thoughts. “John,” she asked, “what did you find most rewarding?”

“Rainbow trout,” John mumbled as he wandered out of the room. He was already on his way out of the conference room and headed toward his desk, and in particular, the file drawer where he kept vacation request forms.

Appendix A

Glossary and Acronyms

alias. A metadata element. An optional second or additional name for a data element. An alias is useful when the data element name provides little insight into its real meaning or use within the system.

business rule. A statement of fact about the data that an organization creates, maintains, or uses.

CASE. Computer-aided software engineering.

data element. The lowest level at which meaningful pieces of information are stored in a database system.

Data Quality Engineering (DQE). A proven process for restoring the validity, accuracy, and reliability of data maintained in legacy systems.

derived data. Derived data results when automated processes (like counting, adding, subtracting, multiplying, etc.) are applied to existing data values to generate new data. (Sometimes called “roll-up” or “summary” data.)

DOD 8320.1-M. *Data Administration Procedures Manual.* The Department of Defense Instruction on data administration.

DOD 8320.1-M-1. *Data Element Standardization Procedures.* The Department of Defense Instruction on data element standardization.

domain, general. A metadata element. The specification of values that can be combined to create valid data values.

domain, specific. A metadata element. An enumerated set of data values allowed in representations of a data element.

format. A metadata element. Defines a certain convention for storing or displaying data values. (Called “picture” in some database systems.)

legacy system. A system designed before standards for system design or database implementation procedures existed.

length. A metadata element. The maximum permissible length of the stored data value, measured in terms of total characters.

mean. A central point in a set of evenly distributed data values. (Sometimes called “average.”)

metadata. Information describing the characteristics of data; data or information about data; descriptive information about an organization’s data, data activities, systems, and holdings. Metadata elements include data element name, alias, domain, length, format, and observed range.

null. The lack of a data value (no spaces, zeros, or characters).

observed range. A metadata element. The complete set of values between the lowest (or smallest) and the highest (or largest) value in a given set of data values.

roll-up data. See derived data.

SME. Subject matter expert.

summary data. See derived data.

standard deviation. A measure of how the data values are spread out around the mean. If the standard deviation is small, then most or all of the data values are clustered around the mean. If the standard deviation is large, then the data values vary widely.

vertical slice. Parallel or related data values in two or more database systems.

Appendix B

Program Team Charter

1. **TEAM NAME:** The Data Quality Engineers
2. **SPONSOR:**
 - A. ***Business Area:*** Data Administration
 - B. ***Process Owner:*** Colonel Mathis
3. **PURPOSE:** To provide written recommendations for a reliable and efficient process that supports DLA Management requirements.
4. **BACKGROUND:**
 - A. ***Previous Work***

The Defense Logistics Agency (DLA) has implemented a variety of information systems that provide data to key DLA decision makers. The most recent and most critical of these systems is the DLA Corporate Executive Information System (EIS), which provides information in the form of performance measures. An infrastructure of computer hardware, software, and networks has been implemented to support the EIS. Two materiel management measures are currently produced: Stock Availability and Number of Back Orders. Additional materiel management measures have been identified and are near implementation.

Having noticed a widespread perception that their management information is invalid, the Defense Contract Management Command (DCMC) chartered a team to investigate in early 1993. A core team led by the Management Information Systems Group was formed. The full team included members from several DCMC Headquarters offices and all districts. The team performed an assessment of a random sample of workload and performance data elements and found a variety of problems and possible root causes. Its techniques can serve as a model for later assessments.

Decision makers using the DLA EIS require data that is accurate, complete, timely, and has integrity. Decisions based upon bad data may waste already shrinking resources. Personnel must also expend time and effort to manually identify and correct bad data. Although DLA Regulation 4700.5, *DLA Data Management Program (DMP)*, prescribes policy that mandates the treatment of data as a DLA corporate resource, no process currently exists to ensure data quality. Potential problems that may exist with the data could result from errors in source data values, improper source data aggregation, and faulty data aggregation algorithms within the EIS. Concerns about accountability for the data also exist. Although the DMP addresses the creation and management of data, it excludes “the derivation and usage of this data for reporting purposes.”

The true level of data quality within the DLA, including the Corporate EIS, is unknown. Metrics can be established and used to measure the current level of data quality. Identified data defects may then be corrected. This will not provide a permanent solution, however. Measures and procedures established by the DCMC team and the Data Quality team, and those from other DLA business areas, will give DLA a mechanism for assessing and improving the process of data management. This will result in higher quality of data products and services provided to DLA's customers.

B. Strategy

The team will support a strategy of employing Total Quality Management methods to recommend improvements to processes identified as causing data defects. The Data Quality Engineers will build on the techniques used by the DCMC management information reliability team. This effort will focus on implemented DLA EIS measures: Stock Availability and Number of Back Orders. Metrics will be established and filters defined to measure data quality. An automated Data Quality Engineering (DQE) tool developed by the United States Marine Corps will be used to make the measurements. Data will initially be assessed for one selected supply center. Metrics established will then be used to assess the remaining centers. The team will provide

monthly progress reports on the DQE team activities to the process owner.

C. Objectives

- a. Conduct a data quality assessment of the two indicators currently implemented within the DLA Corporate EIS.
- b. Develop metrics to measure data quality.
- c. Acquire tools to expand the automation of data audit/assessment and quality control techniques/methods.
- d. Develop an exportable process for continuous improvement of data quality that can be given to data administrators and data stewards for reuse in other projects.
- e. Assign accountability for identified data according to the recommendations of the DLA data stewards.
- f. Gain upper-management support for the continuance of a data quality program.

D. Resources

- a. Time
 1. Final written recommendation and presentation due to the process owner in 6 weeks.
 2. Meetings should consume approximately 2 hours (no more than 5) per week.
 3. Interim progress reports preferred at 2-week intervals.
- b. Money and People
 1. Any requests for overtime pay or additional funding to support this initiative must have written approval.
 2. Team member appointees are on attached sheet. Additional members may be appointed as needed, with prior approval.

