eight

Designing the Human Interface

Blend Images/SuperStock

Chapter Objectives

After studying this chapter, you should be able to:

- Explain the process of designing forms and reports, and the deliverables for their creation.
- Apply the general guidelines for formatting forms and reports.
- Format text, tables, and lists effectively.
- Explain the process of designing interfaces and dialogues, and the deliverables for their creation.
- Describe and apply the general guidelines for interface design, including guidelines for layout design, structuring data-entry fields, providing feedback, and system help.
- Design human-computer dialogues, including the use of dialogue diagramming.
- Discuss interface design guidelines unique to the design of Internet-based electronic commerce systems.

Chapter Preview . . .

Analysts must complete two important activities in the systems design phase, as illustrated in Figure 8-1: designing the human interface and designing databases. In this chapter, you learn guidelines to follow when designing the human-computer interface. In the first section, we describe the process of designing forms and reports and provide guidance on the deliverables produced during this process. Properly formatted segments of information are the building blocks for designing all forms and reports. We present guidelines for formatting information and for designing interfaces and dialogues. Next, we show you a method for representing human-computer dialogues called *dialogue diagramming*. Finally, we close the chapter by examining various human-computer interface design issues for Internet-based applications, specifically as they apply to Pine Valley Furniture's WebStore.

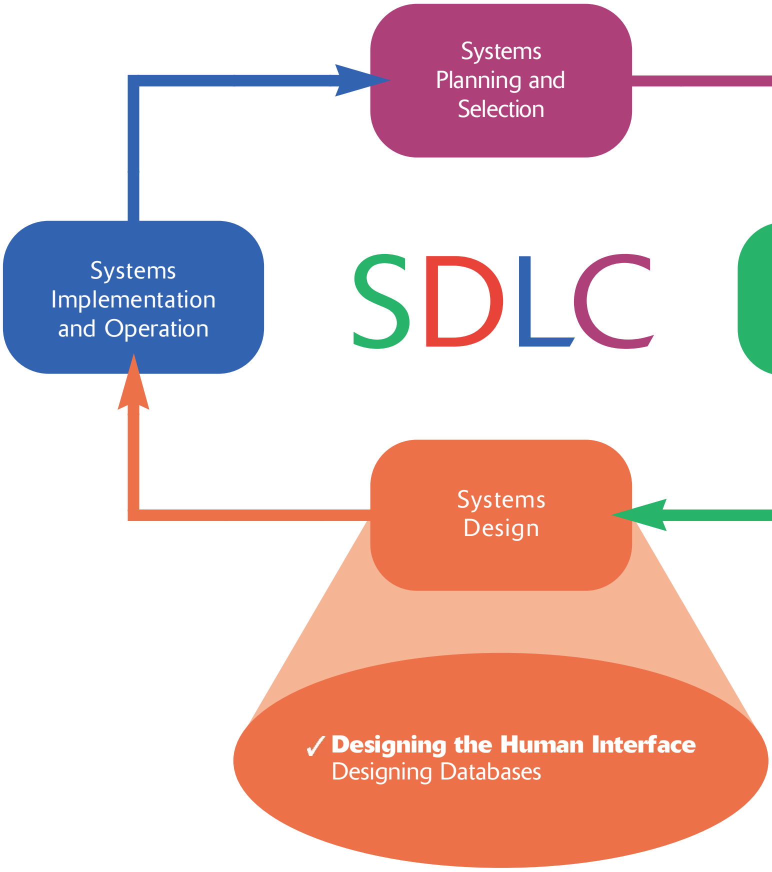

Systems Planning and Selection
Systems Analysis
Systems Design
Systems Implementation and Operation
SDLC
✓ Designing the Human Interface
Designing Databases

FIGURE 8-1
The systems design phase consists of two important activities: designing the human interface and designing databases.

Designing Forms and Reports

System inputs and outputs—forms and reports—are produced at the end of the systems analysis phase of the SDLC. During systems analysis, however, you may not have been concerned with the precise appearance of forms and reports. Instead, you focused on which forms and reports needed to exist and the content they needed to contain. You may have distributed to users the prototypes of forms and reports that emerged during analysis as a way to confirm requirements. Forms and reports are integrally related to the DFD and E-R diagrams developed during requirements structuring. For example, every input form is associated with a data flow entering a process on a DFD, and every output form or report is a data flow produced by a process on a DFD. Therefore, the contents of a form or report correspond to the data elements contained in the associated data flow. Further, the data on all forms and reports must consist of data elements in data stores and on the E-R data model for the application or else be computed from these data elements. (In rare instances, data simply go from system input to system output without being stored within the system.) It is common to discover flaws in DFDs and E-R diagrams as you design forms and reports; these diagrams should be updated as designs evolve.

Form
A business document that contains some predefined data and may include some areas where additional data are to be filled in; typically based on one database record.

If you are unfamiliar with computer-based information systems, it will be helpful to clarify exactly what we mean by a form or report. A **form** is a business document containing some predefined data and often includes some areas where additional data are to be filled in. Most forms have a stylized format and are usually not in simple rows and columns. Examples of business forms are product order forms, employment applications, and class registration sheets. Traditionally, forms have been displayed on a paper medium, but today, video display technology allows us to duplicate the layout of almost any printed form, including an organizational logo or any graphic, on a video display terminal. Forms on a video display may be used for data display or data entry. Additional examples of forms are an electronic spreadsheet, computer sign-on or menu, and an automated teller machine (ATM) transaction layout. On the Internet, form interaction is the standard method of gathering and displaying information when consumers order products, request product information, or query account status.

Report
A business document that contains only predefined data; it is a passive document used only for reading or viewing; typically contains data from many unrelated records or transactions.

A **report** is a business document containing only predefined data; it is a passive document used solely for reading or viewing. Examples of reports are invoices, weekly sales summaries by region and salesperson, and a pie chart of population by age categories. We usually think of a report as printed on paper, but it may be printed to a computer file, a visual display screen, or some other medium such as microfilm. Often a report has rows and columns of data, but a report may consist of any format—for example, mailing labels. Frequently, the differences between a form and a report are subtle. A report is only for reading and often contains data about multiple unrelated records in a computer file. On the other hand, a form typically contains data from only one record or is, at least, based on one record, such as data about one customer, one order, or one student. The guidelines for the design of forms and reports are similar.

The Process of Designing Forms and Reports

Designing forms and reports is a user-focused activity that typically follows a prototyping approach (see Figure 1-12 to review the prototyping method). First, you must gain an understanding of the intended user and task objectives during the requirements determination process. During this process, the intended user must answer several questions that attempt to answer the who, what, when, where, and how related to the creation of all forms or reports, as listed in Table 8-1. Gaining an understanding of these questions is a required first step in the creation of any form or report.

TABLE 8-1: Fundamental Questions When Designing Forms and Reports

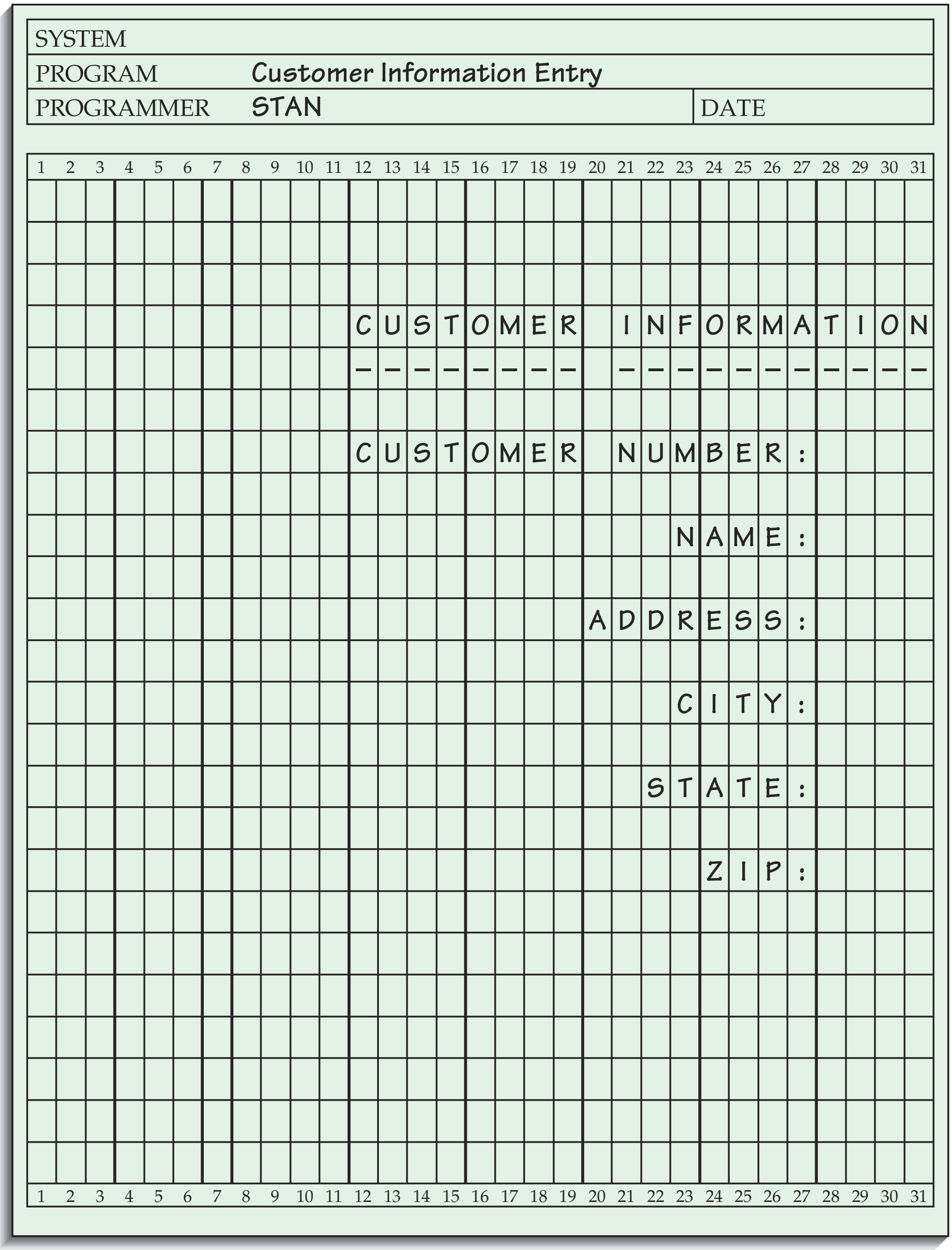
1. Who will use the form or report?
2. What is the purpose of the form or report?
3. When is the form or report needed and used?
4. Where does the form or report need to be delivered and used?
5. How many people need to use or view the form or report?

Understanding the skills and abilities of the users helps you create an effective design. Are your users experienced computer users or novices? What is their educational level, business background, and task-relevant knowledge? Answers to these questions provide guidance for both the format and the content of your designs. Also, what is the purpose of the form or report? What task will users be performing, and what information is needed to complete this task? Other questions are also important to consider. Where will the users be when performing this task? Will users have access to online systems or will they be in the field? How many people will need to use this form or report? If, for example, a report is being produced for a single user, the design requirements and usability assessment will be relatively simple. A design for a larger audience, however, may need to go through a more extensive requirements collection and usability assessment process.

