8

Analyzing Systems Using Data Dictionaries

Systems Analysis and Design, 8e
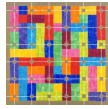## Kendall & Kendall

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.

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

Data Stores

- Data stores are created for each different data entity being stored.
- When data flow base elements are grouped together to form a structural record, a data store is created for each unique structural record.
- Because a given data flow may only show part of the collective data that a structural record contains, many different data flow structures may need to be examined to arrive at a complete data store description.

Describing the Data Store

- The data store ID
- The data store name
- An alias for the table
- A short description of the data store
- The file type (computer or manual)
- File format(Database table or flat file)
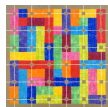
Describing the Data Store (Continued)

- The maximum and average number of records on the file as well as the growth per year
- The file or data set name specifies the file name, if known.
- The data structure should use a name found in the data dictionary.
- Primary and secondary keys
- Comments
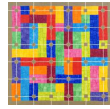
Creating the Data Dictionary

- Data dictionary entries
  - Created after the data flow diagram is completed or
  - Created as the data flow diagram is being developed
- Created using a top-down approach
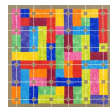
Two Data Flow Diagrams and Corresponding Data Dictionary Entries for Producing an Employee Paycheck (Figure 8.11)
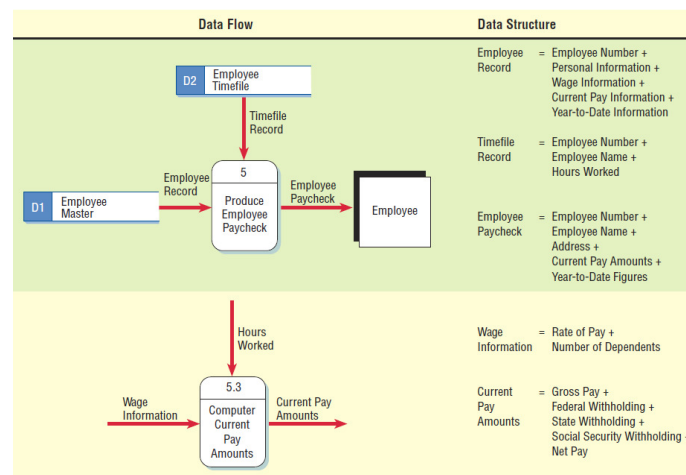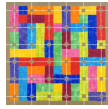
Analyzing Input and Output

- A descriptive name for the input or output
- The user contact responsible
- Whether the data is input or output
- The format of the data flow
- Elements indicating the sequence of the data on a report or screen
- A list of elements

Kendall & Kendall Copyright © 2011 Pearson Education,   8-9

Developing Data Stores

- Represent data at rest.
- Contain information of a permanent or semipermanent (temporary) nature.
- When data stores are created for only one report or screen, we refer to them as "user views".

Kendall & Kendall Copyright © 2011 Pearson Education,   8-10
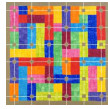
Using the Data Dictionary

- To have maximum power, the data dictionary should be tied into a number of systems programs.
- May be used to
  - Create screens, reports, and forms
  - Generate computer language source code
  - Analyze the system design, detecting flaws and areas that need clarification

Create Screens, Reports, and Forms

- Use the element definition and their lengths.
- Arrange the elements in a pleasing and functional way using design guidelines and common sense.
- Repeating groups become columns.
- Structural records are grouped together on the screen, report, or form.
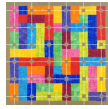
Analyze the System Design, Detecting Flaws and Areas that Need Clarification

- All base elements on an output data flow must be present on an input data flow to the process producing the output.
- A derived element should be created by a process and should be output from at least one process into which it is not input.
- The elements that are present in a data flow coming into or going out of a data store must be contained in the data store.

Using Data Dictionaries to Create XML

- XML is used to exchange data between businesses.
- XML addresses the problem of sharing data when users have different computer systems and software or different database management systems.
- XML documents may be transformed into different output formats.
- XML is a way to define, sort, filter, and translate data into a universal data language that can be used by anyone.
- XML may be created from databases, a form, software programs, or keyed directly into a document, text editor, or XML entry program.
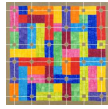
Using Data Dictionaries to Create XML (Continued)

- The data dictionary is an ideal starting point for developing XML content.
- A standard definition of the data is created using a set of tags that are included before and after each data element or structure.
- XML elements may also include attributes.
- The XML document tends to mirror the data dictionary structure.
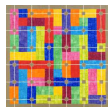
XML Document Type Definitions

- Used to determine if the XML document content is valid
- DTDs may be created using the data dictionary.
- DTD may be used to validate the XML document.