Appendix C

Check on Learning Responses

Chapter 1

1. Who develops the charter and what should the charter contain?

The charter is typically written by the process owner and given to the team leader. The charter should contain: the problem (as known), the problem's history, the desired team outcome, the allotted resource support for the team, the team constraints, the approved schedule and location of team meetings, the process owner's deadline date, the periodic progress report schedule, and the identification of appointed team members, facilitator, and team leader.

2. What role does the process owner serve on a TQM team?

The process owner is the manager with decision authority over the area in question. The process owner's role is to establish an objective, to select an initial team composition, and to be the team's advocate and supporter.

3. Who decides who should be on a TQM team?

The process owner establishes the initial team member composition. However, after assessing its own strengths and weaknesses, the team may decide that it lacks a particular skill or experience. In such cases, the team leader checks with the process owner to determine if changes in the team member composition will be considered.

Chapter 2

1. Why is the customer a critical element in the DQE process?

The customer is the person who expects a service or product as a result of a process. Without the customer, the process has no reason to run.

2. Explain how the DQE team breaks down each problem area.

First, the team defines a product and then breaks the product down into component areas. Then, the team eliminates the components that do not relate directly to the product or process problem. Next the team divides component areas into subcomponents (if necessary) and continues dividing until it reaches the data element level.

3. How does the team decide which data elements are related to the problem area?

The team compares the problem statement to its list of data elements. The team also uses training, “gut feeling,” and experience as additional aids in determining what data elements relate to the problem area.

Chapter 3

1. List some good characteristics of a data element definition.

- a. It describes the meaning that the team sees in the sample data values.
- b. It has only one meaning or interpretation.
- c. It describes a single concept.
- d. It is written using simple, commonly used words.

2. Distinguish between the concepts of “specific” and “general” domain.

A specific domain is a *list* of enumerated data values from which an allowable value must be selected. The general domain, on the other hand, defines the *raw materials* or *building blocks* that can be used to create valid data values.

3. Write a format specification for a data element called ZIP CODE. Explain the conventions you used.

99999-9999, where “9” represents a digit between 0 and 9.

Chapter 4

1. Distinguish between the concepts of “metadata” and “business rule.”

Metadata is information (or data) about the data. A business rule is a statement of fact about a data element or a statement expressing a relationship between data elements.

2. Describe how the team records a relationship between a data element and a business rule.

First, the team forms a numbered list of business rules. The team then records the number of an applicable business rule on the back of the 5”x7” data dictionary card for a given data element.

3. In a sample of 200 data values, how many “suspect” values would you expect to identify if you applied the “95 percent interval observed range” business rule?

If the data values are clustered normally around a central point, then the 95 percent interval range will include approximately 95 out of 100 (or 190 out of 200) values. In other words, approximately 10 values out of 200 will lie outside the 95 percent interval range. The values that lie outside the 95 percent range are called “suspect” because they may or may not be errors.

Chapter 5

1. Briefly describe the technique the team uses to create a policy/procedure business rule.

The team discovers the rule based on policy or procedure by asking subject matter experts to explain how they identified specific data value errors in a given sample. Often, in the explanation of an error, an SME will describe how a given policy or procedure established a limit or other restriction on the values for a given data element.

2. Briefly describe two ways to use data-generating rules.

First, a data-generating rule can be used to compare a generated value with a value already stored in the database. A discrepancy

between the generated and stored values may indicate a data quality problem. Second, a data-generating rule can be used to create data values in cases where data is missing or where the field incorrectly contains a zero or null value.

3. Explain how the use of discovery business rules may result in changes to the team's list of data value errors.

Discovery business rules are based on a relationship between certain data elements. The new relationships may reveal that previously identified "errors" are in fact valid values. Alternatively, a newly defined relationship may allow the team to identify new data value errors.

Chapter 6

1. How does a team prepare a database report request?

The team reviews the data error list and selects from one to five target data elements. It then selects additional data elements as needed to ensure that the records in the printed report on the target data element(s) can be identified. Finally, the team reviews the business rules related to the target data elements and adds additional data elements as needed to support the use of applicable business rules.

2. Why is the data quality baseline an important tool in DQE?

The data quality baseline organizes the error discoveries and measures the impact of each error category. It also allows the team to measure its progress toward quality data.

3. What sort of guidance might the process owner be expected to give as a result of reviewing the quality baseline?

The process owner will have information that he or she needs to determine where limited agency resources should be applied to correct errors. The team therefore would expect the process owner to establish priorities and to set near term objectives.

Chapter 7

1. How does the DQE team develop a data flow diagram?

First the team lists the data element sources, then the team determines the source and destination systems' names. The team labels the boxes with data element names and shows the movement of data by drawing arrows from the source to the destination system.

2. What does the team look for when it compares the source and destination systems' data dictionary cards?

The team looks for the similarities between the source and destination systems' data element names, definitions, and other metadata elements.

3. Explain the "vertical slice" concept?

The "vertical slice" concept is the process of selecting parallel or related data value sets from two or more database systems.

Chapter 8

1. How does the team establish data error categories?

The team examines all records in the database report that failed a given business rule. It then looks for trends among those records and it gives each trend a descriptive name.

2. What sort of guidance can the team get from SMEs in establishing root causes?

SMEs can offer insight into what the root cause may be.

3. How can the team measure its success in eliminating the root cause(s) of poor data quality?

The team repeats its baseline measurement process periodically (after actions have been taken to eliminate root causes). The incremental change in data value quality is displayed using a stacked bar chart technique.

Appendix D

Practice Data Value Samples

See the graphic on the following page.