After collecting the initial requirements, you structure and refine this information into an initial prototype. Structuring and refining the requirements are completed without assistance from the users, although you may occasionally need to contact users to clarify some issue overlooked during analysis. Finally, you ask users to review and evaluate the prototype; then they may accept the design or request that changes be made. If changes are needed, repeat the construction-evaluate-refinement cycle until the design is accepted. Usually, several repetitions of this cycle occur during the design of a single form or report. As with any prototyping process, you should make sure that these iterations occur rapidly in order to gain the greatest benefit from this design approach.

The initial prototype may be constructed in numerous environments, including Visual Basic, Java, or HTML. The obvious choice is to employ standard development tools used within your organization. Often, initial prototypes are simply mock screens that are not working modules or systems. Mock screens can also be produced from a word processor, computer graphics design package, or presentation, software. It is important to remember that the focus of this phase within the SDLC is on the *design*—content and layout. How specific forms or reports are implemented (e.g., the programming language or screen painter code) is left for a later stage. Nonetheless, tools for designing forms and reports are rapidly evolving. In the past, inputs and outputs of all types were typically designed by hand on a coding or layout sheet. For example, Figure 8-2 shows the layout of a data input form using a coding sheet.

Although coding sheets are still used, their importance has diminished because of significant changes in system operating environments and the evolution of automated design tools. Prior to the creation of graphical operating environments, for example, analysts designed many inputs and outputs that were 80 columns (characters) by 25 rows, the standard dimensions for most video displays. These limits in screen dimensions are radically different in graphical operating environments such as Mac OS or Windows where font sizes and screen dimensions can often be changed from user to user. Consequently, the creation of new tools and development environments was needed to help analysts and programmers develop these graphical and flexible designs.



SYSTEM

PROGRAM Customer Information Entry

PROGRAMMER STAN DATE

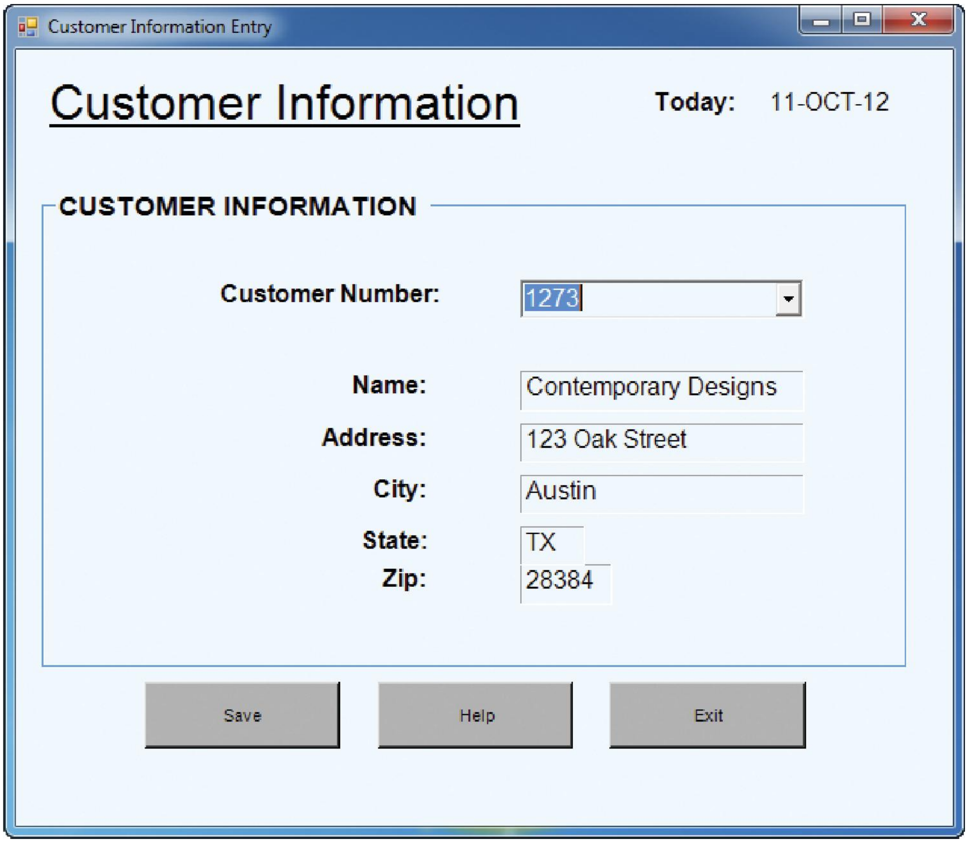
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
											C	U	S	T	O	M	E	R		I	N	F	O	R	M	A	T	I	O	N
											-	-	-	-	-	-	-	-		-	-	-	-	-	-	-	-	-	-	-
											C	U	S	T	O	M	E	R		N	U	M	B	E	R	:				
																						N	A	M	E	:				
																			A	D	D	R	E	S	S	:				
																						C	I	T	Y	:				
																					S	T	A	T	E	:				
																							Z	I	P	:				
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

FIGURE 8-2
The layout of a data input form using a coding sheet.

Figure 8-3 shows an example of the same data input form as designed in Microsoft's Visual Basic.Net. Note the variety of fonts, sizes, and highlighting that was used. Online graphical tools for designing forms and reports are rapidly becoming the standard in most professional development organizations.

Deliverables and Outcomes

Each SDLC activity helps you to construct a system. In order to move from phase to phase, each activity produces some type of deliverable that is used in a later activity. For example, within the systems planning and selection phase of the SDLC, the baseline project plan serves as input to many subsequent SDLC activities. In the case of designing forms and reports, design specifications are


Customer Information Entry

Customer Information

Today: 11-OCT-12

CUSTOMER INFORMATION

Customer Number: 1273

Name: Contemporary Designs

Address: 123 Oak Street

City: Austin

State: TX

Zip: 28384

FIGURE 8-3
A data input screen designed in Microsoft's Visual Basic.Net.

the major deliverables and are inputs to the system implementation and operation phase. Design specifications have three sections:

1. Narrative overview
2. Sample design
3. Testing and usability assessment

PINE
VALLEY
FURNITURE

The narrative overview provides a general overview of the characteristics of the target users, tasks, system, and environmental factors in which the form or report will be used. Its purpose is to explain to those who will actually develop the final form, why this form exists, and how it will be used so that they can make the appropriate implementation decisions. In this section, you list general information and the assumptions that helped shape the design. For example, Figure 8-4 shows an excerpt of a design specification for a Customer Account Status form for Pine Valley Furniture. The first section of the specification, Figure 8-4A, provides a narrative overview containing the information relevant to developing and using the form within PVF. The overview explains the tasks supported by the form, where and when the form is used, characteristics of the people using the form, the technology delivering the form, and other pertinent information. For example, if the form is delivered on a visual display terminal, this section would describe the capabilities of this device, such as navigation and whether it has a touch screen and whether color and a mouse are available.

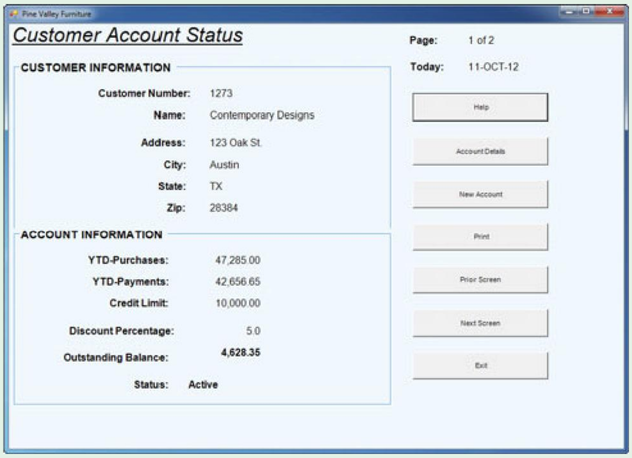
In the second section of the specification, Figure 8-4B, a sample design of the form is shown. This design may be hand-drawn using a coding sheet, although, in most instances, it is developed using standard development tools. Using actual development tools allows the design to be more thoroughly tested and assessed. The final section of the specification, Figure 8-4C, provides all testing and usability assessment information. Some specification information may be irrelevant when designing certain forms and reports. For example, the design of a simple yes/no

FIGURE 8-4
A design specification for a Customer Account Status form for Pine Valley Furniture: (A) The narrative overview containing the information relevant to developing and using the form within PVF, (B) A sample design of the PVF form, (C) Testing and usability assessment information.

(A) Narrative overview

Form: Customer Account Status
Users: Customer account representatives within corporate offices
Task: Assess customer account information: address, account balance, year-to-date purchases and payments, credit limit, discount percentage, and account status.
System: Microsoft Windows
Environment: Standard office environment

(B) Sample design



Customer Account Status

Page: 1 of 2
Today: 11-OCT-12

CUSTOMER INFORMATION

Customer Number: 1273
Name: Contemporary Designs
Address: 123 Oak St.
City: Austin
State: TX
Zip: 28384

ACCOUNT INFORMATION

YTD-Purchases: 47,285.00
YTD-Payments: 42,656.65
Credit Limit: 10,000.00
Discount Percentage: 5.0
Outstanding Balance: 4,628.35
Status: Active

(C) Testing and usability assessment

User-Rated Perceptions (average 14 users):

consistency [1 = consistent to 7 = inconsistent]:	1.52
sufficiency [1 = sufficient to 7 = insufficient]:	1.43
accuracy [1 = accurate to 7 = inaccurate]:	1.67
. . .	

selection form may be so straightforward that no usability assessment is needed. Also, much of the narrative overview may be unnecessary unless intended to highlight some exception that must be considered during implementation.

Formatting Forms and Reports

A wide variety of information can be provided to users of information systems, ranging from text to video to audio. As technology continues to evolve, a greater variety of data types will be used. A definitive set of rules for delivering every type of information to users has yet to be defined because these rules are continuously evolving along with the rapid changes in technology. Research conducted by computer scientists on human-computer interaction has provided numerous general guidelines for formatting information. Many of these guidelines undoubtedly will apply to the formatting of all evolving information types on yet-to-be-determined devices. Keep in mind that designing usable forms and reports requires your active interaction with users. If this single and fundamental activity occurs, you will likely create effective designs.