Slide 1

Data Dictionary

Customer =
 Name +
 Address +
 Current Balance +
 (Order Information) +
 Payment

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

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

Order Information = Order Number +
 Order Date +
 Ship Date +
 Total

Payment =
 [Check | Credit Card] +
 Payment Date +
 Payment Amt

Check =
 Check Number

Credit Card =
 Credit Card Number +
 Expiration Date

XML

```
<?xml version="1.0">
<customers>
<customer number="C15008">
  <name type="1">
    <lastname>Stadler</lastname>
    <firstname>Karen</firstname>
    <middle_initial>L</middle_initial>
  </name>
  <address>
    <street>123 Oak Street</street>
    <apartment>Suite 16</apartment>
    <city>Madison</city>
    <state>WI</state>
    <zip>43704</zip>
    <country>United States</country>
  </address>
  <current_balance>123.45</current_balance>
  <order customer_number="C15008">
    <order_number>00123</order_number>
    <order_date format="yyyymmdd">2008-06-23</order_date>
    <ship_date format="yyyymmdd">2008-06-25</ship_date>
    <total>1345.89</total>
  </order>
  <order customer_number="C15008">
    <order_number>00127</order_number>
    <order_date format="yyyymmdd">2008-09-18</order_date>
    <ship_date format="yyyymmdd">2008-09-26</ship_date>
    <total>240.00</total>
  </order>
  <payment>
    <check>
      <check_number>7234</check_number>
    </check>
    <payment_date format="yyyymmdd">2008-09-30</payment_date>
    <payment_amt>1585.89</payment_amt>
  </payment>
</customer>
</customers>
```
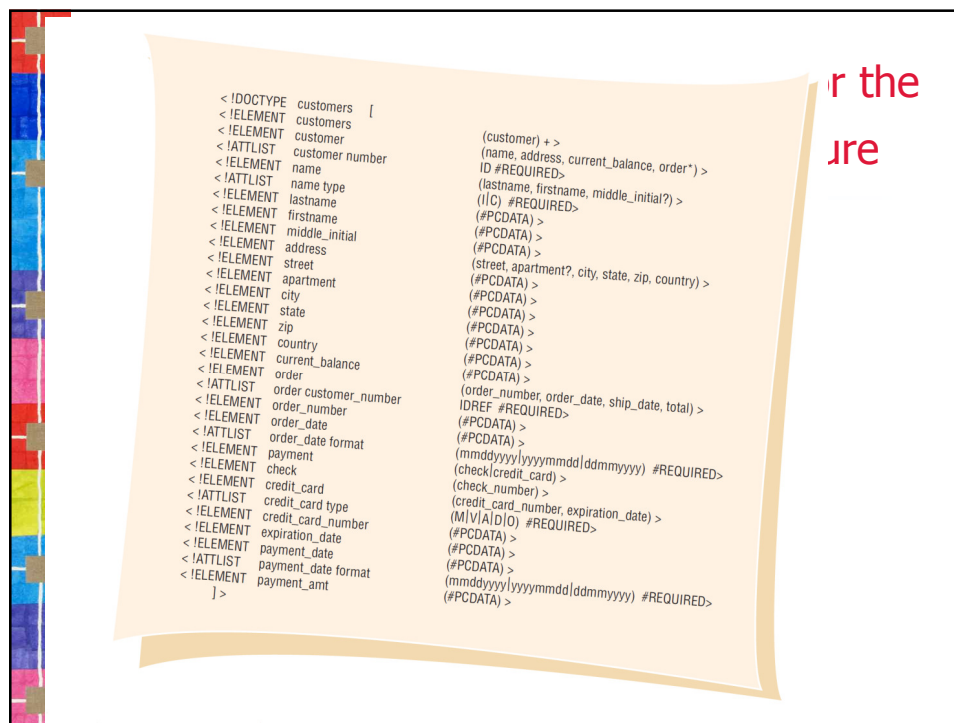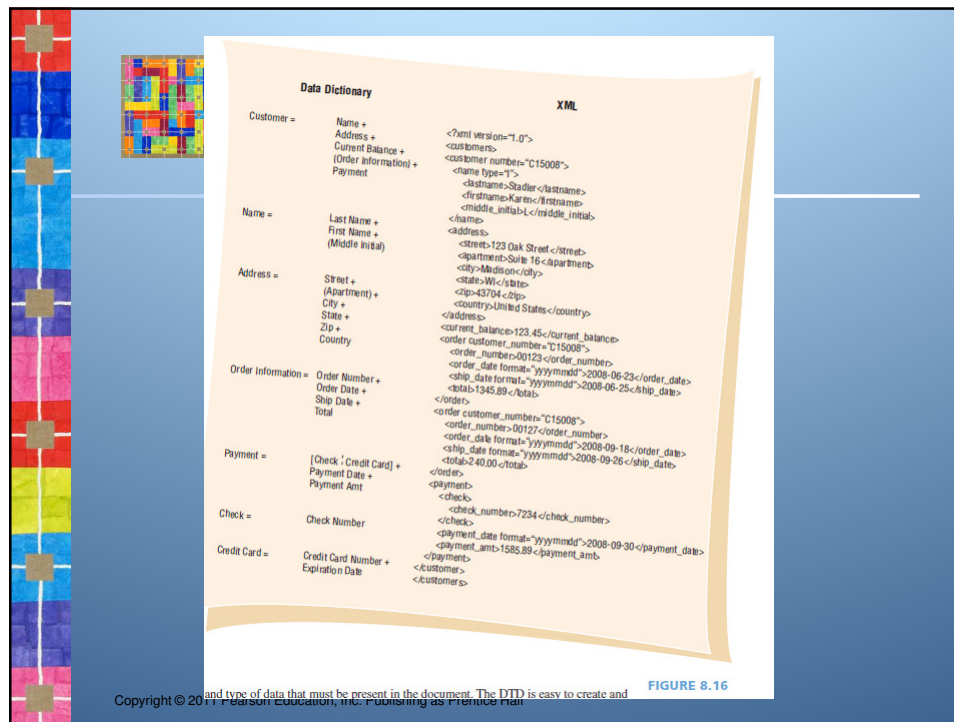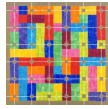
and type of data that must be present in the document. The DTD is easy to create and