(Appendix D continued)

No.	Name	Street Address	City
1	ABBOTT ROBERT LINDSEY	140 MOUNT PLEASANT RD	FAYETTEVILLE
2	ALLEN DANIEL PATRICK	5204 LEITH RD APT F	ALEXANDRIA
3	ALLEN DAVID J	842 TIDBALL ST	SOMERSWORTH
4	ALTDORFER JEFFREY T		PITTSBURGH
5	ALTIERI RICHARD THOMAS	83 COLUMBIA AVENUE	ALEXANDRIA
6	AMMONS THOMAS DANIEL	226 6TH STREET #H	LEXINGTON
7	BAINES MARK DAVID	2108 NW 55TH	LEAVENWORTH
8	BALDVINS LYNN ANN	9511 PERRIN BEITEL 411	COLORADO SPRINGS
9	BARRETT DAVID JOHN	894 B BEECH ST	WATERTOWN
10	BARRIOS DANIEL JUAN	5120 PELHAM STREET	CLARKSVILLE
11	BEATTY THERESE M	7878 YARMOUTH DR	LAKE IN THE HILL
12	BECHTEL PETER BROOKS	148 FLOWER AVE EAST	ITHACA
13	BELL CURTIS LEMAN JR	300 CORONADA STREET	COLORADO SPRINGS
14	BENARD GERALD PAUL	126 WOODFIELD PL	EPSOM
15	BLEEKER SHAWN CURTIS	1105 HETHERINGTON LP	BOONE
16	BOLDUC DONALD CHARLES	4200 B CALLE LADERO	CLARKSVILLE
17	Breagy Stephen Michael	1815 OLMSTEAD APT 225	OKLAHOMA CITY
18	BRESSETT JOHN DANIEL	260 STONEYFIELD DR	TACOMA
19	BRODEUR MARC PHILIP	78 WINTER ST	LAUREL
20	BURGESS GREGG JAMES	4018 TENTH AVENUE	FAYETTEVILLE
21	CHRETIEN GREGORY S	6100 PATRICK HENRY	KANSAS CITY
22	CLEVELAND RICHARD W	10103 MADRONAWOOD DR	WOODBIDGE
23	CLOUTIER PERRY NORMAN	9408 GOODEN DR	BOSTON
24	COTE MARC GEORGES	373 BRISTOL AVE #401	ENGLEWOOD
25	COTE PETER CHARLES	1217 COBBLESTONE LN	SAN DIEGO
26	COUTURE ERIC JOHN	114 OAK TREE DR	WEST GARDINER
27	DANISON JOHN ELLIS	2301 52ND AVE APT 11	SCOTIO
28	DECESARE MICHAEL NEIL	9230 130TH ST	WOODBIDGE
29	DENISH MELVIN AUSTIN	1674 S 60TH ST	ENTERPRISE
30	DESJARDINS ANDREW R	P O BOX 721594	SAN DIEGO
31	DEVEREUX STEPHEN A	717 CAYCE DR	ANCHORAGE
32	DIVNEY ROBERT SCOT	2501 BACON RD 1006	BOSTON

	ST	ZIP Code	Age	Hgt	Wt	DOB	Error Codes
	NC	00604-	23	69	154	11/14/71	
	VA	22303-0000	30	69	161	10/14/64	
	NH	03878-0000	24	70	166	9/30/70	
	PA	15237-0000	28	62	105	6/25/67	
	VA	22305-	30	68	168	9/12/64	
	NC	27292-0000	32	76	205	8/31/62	
	KS	66048-0000	38	78	238	12/5/56	
	CO	80919-0000	41	66	150	7/6/53	
	NY	13601-0000	27	71	175	11/21/71	
	TE	37042-0000	46	66	124	1/1/49	
	IL	60102-0000	38	71	171	5/7/57	
	NY	14850-0000	27	71	184	7/7/67	
	CO	80919-0000	42	67	143	5/16/53	
	NH	03234-0000	24	71	192	11/24/70	
	NC	28607-0000	33	69	168	7/9/61	
	TE	37042-0000	37	69	180	2/23/58	
	OK	73503-0000	39	72	206	9/27/55	
	WA	98444-0000	50	72	190	3/29/45	
	MD	20708-0000	28	72	178	5/19/67	
	NC	28311-0000	33	70	184	3/14/62	
	KS	66442-0000	39	73	141	5/14/56	
	VA	22192-		67	152	10/10/65	
	MA	01433-0000	23	71	165	8/22/71	
	CO	80111-0000	41	71	181	11/15/53	
	CA	92172-0000	43	70	165	12/8/51	
	ME	044345-	24	69	151	7/22/70	
	NY	12302-0000	27	68	155	12/17/67	
	VA	22192-3361	30	64	146	7/2/65	
	AL	36330-0000	36	71	185	3/15/59	
	CA	92310-	43	72	188	12/8/51	
	AK	9507-0000	39	68	177	1/2/56	
	MA	01433-0000	24	69	178	5/11/71	

(Appendix D continued)

No.	Name	Street Address	City
33	DUNCAN PETER JOSEPH	112 OAKLAND DRIVE	WAHIAWA
34	DUPUIS JAMES GEORGE	405 DILLON DRIVE	SECURITY
35	FOSTER LEONA R	268 INDIAN TRAIL	MEAD
36	GAVER KEVIN LOUIS	635 PULMAN PLACE	TUCSON
37	GERLACK FRANK REGINALD	320 MORILINE AVE.	HINESVILLE
38	GLEDHILL RICHARD LOUIS	504 24TH STREET	SAN ANTONIO
39	GODDARD DAVID ROLAND	RD 1 BOX 190A	SALT LAKE CITY
40	GRAVES MICHAEL LEON	7550 KILIHEA COURT	AIEA
41	GROVER RANDY A	5151 CRACKERJACK LN	BOSTON
42	HOLMES ALAN PAUL	P O BOX 70876	VIRGINIA BEACH
43	JEWETT STEVEN CURTIS	1307 UPTON RD 1	WOODBIDGE
44	KANGAS DAVID MARTIN	2 CAPRON STREET	TACOMA
45	KAYE STEPHEN JOSEPH	5602 SNOW LOOP	OAK GROVE
46	KLEIN JEFFREY		GETTYSBURG
47	KRINSKY GLEN LAEL B	1751 ASHLAND CITY RD	NEWPORT NEWS
48	LABRANCHE DAVID F	107 HOOK DRIVE	SEASIDE
49	LAMBERT WAYNE BRUCE	13065 MICHIE COURT	WATERTOWN
50	LAMPHERE JOHN C	397 FORREST HILL DRIVE	COLORADO SPRINGS
51	LARocca STEPHEN ALFRED	19269 WOODSIDE DR	WEST POINT
52	LLOYD LAWRENCE J	8681 BOSTON STREET	TACOMA
53	LOHMAN PAMELA SOUTHAR	1200 E RIVER RD #K139	CLARKSVILLE
54	MATTISON TIMOTHY JAME	402 KEACH LOOP	FAYETTEVILLE
55	MCCARTHY TIMOTHY MICH	13307 HYDE PARK	BENA
56	MCLAUGHLIN MICHAEL R	5601 N 37TH ST EE6T	EAST WINTHROP
57	MCPAHON JAMES GEORGE		BAINBRIDGE
58	MONTROND PETER B	536 ANDOVER RD	KILLEEN
59	MOORE KEVIN MICHAEL	6023 S LIMA ST	SEATTLE
60	NIGARA KAREN LEE	2801 NOBLE FIR COURT	ARLINGTON
61	NEWELL PETER LANGLEY	PO BOX 39	ENTERPRISE
62	NICHOLS BRUCE R JR	1049-9 CHENA ROAD	FAYETTEVILLE
63	NIKONCHUK WILLIAM PAUL	PO BOX 25	HUNTSVILLE
64	NORMAN JERE PACKWOOD	1120B THOMPSON CIRCLE	CENTREVILLE