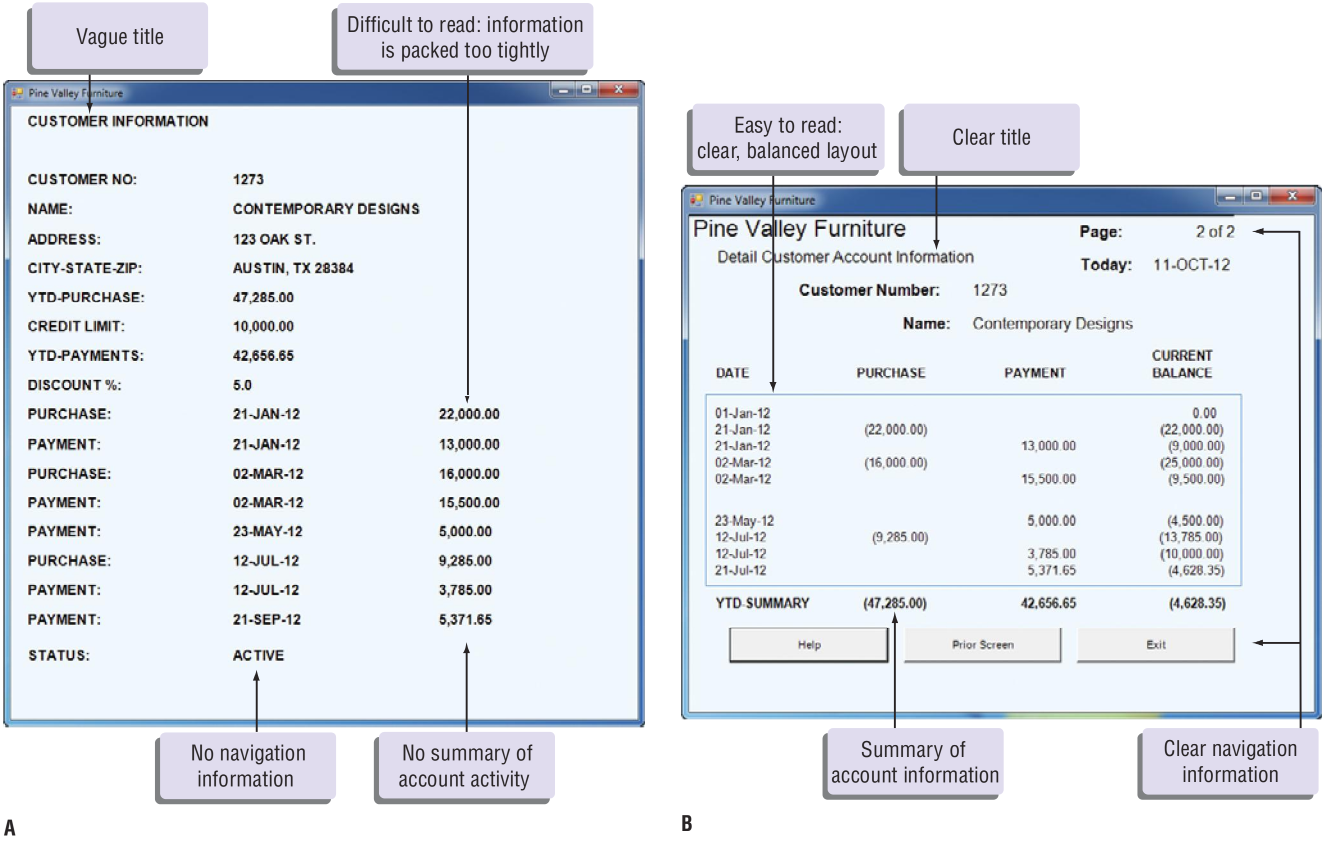
For example, the human-computer interface is one of the greatest challenges for designing mobile applications that run on devices such as the iPhone. In particular, the small video display of these devices presents significant challenges for application designers. Nevertheless, as these and other computing devices evolve and gain popularity, standard guidelines will emerge to make the process of designing interfaces much less challenging.

General Formatting Guidelines Over the past several years, industry and academic researchers have investigated how information formatting influences individual task performance and perceptions of usability. Through this work, several guidelines for formatting information have emerged, as highlighted in Table 8-2. These guidelines reflect some of the general truths of formatting most types of information. The differences between a well-designed form or report and a poorly designed one often will be obvious. For example, Figure 8-5A shows a poorly designed form for viewing a current account balance for a PVF customer. Figure 8-5B is a better design, incorporating several general guidelines from Table 8-2.

The first major difference between the two forms has to do with the title. The title in Figure 8-5A (Customer Information) is ambiguous, whereas the title in Figure 8-5B (Detail Customer Account Information) clearly and specifically describes the contents of the form. The form in Figure 8-5B also includes the date (October 11, 2012) the form was generated so that, if printed, it will be clear to the reader when this occurred. Figure 8-5A displays the account status and customer address, information that is extraneous to viewing the current account balance, which is the intent of the form and provides information that is not in the most useful format for the user. For example, Figure 8-5A provides all customer data, as well as account transactions and a summary of year-to-date purchases and payments. The form does not, however, provide the current outstanding balance of the account, leaving the reader to perform a manual calculation. The layout of information between the two forms also varies in balance and information density. Gaining an understanding of the skills of the

TABLE 8-2: Guidelines for Designing Forms and Reports

Guideline	Description
Use meaningful titles	Clear and specific titles describing content and use of form or report
	Revision date or code to distinguish a form or report from prior versions
	Current date that identifies when the form or report was generated
	Valid date that identifies on what date (or time) the data in the form or report were accurate
Include meaningful information	Only needed information displayed
	Information provided in a usable manner without modification
Balance the layout	Information balanced on the screen or page
	Adequate spacing and margins used
	All data and entry fields clearly labeled
Design an easy navigation system	Clearly show how to move forward and backward
	Clearly show where you are (e.g., page 1 of 3)
	Notify user of the last page of a multipage sequence


Vague title
Difficult to read: information is packed too tightly
No navigation information
No summary of account activity
A
Easy to read: clear, balanced layout
Clear title
Summary of account information
Clear navigation information
B
Help
Prior Screen
Exit

Pine Valley Furniture

CUSTOMER INFORMATION

CUSTOMER NO:	1273	
NAME:	CONTEMPORARY DESIGNS	
ADDRESS:	123 OAK ST.	
CITY-STATE-ZIP:	AUSTIN, TX 28384	
YTD-PURCHASE:	47,285.00	
CREDIT LIMIT:	10,000.00	
YTD-PAYMENTS:	42,656.65	
DISCOUNT %:	5.0	
PURCHASE:	21-JAN-12	22,000.00
PAYMENT:	21-JAN-12	13,000.00
PURCHASE:	02-MAR-12	16,000.00
PAYMENT:	02-MAR-12	15,500.00
PAYMENT:	23-MAY-12	5,000.00
PURCHASE:	12-JUL-12	9,285.00
PAYMENT:	12-JUL-12	3,785.00
PAYMENT:	21-SEP-12	5,371.65
STATUS:	ACTIVE	

Pine Valley Furniture

Pine Valley Furniture
Detail Customer Account Information

Page: 2 of 2
Today: 11-OCT-12

Customer Number: 1273
Name: Contemporary Designs

DATE	PURCHASE	PAYMENT	CURRENT BALANCE
01-Jan-12			0.00
21-Jan-12	(22,000.00)		(22,000.00)
21-Jan-12		13,000.00	(9,000.00)
02-Mar-12	(16,000.00)		(25,000.00)
02-Mar-12		15,500.00	(9,500.00)
23-May-12		5,000.00	(4,500.00)
12-Jul-12	(9,285.00)		(13,785.00)
12-Jul-12		3,785.00	(10,000.00)
21-Jul-12		5,371.65	(4,628.35)
YTD-SUMMARY	**(47,285.00)**	**42,656.65**	**(4,628.35)**

FIGURE 8-5
Contrast of a poorly designed and a well-designed form: (A) A poorly designed form for viewing a current account balance for a PVF customer, (B) A better design that incorporates several general guidelines from Table 8-2.

intended system users and the tasks they will be performing is invaluable when constructing a form or report. By following these general guidelines, your chances of creating effective forms and reports will be enhanced. In the next sections, we discuss specific guidelines for highlighting information, displaying text, and presenting numeric tables and lists.

Highlighting Information As display technologies continue to improve, a greater variety of methods will be available to highlight information. Table 8-3 lists the most commonly used methods for highlighting information. Given this vast array of options, it is important to consider how highlighting can be used to enhance an output without being a distraction. In general, highlighting should be used sparingly to draw the user to or away from certain information and to group together related information. In several situations, highlighting can be a valuable technique for conveying special information:

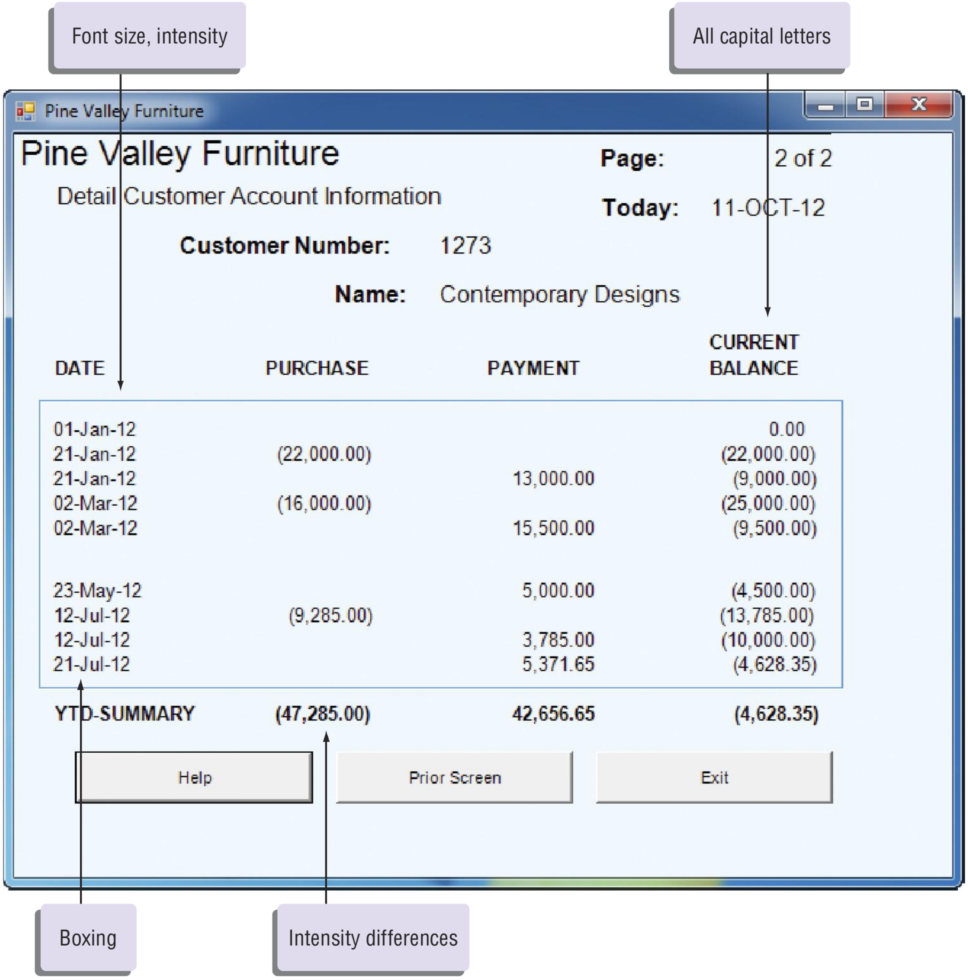
- Notifying users of errors in data entry or processing
- Providing warnings to users regarding possible problems, such as unusual data values or an unavailable device
- Drawing attention to keywords, commands, high-priority messages, and data that have changed or gone outside normal operating ranges

Highlighting techniques can be used singularly or in tandem, depending upon the level of emphasis desired by the designer. Figure 8-6 shows a form where several types of highlighting are used. In this example, columns clarify different categories of data; capital letters and different fonts distinguish labels from actual data; and bolding is used to draw attention to important data.

