8

Analyzing Systems Using Data Dictionaries

Systems Analysis and Design, 8e

Kendall & Kendall

Learning Objectives

- Understand analysts use of data dictionaries for analyzing data-oriented systems.
- Create data dictionary entries for data processes, stores, flows, structures, and logical and physical elements of the systems being studied, based on DFDs.
- Understand the concept of a repository for analysts' project information and the role of CASE tools in creating them.
- Recognize the functions of data dictionaries in helping users update and maintain information systems.

Cataloging

- Data flow diagrams can be used to catalog:
  - Data processes
  - Flows
  - Stores
  - Structures
  - Elements
- Cataloging takes place with the data dictionary

Major Topics

- The data dictionary
- The data repository
- Defining data flow
- Defining data structures
- Defining data elements
- Defining data stores
- Using the data dictionary
- XML

The Data Dictionary

- A reference work of data about data (metadata)
- Collects and coordinates data terms, and confirms what each term means to different people in the organization

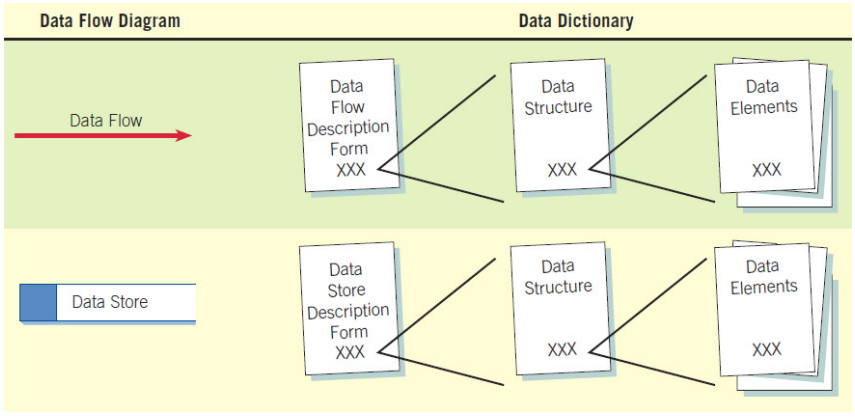
Need for Understanding the Data Dictionary

- Provide documentation.
- Eliminate redundancy.
- Validate the data flow diagram.
- Provide a starting point for developing screens and reports.
- Determine the contents of data stored in files.
- To develop the logic for DFD processes.
- Create XML.

The Data Repository

- A data repository is a large collection of project information.
- It includes:
  - Information about the data maintained by the system
  - Procedural logic and use cases
  - Screen and report design
  - Data relationships
  - Project requirements and final system deliverables
  - Project management information

How Data Dictionaries Relate to Data Flow Diagrams (Figure 8.1)



Data Dictionary Categories

- Data flows
- Data structures
- Elements
- Data stores

Defining the Data Flow

- ID—identification number
- Unique descriptive name
- A general description of the data flow
- The source of the data flow
- The destination of the data flow
- Type of data flow
- The name of the data structure describing the elements
- The volume per unit time
- An area for further comments and notations

Describing Data Structures

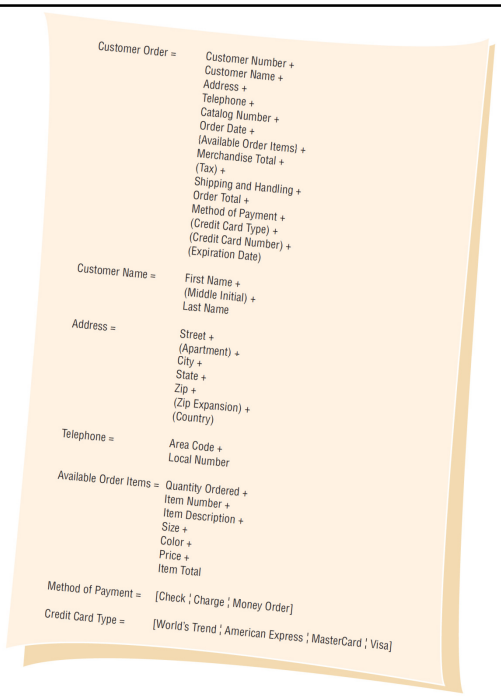
- Data structures are made up of smaller structures and elements.
- An algebraic notation is used to describe data structures.

Algebraic Notation

- Equal sign, meaning "is composed of"
- Plus sign, meaning "and"
- Braces {} meaning repetitive elements
- Brackets [] for an either/or situation
- Parentheses () for an optional element

D
V

Customer Order at
e 8.4)



```
Customer Order =    Customer Number +
                    Customer Name +
                    Address +
                    Telephone +
                    Catalog Number +
                    Order Date +
                    {Available Order Items} +
                    Merchandise Total +
                    (Tax) +
                    Shipping and Handling +
                    Order Total +
                    Method of Payment +
                    (Credit Card Type) +
                    (Credit Card Number) +
                    (Expiration Date)

Customer Name =     First Name +
                    (Middle Initial) +
                    Last Name

Address =           Street +
                    (Apartment) +
                    City +
                    State +
                    Zip +
                    (Zip Expansion) +
                    (Country)

Telephone =         Area Code +
                    Local Number

Available Order Items = Quantity Ordered +
                    Item Number +
                    Item Description +
                    Size +
                    Color +
                    Price +
                    Item Total

Method of Payment = [Check ' Charge ' Money Order]

Credit Card Type =  [World's Trend ' American Express ' MasterCard ' Visa]
```


Structural Records

- A structure may consist of elements or structural records.
- These are a group of elements, such as:
  - Customer name
  - Address
  - Telephone
- Each of these must be further defined until they are broken down into their component elements.