	ST	ZIP Code	Age	Hgt	Wt	DOB	Error Codes
	HI	96786-0000	47	70	175	8/22/47	
	CO	89011-0000	43	71	173	1/3/52	
	WA	99021-0000	50	70	178	2/26/45	
	AZ	85718-0000	42	67	162	9/3/52	
	GA	31313-	34	70	170	11/14/60	
	TX	78244-	41	69	170	6/9/54	
	UT	84121-0000	27	69	161	1/2/68	
	HI	00000-0000	23	67	160	3/18/72	
	MA	02115-1806	24	69	160	3/20/71	
	VA	23451-	31	66	122		
	VA	22060-0000	29	65	146	5/31/66	
	WA	98408-0000	49	68	146	3/20/46	
	KY	42262-0000	27	70	155	1/2/68	
	PA	17325-0000	28	72	161	6/31/67	
	VA	23602-0000	32	99	169	6/12/63	
	CA	93955-0000	44			8/27/50	
	NY	13601-0000	27	69	180	11/5/67	
	CO	80925-0000	42	70	170	4/11/53	
	NY	10996-9999	27	70	170	2/14/68	
	WA	98407-0000	48	71	155	7/27/46	
	TE	37043-	37	71	155	8/24/57	
	NC	28307-0876	33	71	159	3/18/62	
	VA	23018-	31	72	200	3/21/64	
	ME	044343-	24	71	185	8/15/70	
	PA	17502-0000	28	73	221	6/2/67	
	TX	76542-0000	40	67	164	2/29/55	
	WA	98433-0000	50	50	175	6/10/45	
	VA	22201-0000	30	64	125	1/30/65	
	AL	36330-0000	36	67	145	10/9/58	
	NC	28314-0000	33	66	172	11/28/61	
	AL	35803-0000	35	70	160	12/25/59	
	VA	22020-0000	29	62	120	6/22/66	

(Appendix D continued)

No.	Name	Street Address	City
65	OWEN JOAB PATTERSON	1234 DHARAHA DRIVE	CONTOOCCOOK
66	PADDOCK ROBERT E	PO BOX 352	COPPERAS COVE
67	PELKEY RANDALL JAMES	4051 BREAKING DAWN ST	ENTERPRISE
68	PERKINS DAVID GERARD	6565 BONIFAS COURT	COLORADO SPRINGS
69	PERLOFF HAL JOSHUA	225 E JEFFERSON RD	ROSWELL
70	PETTIGREW CHARLES E	98 415 COLBY WAY	WILSON CITY
71	PILLOW KATRINA GARDENER	3348 ROLLINWOOD DR	FRA
72	RAPISIS JOHN ALBERT JR	810 ROYAL CROWN LANE	WATERTOWN
73	RICE JON ANTHONY	6704 S ARTESIAN WAY 11	WATERTOWN
74	ROBERTS GEORGE H III	414D KINGS PARK DR	FT. WALTON BEACH
75	RODESCHIN DARRIN HENRY	16604 BRIARDALE RD	SOUTHERN PINES
76	ROGERS ERVIN LOUIS	N 13018 CHRONICLE	UPPER MARLBORO
77	ROSE MICHAEL WILLIAM	14566 WOODLAND DR	WAHIAWA
78	ROTE CHARLES X	P O BOX 313 RT 202	BRUNSWICK
79	ROY RODNEY KEITH	3817 D COLLIER ST	FAYETTEVILLE
80	SALLIES CHRISTY ANNE	3905 CLEARWATER DR	FORT EUSTIS
81	SILVASY ANNE MARIE	221B LEE ROAD	SAN ANTONIO
82	SPENCER JOHN HERBERT	580 BROCKWAY RD	CLARKSVILLE
83	SULLIVAN CHRISTINE LEE	1213 FAICHENY DR APT 1	JUNCTION CITY
84	TAMKE ERIC KIMBALL	323 METZ ROAD	LIVERPOOL
85	TANGNEY WILLIAM PATRICK	990 CRANDALL DR	LEWIS
86	THIBODEAU CHRISTOPHER	1223 THOMASON DRIVE	FT WAINWRIGHT
87	THOMAS KENNETH E	11444 DUNLORING PL	BALTIMORE
88	THOMPSON JOHN RUSSELL	726 N. DANVILLE ST	ADAMS CENTER
89	TIEDEMANN DAVID OWEN	864-A POPLAR STREET	SIERRA VISTA
90	WALBRIDGE BRYAN A		DEVON
91	Walsh Kenneth Joseph	42 DENBEIGH BLVD	OKLAHOMA CITY
92	WETTLAUFER JOHN N	RT 5A BOX 216	STEILACOOM
93	White Brian Neal	PO BOX 261	LAWTON
94	WHITE TIMOTHY WADE	807 HADLEY RD	HAMPTON
95	WOOD JEROLD ANDREW	5703 FENWICK DRIVE	WOODBIDGE
96	YUILL ROBERT GRAHAM	2015 SUNFLOWER CT	ROCKVILLE