TABLE 8-3: Methods of Highlighting

Blinking and audible tones
Color differences
Intensity differences
Size differences
Font differences
Reverse video
Boxing
Underlining
All capital letters
Offsetting the position of nonstandard information

Highlighting should be used conservatively. For example, blinking and audible tones should be used only to highlight critical information requiring the user's immediate response. Once a response is made, these highlights should be turned off. Additionally, highlighting methods should be consistently selected and used based upon the level of importance of the emphasized information. It is also important to examine how a particular highlighting method appears on



Pine Valley Furniture

Detail Customer Account Information

Page: 2 of 2

Today: 11-OCT-12

Customer Number: 1273

Name: Contemporary Designs

DATE	PURCHASE	PAYMENT	CURRENT BALANCE
01-Jan-12			0.00
21-Jan-12	(22,000.00)		(22,000.00)
21-Jan-12		13,000.00	(9,000.00)
02-Mar-12	(16,000.00)		(25,000.00)
02-Mar-12		15,500.00	(9,500.00)
23-May-12		5,000.00	(4,500.00)
12-Jul-12	(9,285.00)		(13,785.00)
12-Jul-12		3,785.00	(10,000.00)
21-Jul-12		5,371.65	(4,628.35)
YTD-SUMMARY	**(47,285.00)**	**42,656.65**	**(4,628.35)**

FIGURE 8-6
A form in which several types of highlighting are used.

all possible output devices that could be used with the system. For example, some color combinations may convey appropriate information on one display configuration but wash out and reduce legibility on another.

Recent advances in the development of graphical operating environments such as Windows, Mac OS, or Linux provide designers with some standard highlighting guidelines. However, because these guidelines are continuously evolving, they are often quite vague and leave a great deal of control in the hands of the systems developer. To realize the benefits of using standard graphical operating environments—such as reduced user training time and interoperability among systems—you must be disciplined in how you use highlighting.

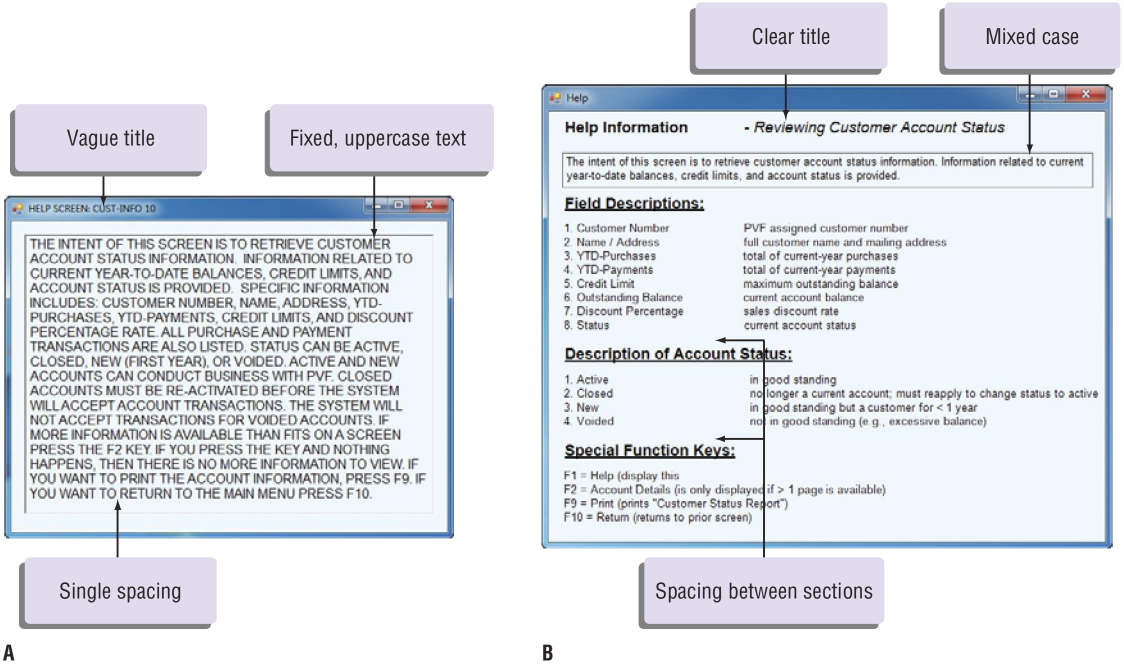
Displaying Text In business-related systems, textual output is becoming increasingly important as text-based applications, such as electronic mail, blogs, and information services (e.g., Dow Jones Industrial Average stock index), are more widely used. The display and formatting of system help screens, which often contain lengthy textual descriptions and examples, is one example of textual data that can benefit from following the simple guidelines that have emerged from systems design research. These guidelines appear in Table 8-4. The first one is simple: You should display text using common writing conventions such as mixed upper- and lowercase and appropriate punctuation. For large blocks of text, and if space permits, text should be double spaced. However, if the text is short, or rarely used, it may make sense to use single spacing and place a blank line between each paragraph. You should also left-justify text with a ragged right margin—research shows that a ragged right margin makes it easier to find the next line of text when reading than when text is both left- and right-justified.

When displaying textual information, you should also be careful not to hyphenate words between lines or use obscure abbreviations and acronyms. Users may not know whether the hyphen is a significant character if it is used to continue words across lines. Information and terminology that are not widely understood by the intended users may significantly influence the usability of the system. Thus, you should use abbreviations and acronyms only if they are significantly shorter than the full text and are commonly known by the intended system users. Figure 8-7 shows two versions of a help screen from an application system at PVF. Figure 8-7A shows many violations of the general guidelines for displaying text, whereas Figure 8-7B shows the same information following the general guidelines. Formatting guidelines for the entry of text and alphanumeric data are also very important and will be discussed later in the chapter.

Designing Tables and Lists Unlike textual information, where context and meaning are derived through reading, the context and meaning of tables and lists are derived from the format of the information. Consequently, the usability of information displayed in tables and alphanumeric lists is likely to be much more influenced by effective layout than most other types of information display. As with

TABLE 8-4: Guidelines for Displaying Text

Case	Display text in mixed upper- and lowercase and use conventional punctuation.
Spacing	Use double spacing if space permits. If not, place a blank line between paragraphs.
Justification	Left-justify text and leave a ragged right margin.
Hyphenation	Do not hyphenate words between lines.
Abbreviations	Use abbreviations and acronyms only when they are widely understood by users and are significantly shorter than the full text.



HELP SCREEN: CUST-INFO 10

THE INTENT OF THIS SCREEN IS TO RETRIEVE CUSTOMER ACCOUNT STATUS INFORMATION. INFORMATION RELATED TO CURRENT YEAR-TO-DATE BALANCES, CREDIT LIMITS, AND ACCOUNT STATUS IS PROVIDED. SPECIFIC INFORMATION INCLUDES: CUSTOMER NUMBER, NAME, ADDRESS, YTD-PURCHASES, YTD-PAYMENTS, CREDIT LIMITS, AND DISCOUNT PERCENTAGE RATE. ALL PURCHASE AND PAYMENT TRANSACTIONS ARE ALSO LISTED. STATUS CAN BE ACTIVE, CLOSED, NEW (FIRST YEAR), OR VOIDED. ACTIVE AND NEW ACCOUNTS CAN CONDUCT BUSINESS WITH PVF. CLOSED ACCOUNTS MUST BE RE-ACTIVATED BEFORE THE SYSTEM WILL ACCEPT ACCOUNT TRANSACTIONS. THE SYSTEM WILL NOT ACCEPT TRANSACTIONS FOR VOIDED ACCOUNTS. IF MORE INFORMATION IS AVAILABLE THAN FITS ON A SCREEN PRESS THE F2 KEY. IF YOU PRESS THE KEY AND NOTHING HAPPENS, THEN THERE IS NO MORE INFORMATION TO VIEW. IF YOU WANT TO PRINT THE ACCOUNT INFORMATION, PRESS F9. IF YOU WANT TO RETURN TO THE MAIN MENU PRESS F10.

Help

Help Information - *Reviewing Customer Account Status*

The intent of this screen is to retrieve customer account status information. Information related to current year-to-date balances, credit limits, and account status is provided.

Field Descriptions:

1. Customer Number	PVF assigned customer number
2. Name / Address	full customer name and mailing address
3. YTD-Purchases	total of current-year purchases
4. YTD-Payments	total of current-year payments
5. Credit Limit	maximum outstanding balance
6. Outstanding Balance	current account balance
7. Discount Percentage	sales discount rate
8. Status	current account status

Description of Account Status:

1. Active	in good standing
2. Closed	no longer a current account; must reapply to change status to active
3. New	in good standing but a customer for < 1 year
4. Voided	not in good standing (e.g., excessive balance)

Special Function Keys:

F1 = Help (display this
F2 = Account Details (is only displayed if > 1 page is available)
F9 = Print (prints "Customer Status Report")
F10 = Return (returns to prior screen)

FIGURE 8-7
Contrasting two help screens from an application system at PVF: (A) A poorly designed help screen with many violations of the general guidelines for displaying text, (B) An improved design for a help screen.

the display of textual information, tables and lists can also be greatly enhanced by following a few simple guidelines. These are summarized in Table 8-5.