FIGURE 8.16

Slide 2

r the ... ure

```
<!DOCTYPE customers  [
<!ELEMENT customers        (customer) + >
<!ELEMENT customer         (name, address, current_balance, order*) >
<!ATTLIST customer number  ID #REQUIRED>
<!ELEMENT name             (lastname, firstname, middle_initial?) >
<!ATTLIST name type        (I|C) #REQUIRED>
<!ELEMENT lastname         (#PCDATA) >
<!ELEMENT firstname        (#PCDATA) >
<!ELEMENT middle_initial   (#PCDATA) >
<!ELEMENT address          (street, apartment?, city, state, zip, country) >
<!ELEMENT street           (#PCDATA) >
<!ELEMENT apartment        (#PCDATA) >
<!ELEMENT city             (#PCDATA) >
<!ELEMENT state            (#PCDATA) >
<!ELEMENT zip              (#PCDATA) >
<!ELEMENT country          (#PCDATA) >
<!ELEMENT current_balance  (#PCDATA) >
<!ELEMENT order            (order_number, order_date, ship_date, total) >
<!ATTLIST order customer_number  IDREF #REQUIRED>
<!ELEMENT order_number     (#PCDATA) >
<!ELEMENT order_date       (#PCDATA) >
<!ATTLIST order_date format  (mmddyyyy|yyyymmdd|ddmmyyyy) #REQUIRED>
<!ELEMENT payment          (check|credit_card) >
<!ELEMENT check            (check_number) >
<!ELEMENT credit_card      (credit_card_number, expiration_date) >
<!ATTLIST credit_card type   (M|V|A|D|O) #REQUIRED>
<!ELEMENT credit_card_number  (#PCDATA) >
<!ELEMENT expiration_date  (#PCDATA) >
<!ELEMENT payment_date     (#PCDATA) >
<!ATTLIST payment_date format  (mmddyyyy|yyyymmdd|ddmmyyyy) #REQUIRED>
<!ELEMENT payment_amt      (#PCDATA) >
] >
```

XML Schemas

- A more precise way to define the content of an XML document
- Includes exact number of times an element may occur
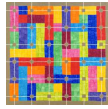- Includes type of data within elements

Summary

- The data dictionary
  - A reference work containing data about data
  - Includes all data items from data flow diagrams
- Repository
  - A larger collection of project information
- Defining data structures
- Defining elements

Summary (Continued)

- Defining data stores
- Data dictionary entries
- Using the data dictionary
- Data dictionary analysis
- Data dictionary to XML