	ST	ZIP Code	Age	Hgt	Wt	DOB	Error Codes
	NH	03229-0000	24	75	172	12/2/70	
	TX	76522-0000	41	71	173	2/29/56	
	AL	36330-0000	36	71	182	9/17/58	
	CO	80906-0000	41	72	170	8/31/53	
	NO	88201-1220	21	74	192	2/12/74	
	NY	13603-0000	32	71	207	1/1/63	
	AK	9505-0000	65	71	165	11/21/49	
	NY	13601-0000	27	72	185		
	NY	13601-	27	72	195	11/24/67	
	FL	32547-0000	34	76	205	7/23/60	
	NC	28387-0000	33	66	159	10/6/61	
	MD	20772-0000	28	71	166	3/25/67	
	HI	96786-0000	47	73	195	8/11/47	
	ME	044011-	24	68	233	9/6/70	
	NC	28311-0000	33	70	165	12/13/61	
	VA	23604-	32	74	180	6/4/63	
	TX	78217-	40	68	154	7/23/54	
	TE	37042-0000	37	69	180	11/16/57	
	KS	66441-	38	67	150	3/5/71	
	NY	13090-0000	27	68	123	11/24/67	
	NC	28307-0000	33	73	203	5/5/62	
	AK	9703-0000	16	68	177	1/1/79	
	MD	21239-0000	28	72	142	10/17/66	
	NY	13606-0000	27	73	190	9/15/67	
	AZ	85635-	42	67	160	2/4/53	
	PA	19333-0000	28	74	180	6/2/67	
	OK	73503-0000	39	69	174	8/7/55	
	WA	98388-0000	47	69	159	7/10/47	
	OK	73505-	40	70	180	6/4/55	
	CAL	00000-0000	23	67	174	1/25/72	
	VA	22192-0000	29	72	154	9/18/65	
	MD	20855-	28	69	95	12/11/66	



.....

Appendix E

Answers to Practical Exercises

The following questions and answers refer to the data sample in appendix D.

Chapter 3

1. Create the definition and domain metadata for the data element called NAME:

Definition: A word or words (including abbreviations) used to identify an individual person.

Domain: General, Alpha

2. Create the length and format metadata for the data element called ZIP CODE:

Length: 10

Format: 99999-9999 where the digit “9” represents any digit between 0 and 9 (Last four digits are optional).

3. Create the observed range metadata for the data element called WT:

Observed range: 95 (record 96) to 238 (record 7).

Chapter 4

1. Write a domain-based business rule for the data element called NAME. Highlight the data values that fail to comply with your rule.

Rule 4-1: NAME must consist only of uppercase letters, <space>, or punctuation (period and/or comma).

The following records contained numbers: 16, 60, 76, 85, and 88. The following records contained lowercase letters: 17, 91, and 93. See the error list at the end of this appendix.

2. Write a domain-based business rule for the data element called *sr*. (Note that appendix G may be referenced in your response.) Highlight the data fields that fail to comply with your rule.

Rule 4-2: The value in *sr* must consist of one of the values listed in appendix G.

The following records failed this test: 10, 16, 53, 69, 82, and 94. See the error list at the end of this appendix.

3. A subject matter expert mentions that U.S. ZIP Codes consist of five (mandatory) digits followed by a dash and then four (optional) digits. Write one or more format-based business rules for the data element called *ZIP CODE*.

Rule 4-3-1: *ZIP CODE* must contain exactly five digits followed by a dash (-).

Rule 4-3-2: *ZIP CODE* must contain either four digits or four spaces after the dash (-).

4. Calculate the 95 percent interval for the data element called *wt*. (The mean value is 168.73 and the standard deviation is 23.83.) Highlight the values that fall outside the 95 percent interval.

Mean - 2 times the standard deviation

$$168.73 - (2 \times 23.83)$$

Low value: 121.07

Mean + 2 times the standard deviation

$$168.73 + (2 \times 23.83)$$

High value: 216.39

Rule 4-4-1: For *wt*, values that fall outside the 95 percent range (121.07 to 216.39) are suspect.

Records 4, 7, 57, 64, 78, and 95 are identified as suspect. See the error list at the end of this appendix.



Note: The following similar rules apply to the other numeric fields: *AGE* and *HT*.

Rule 4-4-2: For AGE, values that fall outside the 95 percent range (17.13 to 50.86) are suspect.

Records 17 and 86 are identified as suspect. See the error list at the end of this appendix.

Rule 4-4-3: For HT, values that fall outside the 95 percent range (60.75 to 79.04) are suspect.

Records 47 and 59 are identified as suspect. See the error list at the end of this appendix.

Chapter 5

1. A subject matter expert mentions the mandatory retirement age for the organization is 60. Use this information to build and apply a policy/procedure business rule. Highlight the data values that fail to comply with your rule.

Rule 5-1: The values in the data element called AGE must be less than or equal to 60.

Record 71 fails this test. See the error list at the end of this appendix.

2. Create a data-generating business rule for the data elements called AGE and DOB. Highlight the data values that fail to comply with your rule. Use your rule to calculate the missing value for the data element called AGE in record 22. (Assume this data sample was created 1 June 1995.)

To verify a value or calculate a missing value for AGE, use the following rule.

Rule 5-2: For the data elements called AGE and DOB, one of the following relationships must exist:

If the first two digits of the value in DOB are less than 06, then the value in AGE must be equal to 95 minus the value of the last two digits in DOB. For example, in record 10, the value in the

data element called `DOB` is 01/01/49. Since the value of the first two digits is less than 06, calculate the value of `AGE` as $95-49=46$. This agrees with the value actually present in `AGE` for record 10.

If the first two digits of the value in `DOB` are equal to or greater than 06, then the value in `AGE` must be equal to 94 minus the value of the last two digits in `DOB`. For example, in record 1, the value in `DOB` is 11/14/71. Since the value of the first two digits is greater than 06, calculate the value of `AGE` as $94-71=23$. This agrees with the value actually present in `AGE` for record 1.



Note: This business rule takes note of the fact that a person may not have celebrated his or her birthday by 1 June 1995. Hence, two possible relationships.

The value in `DOB` for record 22 is 10/10/65. Since the value of the first two digits is greater than 06, the value of `AGE` is $94-65=29$. (Richard Cleveland celebrated his 30th birthday after this report was created.)