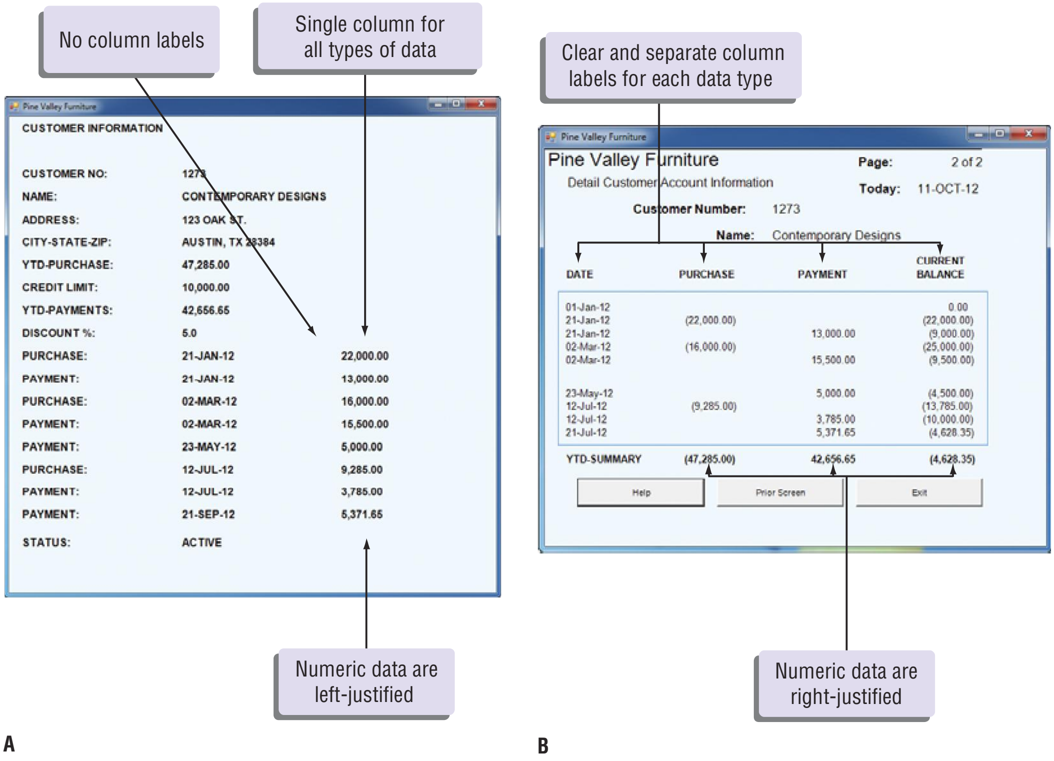
Figure 8-8 displays two versions of a form design from a Pine Valley Furniture application system that displays customer year-to-date transaction information in a table format. Figure 8-8A displays the information without consideration of the guidelines presented in Table 8-5, and Figure 8-8B (only page 2 of 2 is shown) displays this information after consideration of these guidelines.

One key distinction between these two display forms relates to labeling. The information reported in Figure 8-8B has meaningful labels that stand out more clearly compared to the display in Figure 8-8A. Transactions are sorted by date and transaction type, and numeric data are right-justified and aligned by decimal point in Figure 8-8B, which helps to facilitate scanning. Adequate space is left between columns, and blank lines are inserted after every five rows in Figure 8-8B to help ease the finding and reading of information. Such spacing also provides room for users to annotate data that catch their attention. Using the guidelines presented in Table 8-5 helped create an easy-to-read layout of the information for the user.

Most of the guidelines in Table 8-5 are rather obvious, but this and other tables serve as a quick reference to validate that your form and report designs will be usable. It is beyond our scope here to discuss each of these guidelines, but you should read each carefully and think about why it is appropriate. For example, why should labels be repeated on subsequent screens and pages (the first guideline in Table 8-5)? One explanation is that pages may be separated or copied, and the original labels will no longer be readily accessible to the reader of the data. Why should long alphanumeric data (see the last guideline) be broken into small groups? (If you have a credit card or bank check, look at how your account number is displayed.) Two reasons are that the characters will be easier to remember as you read and type them, and this approach provides a natural and consistent place to pause when you speak them over the phone (e.g., when you are placing a phone order for products in a catalog).

TABLE 8-5: General Guidelines for Displaying Tables and Lists

Guideline	Description
Use meaningful labels	All columns and rows should have meaningful labels.
	Labels should be separated from other information by using highlighting.
	Redisplay labels when the data extend beyond a single screen or page.
Format columns, rows, and text	Sort in a meaningful order (e.g., ascending, descending, or alphabetical).
	Place a blank line between every five rows in long columns.
	Similar information displayed in multiple columns should be sorted vertically (i.e., read from top to bottom, not left to right).
	Columns should have at least two spaces between them.
	Allow white space on printed reports for user to write notes.
	Use a single typeface, except for emphasis.
	Use same family of typefaces within and across displays and reports.
	Avoid overly fancy fonts.
Format numeric, textual, and alphanumeric data	Right-justify *numeric data* and align columns by decimal points or other delimiter.
	Left-justify *textual data*. Use short line length, usually 30 to 40 characters per line (this guideline is what newspapers use, and it is easy to speed-read).
	Break long sequences of *alphanumeric data* into small groups of three to four characters each.



No column labels

Single column for all types of data

CUSTOMER INFORMATION

CUSTOMER NO:	1273	
NAME:	CONTEMPORARY DESIGNS	
ADDRESS:	123 OAK ST.	
CITY-STATE-ZIP:	AUSTIN, TX 28384	
YTD-PURCHASE:	47,285.00	
CREDIT LIMIT:	10,000.00	
YTD-PAYMENTS:	42,656.65	
DISCOUNT %:	5.0	
PURCHASE:	21-JAN-12	22,000.00
PAYMENT:	21-JAN-12	13,000.00
PURCHASE:	02-MAR-12	16,000.00
PAYMENT:	02-MAR-12	15,500.00
PAYMENT:	23-MAY-12	5,000.00
PURCHASE:	12-JUL-12	9,285.00
PAYMENT:	12-JUL-12	3,785.00
PAYMENT:	21-SEP-12	5,371.65
STATUS:	ACTIVE	

Numeric data are left-justified

Clear and separate column labels for each data type

Pine Valley Furniture

Detail Customer Account Information

Page: 2 of 2

Today: 11-OCT-12

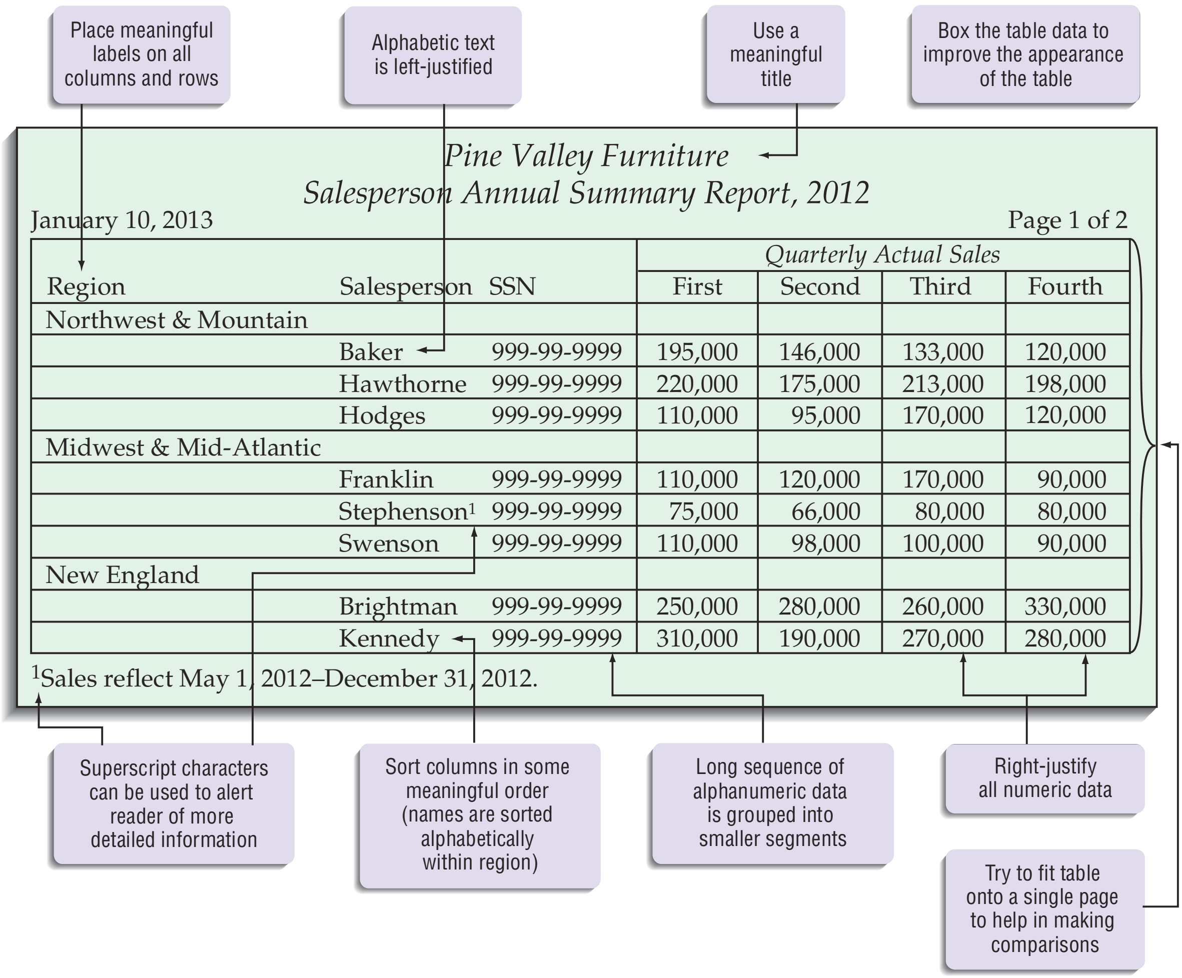
Customer Number: 1273

Name: Contemporary Designs

DATE	PURCHASE	PAYMENT	CURRENT BALANCE
01-Jan-12			0.00
21-Jan-12	(22,000.00)		(22,000.00)
21-Jan-12		13,000.00	(9,000.00)
02-Mar-12	(16,000.00)		(25,000.00)
02-Mar-12		15,500.00	(9,500.00)
23-May-12		5,000.00	(4,500.00)
12-Jul-12	(9,285.00)		(13,785.00)
12-Jul-12		3,785.00	(10,000.00)
21-Jul-12		5,371.65	(4,628.35)
YTD-SUMMARY	(47,285.00)	42,656.65	(4,628.35)

Numeric data are right-justified

FIGURE 8-8
Contrasting two Pine Valley Furniture forms: (A) A poorly designed form, (B) An improved design form.



Place meaningful labels on all columns and rows

Alphabetic text is left-justified

Use a meaningful title

Box the table data to improve the appearance of the table

Pine Valley Furniture
Salesperson Annual Summary Report, 2012

January 10, 2013 Page 1 of 2

			Quarterly Actual Sales			
Region	Salesperson	SSN	First	Second	Third	Fourth
Northwest & Mountain						
	Baker	999-99-9999	195,000	146,000	133,000	120,000
	Hawthorne	999-99-9999	220,000	175,000	213,000	198,000
	Hodges	999-99-9999	110,000	95,000	170,000	120,000
Midwest & Mid-Atlantic						
	Franklin	999-99-9999	110,000	120,000	170,000	90,000
	Stephenson[1]	999-99-9999	75,000	66,000	80,000	80,000
	Swenson	999-99-9999	110,000	98,000	100,000	90,000
New England						
	Brightman	999-99-9999	250,000	280,000	260,000	330,000
	Kennedy	999-99-9999	310,000	190,000	270,000	280,000

[1] Sales reflect May 1, 2012–December 31, 2012.

Superscript characters can be used to alert reader of more detailed information

Sort columns in some meaningful order (names are sorted alphabetically within region)

Long sequence of alphanumeric data is grouped into smaller segments

Right-justify all numeric data

Try to fit table onto a single page to help in making comparisons

FIGURE 8-9
Tabular report illustrating good report design guidelines.