Structural Records Used in Different Systems

- Structural records and elements that are used within many different systems are given a non-system-specific name, such as street, city, and zip.
- The names do not reflect a functional area.
- This allows the analyst to define them once and use in many different applications.

Structural Record Example

Customer Name =	First Name + (Middle Initial) + Last Name
Address =	Street + (Apartment) + City + State + Zip + (Zip Expansion) + (Country)
Telephone =	Area Code + Local Number

Logical and Physical Data Structures

- Logical:
  - Show what data the business needs for its day-to-day operations.
- Physical:
  - Include additional elements necessary for implementing the system.

Physical Data Structures

- Key fields used to locate records
- Codes to identify record status
- Transaction codes to identify different record types
- Repeating group entries
- Limits on items in a repeating group
- Password

ple from
ure 8.6)

Element Description Form

ID
Name Customer Number
Alias Client Number
Alias Receivable Account Number
Description Uniquely identifies a customer who has made any business transaction within the last five years.

Element Characteristics

Length 6 Dec. Pt. ____
Input Format 9(6)
Output Format 9(6)
Default Value
☑ Continuous or ☐ Discrete

☐ Alphabetic
☐ Alphanumeric
☐ Date
☑ Numeric
☐ Base or ☑ Derived

Validation Criteria

Continuous
Upper Limit <999999
Lower Limit >0

Discrete Value | Meaning

Comments The customer number must pass a modulus-11 check digit test. It is derived because it is computer generated and a check digit is added.

all 8-19

Data Element Characteristics

- Element ID
- The name of the element
- Aliases
- A short description of the element
- Element is base or derived
- Element length
- Type of data
- Input and output formats
- Validation criteria
- Default value
- An additional comment or remark area

Element ID

- Optional entry
- Allows the analyst to build automated data dictionary entries

The Name of the Element

- Should be:
  - Descriptive
  - Unique
- Based on what the element is commonly called in most programs or by the major user of the element

Aliases

- Synonyms or other names for the element
- Names used by different users in different systems
- A CUSTOMER NUMBER may also be called a RECEIVABLE ACCOUNT NUMBER or a CLIENT NUMBER.

Short Description of the Element

- An example might be:
  - Uniquely identifies a customer who has made any business transactions within the last five years

Element Is Base or Derived

- A base element is one that has been initially keyed into the system.
- A derived element is one that is created by a process, usually as the result of a calculation or a series of decision-making statements.

Element Length

What should the element length be?

- Some elements have standard lengths, state abbreviations, zip codes, or telephone numbers.
- For other elements, the length may vary and the analyst and user community must decide the final length.

Element Length Considerations

- Numeric amount lengths
- Name and address fields
- Other fields

Name and Address Length

Element	Length	Percent of data that will fit (United States)
Last Name	11	98
First Name	18	95
Company Name	20	95
Street	18	90
City	17	99

Data Truncation

- If the element is too small, the data will be truncated.
- The analyst must decide how this will affect the system outputs.
- If a last name is truncated, mail would usually still be delivered.
- A truncated email address or Web address is not usable.

Type of Data

- Alphanumeric or text data
- Formats
  - Mainframe: packed, binary, display
  - Microcomputer (PC) formats
  - PC formats, such as Currency, Number, or Scientific, depend on how the data will be used

Some Examples of Data Formats Used in PC Systems (Figure 8.7)

Data Type	Meaning
Bit	A value of 1 or 0, a true/false value
Char, varchar, text	Any alphanumeric character
Datetime, smalldatetime	Alphanumeric data, several formats
Decimal, numeric	Numeric data that are accurate to the least significant digit; can contain a whole and decimal portion
Float, real	Floating-point values that contain an approximate decimal value
Int, smallint, tinyint	Only integer (whole digit) data
Currency, money, smallmoney	Monetary numbers accurate to four decimal places
Binary, varbinary, image	Binary strings (sound, pictures, video)
Cursor, timestamp, uniqueidentifier	A value that is always unique within a database
Autonumber	A number that is always incremented by one when a record is added to a database table

Format Character Codes (Figure 8.8)

Formatting Character	Meaning
X	May enter or display/print any character
9	Enter or display only numbers
Z	Display leading zeros as spaces
,	Insert commas into a numeric display
.	Insert a period into a numeric display
/	Insert slashes into a numeric display
-	Insert a hyphen into a numeric display
V	Indicate a decimal position (when the decimal point is not included)

Validation Criteria

- Ensure that accurate data are captured by the system
- Elements are either:
  - Discrete, meaning they have fixed values
  - Continuous, with a smooth range of values

Default Value

- Include any default value the element may have
- The default value is displayed on entry screens.
- Reduces the amount of keying
  - Default values on GUI screens
    - Initially display in drop-down lists
    - Are selected when a group of radio buttons are used

Comment or Remarks Area

- This might be used to indicate the format of the date, special validation that is required, the check-digit method used, and so on