Chapter 8

1. Apply the following business rule to the data element called `ZIP CODE`. Highlight the values that fail to comply with your business rule.

Rule 8-1: The data element called `ZIP CODE` consists of a mandatory five numeric characters, followed by a dash (-) and an optional four additional, numeric characters.

Create an error list containing all the data values that failed to comply with this business rule. Sort the list by type of error or any other common trend. Group the errors by trend and write a brief description of each trend.

Figure E-1 contains the `ZIP Code` values that failed this business. Sorting them by `ZIP Code` showed an interesting fact. It grouped them by state and revealed consistent patterns of

errors. Sorting the list by error type, that is records that were too short or too long or improper domain (non-numeric), would provide the same information.

All entries from Maine (ME) have six digits instead of five before the dash (-). No other record had a similar error. The source of data from Maine should be checked to see why this consistent pattern exists.

All entries from Alaska (AK) have only four digits instead of five before the dash (-). No other record had a similar error. The source of data from Alaska should be checked to see why this consistent pattern exists.

All entries from Washington (WA) have the letter O instead of digits following the dash (-). No other record had a similar error. The source of data from Washington should be checked to see why this consistent pattern exists.

The table shows the value of creating an error list where the faulty records can be examined as a group and patterns easily seen.

No.	ST	ZIP Code	Error Codes
8	ME	044011-	8-1
6	ME	044343-	8-1
3	ME	044345-	8-1
7	AK	9505-0000	8-1
3	AK	9507-0000	8-1
9	AK	9703-0000	8-1
9	WA	98388-0000	8-1
5	WA	98407-0000	8-1
4	WA	98408-0000	8-1
6	WA	98433-0000	8-1
2	WA	98444-0000	8-1
4	WA	99021-0000	8-1

Figure E-1. Error List for Rule 8-1

(Appendix E continued)

No.	Name	Street Address	City
4	ALTDORFER JEFFREY T		PITTSBURGH
7	BAINES MARK DAVID	2108 NW 55TH	LEAVENWORTH
10	BARRIOS DANIEL JUAN	5120 PELHAM STREET	CLARKSVILLE
16	BOLDUC DONALD CHARLES	4200 B CALLE LADERO	CLARKSVILLE
17	Breagy Stephen Michael	1815 OLMSTEAD APT 225	OKLAHOMA CITY
18	BRESSETT JOHN DANIEL	260 STONEYFIELD DR	TACOMA
22	CLEVELAND RICHARD W	10103 MADRONAWOOD DR	WOODBIDGE
26	COUTURE ERIC JOHN	114 OAK TREE DR	WEST GARDINER
31	DEVEREUX STEPHEN A	717 CAYCE DR	ANCHORAGE
35	FOSTER LEONA R	268 INDIAN TRAIL	MEAD
44	KANGAS DAVID MARTIN	2 CAPRON STREET	TACOMA
46	KLEIN JEFFREY		GETTYSBURG
47	KRINSKY GLEN LAEL B	1751 ASHLAND CITY RD	NEWPORT NEWS
52	LLOYD LAWRENCE J	8681 BOSTON STREET	TACOMA
53	LOHMAN PAMELA SOUTHAR	1200 E RIVER RD #K139	CLARKSVILLE
56	MCLAUGHLIN MICHAEL R	5601 N 37TH ST EE6T	EAST WINTHROP
57	MCMAHON JAMES GEORGE		BAINBRIDGE
58	MONTROND PETER B	536 ANDOVER RD	KILLEEN
59	MOORE KEVIN MICHAEL	6023 S LIMA ST	SEATTLE
60	NIGARA KAREN LEE	2801 NOBLE FIR COURT	ARLINGTON
64	NORMAN JERE PACKWOOD	1120B THOMPSON CIRCLE	CENTREVILLE
69	PERLOFF HAL JOSHUA	225 E JEFFERSON RD	ROSWELL
71	PILLOW KATRINA GARDENER	3348 ROLLINWOOD DR	FRA
76	ROGERS ERVIN LOUIS	N 13018 CHRONICLE	UPPER MARLBORO
78	ROTE CHARLES X	P O BOX 313 RT 202	BRUNSWICK
82	SPENCER JOHN HERBERT	580 BROCKWAY RD	CLARKSVILLE
85	TANGNEY WILLIAM PATRICK	990 CRANDALL DR	LEWIS
86	THIBODEAU CHRISTOPHER	1223 THOMASON DRIVE	FT WAINWRIGHT
88	THOMPSON JOHN RUSSELL	726 N. DANVILLE ST	ADAMS CENTER
91	Walsh Kenneth Joseph	42 DENBEIGH BLVD	OKLAHOMA CITY
92	WETTLAUFER JOHN N	RT 5A BOX 216	STEILACOOM
93	White Brian Neal	PO BOX 261	LAWTON
94	WHITE TIMOTHY WADE	807 HADLEY RD	HAMPTON
96	YUILL ROBERT GRAHAM	2015 SUNFLOWER CT	ROCKVILLE

	ST	ZIP Code	Age	Hgt	Wt	DOB	Error Codes
	PA	15237-0000	28	62	105	6/25/67	4-4-1
	KS	66048-0000	38	78	238	12/5/56	4-4-1
	TE	37042-0000	46	66	124	1/1/49	4-2
	TE	37042-0000	37	69	180	2/23/58	4-1, 4-2
	OK	73503-0000	39	72	206	9/27/55	4-1
	WA	98444-0000	50	72	190	3/29/45	8-1
	VA	22192-		67	152	10/10/65	5-2 (Age = 29)
	ME	044345-	24	69	151	7/22/70	8-1
	AK	9507-0000	39	68	177	1/2/56	8-1
	WA	99021-0000	50	70	178	2/26/45	8-1
	WA	98408-0000	49	68	146	3/20/46	8-1
	PA	17325-0000	28	72	161	6/31/67	G-19 (From Appendix G)
	VA	23602-0000	32	99	169	6/12/63	4-4-2
	WA	98407-0000	48	71	155	7/27/46	8-1
	TE	37043-	37	71	155	8/24/57	4-2
	ME	044343-	24	71	185	8/15/70	8-1
	PA	17502-0000	28	73	221	6/2/67	4-4-1
	TX	76542-0000	40	67	164	2/29/55	G-17 (From Appendix G)
	WA	98433-0000	50	50	175	6/10/45	4-4-1, 8-1
	VA	22201-0000	30	64	125	1/30/65	4-1
	VA	22020-0000	29	62	120	6/22/66	4-4-1
	NO	88201-1220	21	74	192	2/12/74	4-2
	AK	9505-0000	65	71	165	11/21/49	4-4-3, 5-1, 8-1
	MD	20772-0000	28	71	166	3/25/67	4-1
	ME	044011-	24	68	233	9/6/70	4-4-1, 8-1
	TE	37042-0000	37	69	180	11/16/57	4-2
	NC	28307-0000	33	73	203	5/5/62	4-1
	AK	9703-0000	16	68	177	1/1/79	4-4-3, 8-1
	NY	13606-0000	27	73	190	9/15/67	4-1
	OK	73503-0000	39	69	174	8/7/55	4-1
	WA	98388-0000	47	69	159	7/10/47	8-1
	OK	73505-	40	70	180	6/4/55	4-1
	CAL	00000-0000	23	67	174	1/25/72	4-2
	MD	20855-	28	69	95	12/11/66	4-4-1