When you design the display of numeric information, you must determine whether a table or a graph should be used. In general, tables are best when the user's task involves finding an individual data value from a larger data set, whereas line and bar graphs are more appropriate for analyzing data changes over time. For example, if the marketing manager for Pine Valley Furniture needed to review the actual sales of a particular salesperson for a particular quarter, a tabular report such as the one shown in Figure 8-9 would be most useful. This report has been annotated to emphasize good report design practices. The report has both a printed date as well as a clear indication, as part of the report title, of the period over which the data apply. Sufficient white space also provides some room for users to add personal comments and observations. Often, to provide such white space, a report must be printed in landscape, rather than portrait, orientation. Alternatively, if the marketing manager wished to compare the overall sales performance of each sales region, a line or bar graph would be more appropriate, as illustrated in Figure 8-10.

Paper versus Electronic Reports When a report is produced on paper rather than on a computer display, you need to consider some additional things. For example, laser printers (especially color laser printers) and ink-jet printers allow you to produce a report that looks exactly as it does on the display screen. Thus, when using these types of printers, you can follow our general design guidelines to create a report with high usability. However, other types of

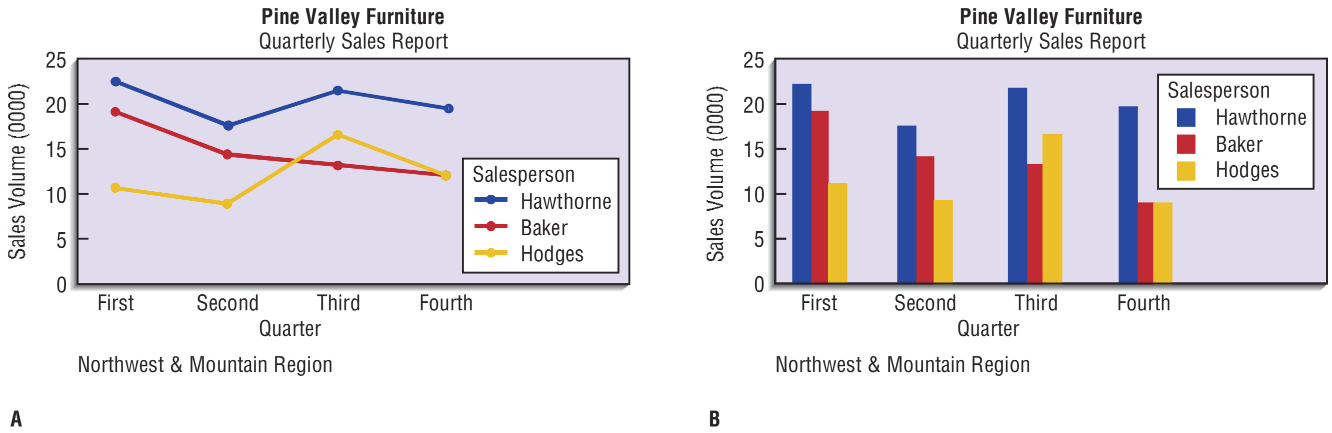


FIGURE 8-10
Graphs showing quarterly sales at Pine Valley Furniture: (A) Line graph, (B) Bar graph.

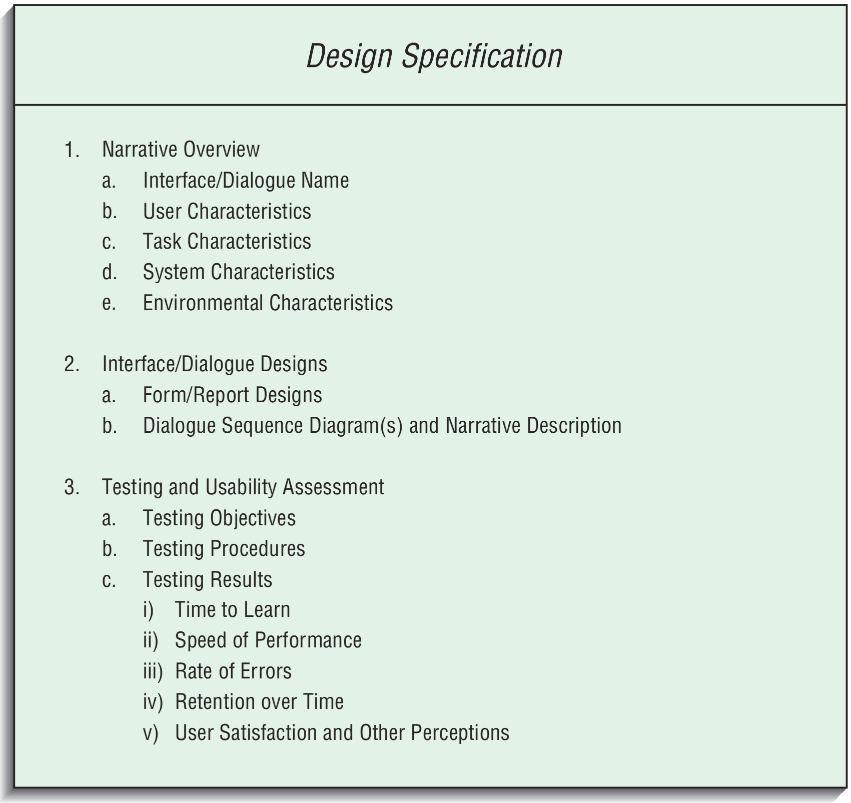
printers cannot closely reproduce the display screen image onto paper. For example, many business reports are produced using high-speed impact printers that produce characters and a limited range of graphics by printing a fine pattern of dots. The advantages of impact printers are that they are fast, reliable, and relatively inexpensive. Their drawbacks are that they have a limited ability to produce graphics and have a somewhat lower print quality. In other words, they are good at rapidly producing reports that contain primarily alphanumeric information but cannot exactly replicate a screen report onto paper. For this reason, impact printers are mostly used for producing large batches of reports, such as a batch of phone bills for your telephone company, on a wide range of paper widths and types. When designing reports for impact printers, you use a coding sheet similar to the one displayed in Figure 8-2, although coding sheets for designing printer reports typically can have up to 132 columns. Like the process for designing all forms and reports, you follow a prototyping process and carefully control the spacing of characters in order to produce a high-quality report. However, unlike other form and report designs, you may be limited in the range of formatting, text types, and highlighting options. Nonetheless, you can easily produce a highly usable report of any type if you carefully and creatively use the available formatting options.

Designing Interfaces and Dialogues

Interface and dialogue design focuses on how information is provided to and captured from users. Dialogues are analogous to a conversation between two people. The grammatical rules followed by each person during a conversation are analogous to the human-computer interface. The design of interfaces and dialogues involves defining the manner in which humans and computers exchange information. A good human-computer interface provides a uniform structure for finding, viewing, and invoking the different components of a system. In this section we describe how to design interfaces and dialogues.

The Process of Designing Interfaces and Dialogues

Similar to designing forms and reports, the process of designing interfaces and dialogues is a user-focused activity. You follow a prototyping methodology of iteratively collecting information, constructing a prototype, assessing usability, and making refinements. To design usable interfaces and dialogues, you must answer the same who, what, when, where, and how questions used to guide the



Design Specification

1. Narrative Overview
   - a. Interface/Dialogue Name
   - b. User Characteristics
   - c. Task Characteristics
   - d. System Characteristics
   - e. Environmental Characteristics

2. Interface/Dialogue Designs
   - a. Form/Report Designs
   - b. Dialogue Sequence Diagram(s) and Narrative Description

3. Testing and Usability Assessment
   - a. Testing Objectives
   - b. Testing Procedures
   - c. Testing Results
     - i) Time to Learn
     - ii) Speed of Performance
     - iii) Rate of Errors
     - iv) Retention over Time
     - v) User Satisfaction and Other Perceptions

FIGURE 8-11
An outline for a design specification for interfaces and dialogues.

design of forms and reports (see Table 8-1). Thus, this process parallels that of designing forms and reports.

Deliverables and Outcomes

The deliverable and outcome from system interface and dialogue design is the creation of a design specification. This specification is similar to the specification produced for form and report designs—with one exception. Recall that the design specification for forms and reports had three sections (see Figure 8-4):

1. Narrative overview
2. Sample design
3. Testing and usability assessment

For interface and dialogue designs, one additional subsection is included: a section outlining the dialogue sequence—the ways a user can move from one display to another. Later in the chapter you will learn how to design a dialogue sequence by using dialogue diagramming. An outline for a design specification for interfaces and dialogues is shown in Figure 8-11.

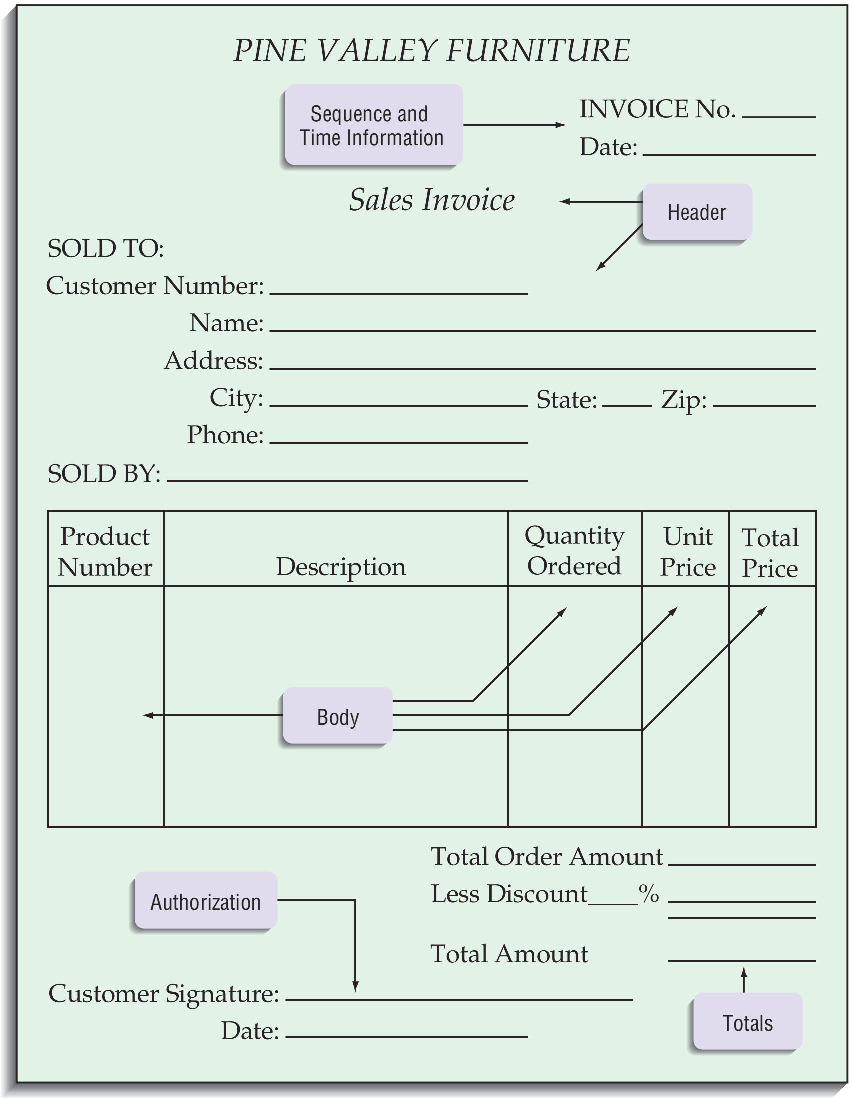
Designing Interfaces

In this section we discuss the design of interface layouts. This discussion provides guidelines for structuring and controlling data-entry fields, providing feedback, and designing online help. Effective interface design requires you to gain a thorough understanding of each of these concepts.