.....

Appendix F

U.S. State Code Abbreviations

State	Abbreviation
Alabama	AL
Alaska	AK
Arizona	AZ
Arkansas	AR
California	CA
Colorado	CO
Connecticut	CT
Delaware	DE
Dist. of Col.	DC
Florida	FL
Georgia	GA
Guam	GU
Hawaii	HI
Idaho	ID
Illinois	IL
Indiana	IN
Iowa	IA
Kansas	KS
Kentucky	KY
Louisiana	LA
Maine	ME
Maryland	MD
Massachusetts	MA
Michigan	MI
Minnesota	MN
Mississippi	MS
Missouri	MO

State	Abbreviation
Montana	MT
Nebraska	NE
Nevada	NV
New Hampshire	NH
New Jersey	NJ
New Mexico	NM
New York	NY
North Carolina	NC
North Dakota	ND
Ohio	OH
Oklahoma	OK
Oregon	OR
Pennsylvania	PA
Puerto Rico	PR
Rhode Island	RI
South Carolina	SC
South Dakota	SD
Tennessee	TN
Texas	TX
Utah	UT
Vermont	VT
Virginia	VA
Virgin Islands	VI
Washington	WA
West Virginia	WV
Wisconsin	WI
Wyoming	WY

Appendix G

Date Format Business Rule List

Rule Sets Based on Date Fields

Rule sets based on date fields may check:

- The validity of the formatting (picture),
- The observed range or domain, or
- Internal and external relationships.

Dates may be written in many formats. Some examples include:

DDMMYY 060594
MMDDYY 050694
MM/DDYY 5/6/94
(Leading zeros may be optional)

DD MMM YY 5 Jun 94
MMM... DD YYYY June 6 1994
(Month will vary from 3 to 9 characters, leading ZERO may be optional)

YYDDD 94156 (Julian Date)
MMDDYYYY 05061994
(Newer systems use four digits for year to cover the upcoming change in century)

Use the information provided in the format to select appropriate rule sets.

1. The (data element name) data element must contain (number) characters exactly.
2. The (data element name) data element may contain (range of numbers) characters. (Use this rule set if leading zeros are optional or if Month is spelled out.)

3. The (data element name) data element domain is numeric.
4. The (data element name) data element domain is numeric or slashes (/).
5. The (data element name) data element domain is alphanumeric.
6. The (data element name) data element domain is alphanumeric or spaces.
7. The Day characters in the (data element name) data element have a range of 1-31, inclusive. (Use this rule set if leading zeros are optional.)
8. The Day characters in the (data element name) data element have a range of 01-31, inclusive.
9. The Day characters in the (data element name) data element have a range of 001-365, inclusive. (Use this rule set for Julian dates.)
10. The Day characters in the (data element name) data element have a range of 001-366, inclusive. (Use this rule set for Julian dates that include leap years.)
11. The Month characters in the (data element name) data element have a range of 01-12, inclusive.
12. The Month characters in the (data element name) data element have a range of 1-12, inclusive. (Use this rule set if leading zeros are optional.)

-
13. The Month characters in the (data element name) data element must have one of the following values: JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC. (This rule set may be modified to require all uppercase letters or to permit a mix of upper- and lowercase.)

 14. The Month characters in the (data element name) data element must have one of the following values: JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER. (This rule set may be modified to require all uppercase letters or to permit a mix of upper- and lowercase.)

 15. The Year characters in the (data element name) data element must have a range of 00-99. (This range may be modified by usage, i.e., if no future dates were allowed, the range would be 00-95.)

 16. The Year characters in the (data element name) data element must have a range of 0001-9999. (This range may be modified by usage, i.e., most dates will be within the last 100-200 years allowing a range of 1895-1995 or 1795-1995.)

Rule sets may be combined to further identify data errors:

17. If the Month characters in the (data element name) data element are 02 (or 2, if leading zeros are not required), FEB, or FEBRUARY, the Day characters must have a range of 1-28, inclusive.

18. If the Month characters in the (data element name) data element are 02 (or 2, if leading zeros are not required) *and* the Year characters are evenly divisible by four (i.e., it is a Leap Year), the Day characters must have a range of 1-29, inclusive.

19. If the Month characters in the (data element name) data element are listed below, the Day characters must have a range of 1-30, inclusive:

APR	APRIL	04 (or 4)
JUN	JUNE	06 (or 6)
SEP	SEPTEMBER	09 (or 9)
NOV	NOVEMBER	11

20. If the Month characters in the (data element name) data element are listed below, the Day characters must have a range of 1-31, inclusive:

JAN	JANUARY	01 (or 1)
MAR	MARCH	03 (or 3)
MAY	MAY	05 (or 5)
JUL	JULY	07 (or 7)
AUG	AUGUST	08 (or 8)
OCT	OCTOBER	10
DEC	DECEMBER	12

Numerous rule sets may be created to further narrow the allowed range of values based on the usage of the data element:

- Is it a person's birth date? Is the person a child? An adult? Retired?
- Is it a report of a past event? Is it a recent event? Annual events would normally consider Month and Year.
- Does it represent a future event or requirement? Annual events would normally consider Month and Year.

21. The Year characters in the (data element name) data element must have a range of 30-81 (or 1930-1981). (This is an example of a data element reflecting Date of Birth for soldiers whose ages may be expected to fall between 18 and 65.)

22. The Year characters in the (data element name) data element must have a range of 94-95 (or 1994-1995). (This

is an example of a rule set for an annual event which does not consider Month.)

23. The values in the (data element name) data element must have one of the following relationships: a Year value of 94 (or 1994) and a Month value greater than 05 (MAY) or a Year value of 95 (or 1995). (This is an example of a rule set for an annual event which does consider Month.)

Many rule sets may be designed to compare the order of dates, if more than one is present within a record:

- An event must begin before it ends.
- An item must be ordered before it may be shipped or delivered.
- A person must be born before they are married, get a driver's license, join the Army, etc.

24. The values in the Year characters in the (data element #1 name) data element must be less than or equal to those in the (data element #2 name) data element. (This is an example of a rule set that requires that the event characterized by data element #1 happen before the event characterized by data element #2.)



Other rule sets may identify a requirement for certain events to happen (or not happen) within a specific interval of time:

- A person must be at least 10 years old to get a military dependent's ID card.
- A person must get married within 3 months of the date a wedding license is issued.
- A supply requisition which has not been completed within 60 days of initial order is overdue.
- Replacement passports must be obtained at least every 10 years.

25. The values in the Year characters in the (data element #1 name) data element must be less than those in the (data element #2 name) data element by at least 10. (This is an example of a rule set that requires that the event characterized by data element #2 not happen until at least 10 years after the event characterized by data element #1.)

Index

Page numbers shown in **bold** indicate where a term is defined.
An *italic* font refers to the story line.

-  lightning bolt **viii**
-  tip **xii**
- 95 percent interval 84-85, **85**

A

- action item list 158, **160**
- aggregated data 130, 133, 143, 144
- alias. See data element alias.
- average. See mean.

B

- business rule x, 63, **64, 67**, 89, 91, 120, 164, 167
 - domain-based **69, 77**
 - format-based **77**
 - observed range based 82-85, **85**
 - data-generating 93, 95, **105**
 - discovery **92, 93, 95**
 - if...then... 95, **101, 167**
- business rule/data element relationship 74, 80, 88

C

- charter 2, 5, **7, 167**
- customer's product **23, 167**

D

- database report log **119, 161**
- database value report 111, **115, 151, 161**
- data dictionary **41, 138**
- data element **27, 28**
 - alias 39, **41, 43**
 - definition x, 35, 39, **41, 45-46,**
 - diagram **29, 91, 138**
 - domain 39, **50**
 - duplication 47
 - format **55, 56, 77**
 - length **36, 51, 52**

name x, 35, 39, 43
observed range x, **56**, 58
data error category **153**, 156, 163
data flow diagram 135, **136**
data linkages **130**, **134**, 138, 140
data quality baseline 112, **125**, 133, 161, 167
Data Quality Engineering viii, ix
data quality metric **124**, 125, 161
data value **36**
error list 74, 81, 88, 99, 102, 107, 167
sample ix, **21**, 30, 31, 43, 52, 56, 70, 78, 82, 86, 97, 99,
101, 102, 105, 129, 136
destination system 135, 137, **138**, 139, 142, 145
domain. See also data element domain.
alpha **50**
alphanumeric **50**
general **50**
numeric **36**, **50**, 52
specific **50**

E

error category list **154**, 156

F

format. See data element format.

I

interface problem **159**

internal system problem **158**

L

legacy system **vii**

length. See data element length.

limiting criteria **117**, 118, 119

look-up table. See domain, specific.

M

maximum character count quantity. See data element length.

mean **83**

metadata x, **35**, 39, **41**, 60, 63, 67, 72, 80, 89, 91, 129, 167

N

numeric domain. See domain, numeric.

O

observed range. See data element observed range.

P

policy problem **158**

policy/procedure business rule 93, 95, **98**, 167

problem statement **24**, 27, 28, 91

procedure problem **158**

process owner **2**, 7, **13**, 126, 133

product component **26**, 167

R

record **31**

roll-up data. See aggregated data.

root cause 92, 133, 146, **147**, **158**, 163, 164

S

sample data value. See data value sample.

source **29**

source system 129, 130, **133**, 136, 137, 138, 139, 142, 145

specific business rule list **120**

standard deviation **83**

storyboard 11, 167

subject matter expert (SME) 30, 49, 72, 73, 76, 80, 87, 92, 97,
101, 104, 116, 136, 156, 163

summary data. See aggregated data.

suspect data element **28**

T

TQM **2**, 164

TQM customer 18, **23**, 91, 167

training problem **158**

U

unassigned error **159**

V
vertical slice *130, 143*

Credits and Contacts

This handbook was prepared by PRC Inc., under contract to the Defense Logistics Agency. (Contract/Purchase Order Number GS-22F-0053B, Delivery Order YK01 issued by ADP/T Contracting Office, DACO-PA, Cameron Station, Alexandria, Virginia 22304-6100, dated 28 June 1994.)

A Total Quality Management (TQM) approach was used to develop material contained in the handbook. The TQM key team members are listed in the table below.

Name	Role	Organization
Steve Broussard	DLA Leader	DLA
Karen Dean	Staff Reviewer/Contributor	DLA
Mickey Slater	Staff Reviewer/Contributor	DLA
Joe Lehman	PRC Leader	PRC
John Gossner	Principal Writer	PRC
Fran Kassinger	Principal Writer	PRC
Pam Ahn	Graphic Arts Editor	PRC
Belinda Lai	Graphics Arts/Text Editor	PRC
Colleen Dernbach-Pelar	Technical Editor	PRC

Team Member Assignment

The team wishes to recognize Mr. Dick Horne, DLA, for having the original concept for a data quality handbook and then for providing advice and guidance at key points throughout the process of bringing the handbook task to successful completion.

This handbook supplements the DLA instruction on data management.

For comments or further information, please call (703) 767-2165/2172 or (DSN) 427-2165/2172.