Designing Layouts To ease user training and data recording, use standard formats for computer-based forms and reports similar to paper-based forms and reports for recording or reporting information. A typical paper-based form for

PINE VALLEY FURNITURE

FIGURE 8-12
Paper-based form for reporting customer sales activity at Pine Valley Furniture.



PINE VALLEY FURNITURE

INVOICE No. ______
Date: ______

Sales Invoice

SOLD TO:
Customer Number: ______
Name: ______
Address: ______
City: ______ State: ____ Zip: ______
Phone: ______
SOLD BY: ______

Product Number	Description	Quantity Ordered	Unit Price	Total Price

Total Order Amount ______
Less Discount____% ______
Total Amount ______

Customer Signature: ______
Date: ______

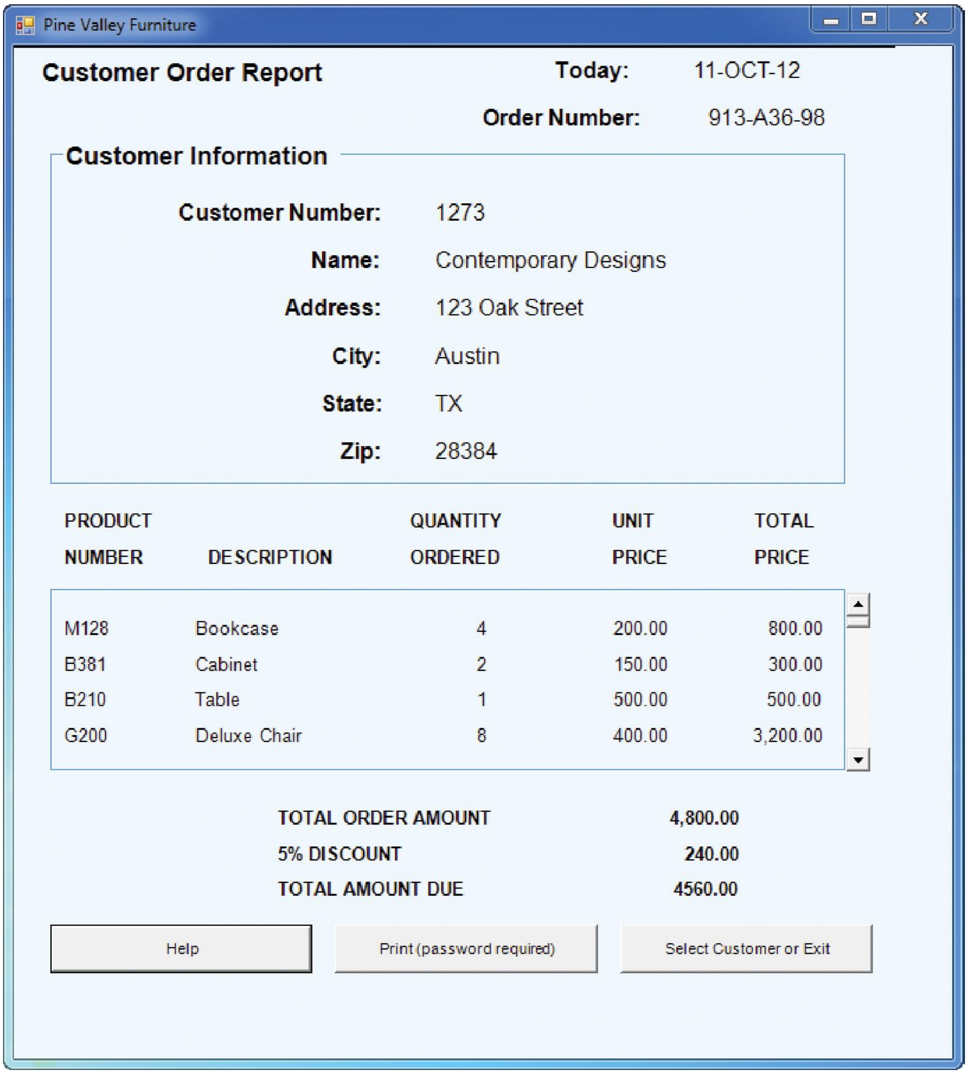
Sequence and Time Information
Header
Body
Authorization
Totals

reporting customer sales activity is shown in Figure 8-12. This form has several general areas common to most forms:

- Header information
- Sequence and time-related information
- Instruction or formatting information
- Body or data details
- Totals or data summary
- Authorization or signatures
- Comments

In many organizations, data are often first recorded on paper-based forms and then later recorded within application systems. When designing layouts to record or display information on paper-based forms, try to make both as similar as possible. Additionally, data-entry displays should be consistently formatted across applications to speed data entry and reduce errors. Figure 8-13 shows an equivalent computer-based form to the paper-based form shown in Figure 8-12.

The design of between-field navigation is another item to consider when designing the layout of computer-based forms. Because you can control the



Customer Order Report

Today: 11-OCT-12

Order Number: 913-A36-98

Customer Information

Customer Number:	1273
Name:	Contemporary Designs
Address:	123 Oak Street
City:	Austin
State:	TX
Zip:	28384

PRODUCT NUMBER	DESCRIPTION	QUANTITY ORDERED	UNIT PRICE	TOTAL PRICE
M128	Bookcase	4	200.00	800.00
B381	Cabinet	2	150.00	300.00
B210	Table	1	500.00	500.00
G200	Deluxe Chair	8	400.00	3,200.00

TOTAL ORDER AMOUNT	4,800.00
5% DISCOUNT	240.00
TOTAL AMOUNT DUE	4560.00

FIGURE 8-13
Computer-based form for reporting customer sales activity at Pine Valley Furniture.

sequence for users to move between fields, standard screen navigation should flow from left-to-right and top-to-bottom just as when you work on paper-based forms. For example, Figure 8-14 contrasts the flow between fields on a form used to record business contacts. Figure 8-14A uses a consistent left-to-right, top-to-bottom flow. Figure 8-14B uses a flow that is nonintuitive. When appropriate, you should also group data fields into logical categories with labels describing the contents of the category. Areas of the screen not used for data entry or commands should be inaccessible to the user.

When designing the navigation procedures within your system, flexibility and consistency are primary concerns. Users should be able to move freely forward and backward or to any desired data-entry fields. Users should be able to navigate each form in the same way or in as similar a manner as possible. Additionally, data should not usually be permanently saved by the system until the user makes an explicit request to do so. This option allows the user to abandon a data-entry screen, back up, or move forward without adversely impacting the contents of the permanent data.

Consistency extends to the selection of keys and commands. Assign each key or command only one function. This assignment should be consistent throughout the entire system and across systems, if possible. Depending upon the application, various types of functional capabilities will be required to provide smooth navigation and data entry. Table 8-6 provides a checklist for testing

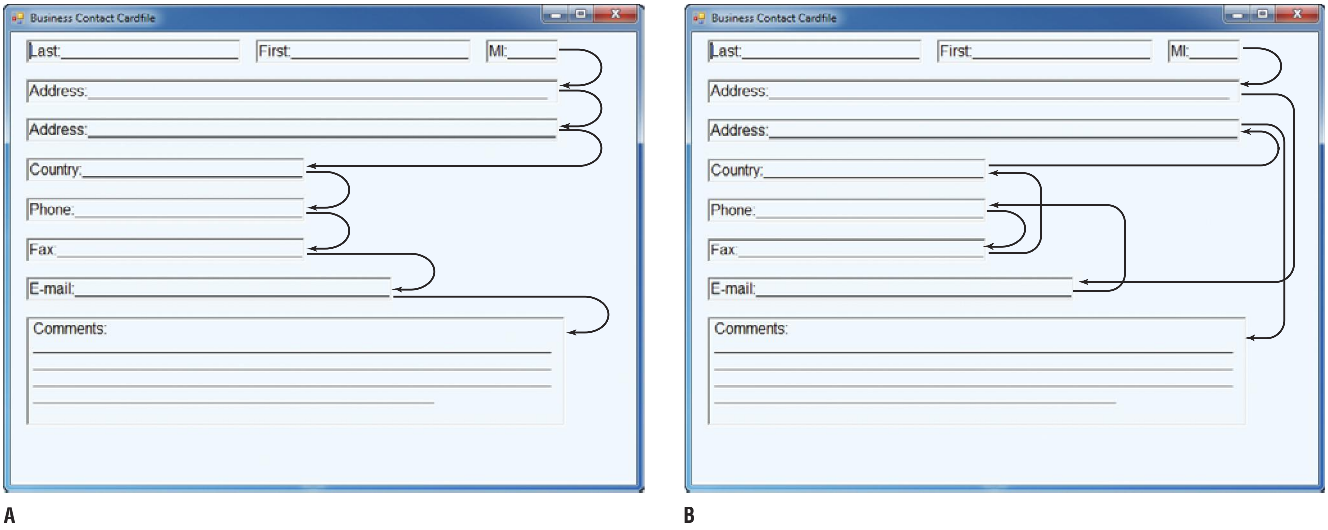


FIGURE 8-14
Contrasting the navigation flow within a data-entry form: (A) Proper flow between data-entry fields with a consistent left-to-right, top-to-bottom flow, (B) Poor flow between data-entry fields with inconsistent flow.

the functional capabilities for providing smooth and easy navigation within a form. For example, a good interface design provides a consistent way for moving the cursor to different places on the form, editing characters and fields, moving among form displays, and obtaining help. These functions may be provided by keystrokes, mouse, menu, or function keys. It is possible that, for a single application, not all capabilities listed in Table 8-6 may be needed in order

TABLE 8-6: Checklist for Validating the Usability of User Interface

Cursor-Control Capabilities

Move the cursor forward to the next data field.

Move the cursor backward to the previous data field.

Move the cursor to the first, last, or some other designated data field.

Move the cursor forward one character in a field.

Move the cursor backward one character in a field.

Editing Capabilities

Delete the character to the left of the cursor.

Delete the character under the cursor.

Delete the whole field.

Delete data from the whole form (empty the form).

Exit Capabilities

Transmit the screen to the application program.

Move to another screen/form.

Confirm the saving of edits or go to another screen/form.

Help Capabilities

Get help on a data field.

Get help on a full screen/form.

Source: Based on J. S. Dumas (1988). *Designing User Interfaces for Software*. Upper Saddle River, NJ: Prentice Hall.

to create a good user interface. Yet, the capabilities that are used should be consistently applied to provide an optimal user environment. Table 8-6 provides you with a checklist for validating the usability of user interface designs.

Structuring Data Entry You should consider several guidelines when structuring data-entry fields on a form. These guidelines are listed in Table 8-7. The first is simple, yet, is often violated by designers. To minimize data-entry errors and user frustration, never require the user to enter information that is already available within the system or information that can be easily computed by the system. For example, never require the user to enter the current date and time, because each of these values can be easily retrieved from the computer system's internal calendar and clock. By allowing the system to do these tasks, the user simply confirms that the calendar and clock are working properly.

Other guidelines are equally important. For example, suppose that a bank customer is repaying a loan on a fixed schedule with equal monthly payments. Each month when a payment is sent to the bank, a clerk needs to record that the payment has been received into a loan-processing system. Within such a system, default values for fields should be provided whenever appropriate, which allows the clerk to enter specific data into the system only when the customer pays more or less than the scheduled amount. In all other cases, the clerk simply verifies that the check is for the default amount provided by the system and presses a single key to confirm the receipt of payment.

When entering data, do not require the user to specify the dimensional units of a particular value, for example, whether an amount is in dollars or a weight is in tons. Use field formatting and the data-entry prompt to make clear the type of data being requested. In other words, place a caption describing the data to be entered adjacent to each data field so that the user knows what type of data is being requested. As with the display of information, all data entered onto a form should automatically justify in a standard format (e.g., date, time, money).

TABLE 8-7: Guidelines for Structuring Data-Entry Fields

Entry	Never request data that are already online or that can be computed, for example, do not request customer data on an order form if that data can be retrieved from the database, and do not request extended prices that can be computed from quantity sold and unit prices.
Defaults	Always provide default values when appropriate, for example, assume today's date for a new sales invoice, or use the standard product price unless overridden.
Units	Make clear the type of data units requested for entry, for example, indicate quantity in tons, dozens, pounds, etc.
Replacement	Use character replacement when appropriate, for example, allow the user to look up the value in a table or automatically fill in the value once the user enters enough significant characters.
Captioning	Always place a caption adjacent to fields; see Table 8-8 for caption options.
Format	Provide formatting examples when appropriate, for example, automatically show standard embedded symbols, decimal points, credit symbols, or dollar signs.
Justify	Automatically justify data entries; numbers should be right-justified and aligned on decimal points, and text should be left-justified.
Help	Provide context-sensitive help when appropriate, for example, provide a hot key, such as the F1 key, that opens the help system on an entry that is most closely related to where the cursor is on the display.

TABLE 8-8: Display Design Options for Entering Text

<table>
<tr><th>Options</th><th>Example</th></tr>
<tr><td>Line caption</td><td>Phone Number ( ) -</td></tr>
<tr><td>Drop caption</td><td>( ) -
Phone Number</td></tr>
<tr><td>Boxed caption</td><td>Phone Number</td></tr>
<tr><td>Delimited characters</td><td>|(| | | |)| | | |-| | | | |
Phone Number</td></tr>
<tr><td>Check-off boxes</td><td>Method of payment (check one)
❑ Check
❑ Cash
❑ Credit card: Type</td></tr>
</table>

Table 8-8 illustrates display design options for printed forms. For data entry on video display terminals, highlight the area in which text is entered so that the exact number of characters per line and number of lines are clearly shown. You can also use check-off boxes or radio buttons to allow users to choose standard textual responses. Use data-entry controls to ensure that the proper type of data (alphabetic or numeric, as required) is entered. Data-entry controls are discussed next.

Controlling Data Input One objective of interface design is to reduce data-entry errors. As data are entered into an information system, steps must be taken to ensure that the input is valid. As a systems analyst, you must anticipate the types of errors users may make and design features into the system's interfaces to avoid, detect, and correct data-entry mistakes. Several types of data errors are summarized in Table 8-9. Data errors can occur from appending extra data onto a field, truncating characters off a field, transcripting the wrong characters into a field, or transposing one or more characters within a field. Systems designers have developed numerous tests and techniques for detecting invalid data before saving or transmission, thus improving the likelihood that data will be valid. Table 8-10 summarizes these techniques. These tests and techniques are often incorporated into both data-entry screens and when data are transferred from one computer to another.

Correcting erroneous data is much easier to accomplish before it is permanently stored in a system. Online systems can notify a user of input problems as data are being entered. When data are processed online as events occur, it is much less likely that data-validity errors will occur and not be caught. In an online system, most problems can be easily identified and resolved before permanently saving data to a storage device using many of the techniques

TABLE 8-9: Types of Data Errors

Data Error	Description
Appending	Adding additional characters to a field
Truncating	Losing characters from a field
Transcripting	Entering invalid data into a field
Transposing	Reversing the sequence of one or more characters in a field

TABLE 8-10: Techniques Used by Systems Designers to Detect Data Errors before Saving or Transmission

Validation Test	Description
Class or composition	Test to ensure that data are of proper type (e.g., all numeric, all alphabetic, alphanumeric)
Combinations	Test to see that value combinations of two or more data fields are appropriate or make sense (e.g., does the quantity sold make sense given the type of product?)
Expected values	Test to see whether data are what is expected (e.g., match with existing customer names, payment amount, etc.)
Missing data	Test for existence of data items in all fields of a record (e.g., is there a quantity field on each line item of a customer order?)
Pictures/templates	Test to ensure that data conform to a standard format (e.g., are hyphens in the right places for a student ID number?)
Range	Test to ensure data are within a proper range of values (e.g., is a student's grade-point average between 0 and 4.0?)
Reasonableness	Test to ensure data are reasonable for situation (e.g., pay rate for a specific type of employee)
Self-checking digits	Technique by which extra digits, derived using a standard formula (see Figure 8-15), are added to a numeric field before transmission and checked after transmission
Size	Test for too few or too many characters (e.g., is social security number exactly nine digits?)
Values	Test to make sure values come from a set of standard values (e.g., two-letter state codes)

described in Table 8-10. However, in systems where data inputs are stored and entered (or transferred) in batches, the identification and notification of errors are more difficult. Batch processing systems can, however, reject invalid inputs and store them in a log file for later resolution.

Most of the straightforward tests and techniques shown in Table 8-10 are widely used. Some can be handled by data-management technologies, such as a database management system (DBMS), to ensure that they are applied for all data-maintenance operations. If a DBMS cannot perform these tests, then you must design the tests into program modules. Self-checking digits, shown in Figure 8-15, is an example of a sophisticated program. The figure provides a description and an outline of how to apply the technique. A short example then shows how a check digit is added to a field before data entry or transfer. Once entered or transferred, the check digit algorithm is again applied to the field to "check" whether the check digit received obeys the calculation. If it does, it is likely (but not guaranteed, because two different values could yield the same check digit) that no data transmission or entry error occurred. If not equal, then some type of error occurred.

In addition to validating the data values entered into a system, controls must be established to verify that all input records are correctly entered and processed only once. A common method used to enhance the validity of entering batches of data records is to create an **audit trail** of the entire sequence of data entry, processing, and storage. In such an audit trail, the actual sequence, count, time, source location, and human operator are recorded in a separate transaction log in the event of a data input or processing error. If an error occurs, corrections can be made by reviewing the contents of the log. Detailed

Audit trail
A record of the sequence of data entries and the date of those entries.

FIGURE 8-15
How check digits are calculated.

Description	Techniques where extra digits are added to a field to assist in verifying its accuracy
Method	1. Multiply each digit of a numeric field by weighting factor (e.g., 1,2,1,2, . . .). 2. Sum the results of weighted digits. 3. Divide sum by modulus number (e.g., 10). 4. Subtract remainder of division from modulus number to determine check digit. 5. Append check digits to field.
Example	Assume a numeric part number of: 12473 1–2. Multiply each digit of part number by weighting factor from right to left and sum the results of weighted digits: 1 2 4 7 3 ×1 ×2 ×1 ×2 ×1 1 + 4 + 4 + 14 + 3 = 26 3. Divide sum by modulus number. 26/10 = 2 remainder 6 4. Subtract remainder from modulus number to determine check digit. check digit = 10 – 6 = 4 5. Append check digits to field. Field value with appended check digit = 124734

logs of data inputs not only are useful for resolving batch data-entry errors and system audits, but also serve as a powerful method for performing backup and recovery operations in the case of a catastrophic system failure.

Providing Feedback When you talk with friends, you expect them to give you feedback by nodding and replying to your questions and comments. Without feedback, you would be concerned that they were not listening. Similarly, when designing system interfaces, providing appropriate feedback is an easy way to make a user's interaction more enjoyable; not providing feedback is a sure way to frustrate and confuse. System feedback can consist of three types:

1. Status information
2. Prompting cues
3. Error and warning messages

1. *Status Information.* Providing status information is a simple technique for keeping users informed of what is going on within a system. For example, relevant status information, such as displaying the current customer name or time, placing appropriate titles on a menu or screen, and identifying the number of screens following the current one (e.g., Screen 1 of 3), all provide needed feedback to the user. Providing status information during processing operations is especially important if the operation takes longer than a second or two. For example, when opening a file, you might display, "Please wait while I open the file," or when performing a large calculation, flash the message "Working . . ." to the user. Further, it is important to tell the user that besides working, the system has accepted the user's input and the input was in the correct form. Sometimes it is important to give the user a chance to obtain more feedback. For example, a function key could toggle between showing

a "Working . . ." message and giving more specific information as each intermediate step is accomplished. Providing status information reassures users that nothing is wrong and makes them feel in command of the system, not vice versa.

2. *Prompting Cues.* A second feedback method is to display prompting cues. When prompting the user for information or action, it is useful to be specific in your request. For example, suppose a system prompted users with the following request:

   READY FOR INPUT: ______

   With such a prompt, the designer assumes that the user knows exactly what to enter. A better design would be specific in its request, possibly providing an example, default values, or formatting information. An improved prompting request might be as follows:

   Enter the customer account number (123-456-7):____-___-_

3. *Error and Warning Messages.* A final method available to you for providing system feedback is using error and warning messages. Following a few simple guidelines can greatly improve the usefulness of these messages. First, make messages specific and free of error codes and jargon. Additionally, messages should never scold the user but attempt to guide the user toward a resolution. For example, a message might say, "No customer record found for that customer ID. Please verify that digits were not transposed." Messages should be in user, not computer, terms. Terms such as *end of file*, *disk I/O error*, or *write protected* may be too technical and not helpful for many users. Multiple messages can be useful so that a user can get more detailed explanations if wanted or needed. Also, make sure error messages appear in roughly the same format and placement each time so that they are recognized as error messages and not as some other information. Examples of bad and good messages are provided in Table 8-11. Use these guidelines to provide useful feedback in your designs. A special type of feedback is answering help requests from users. This important topic is described next.

Providing Help Designing a help system is one of the most important interface design issues you will face. When designing help, you need to put yourself in the user's place. When accessing help, the user likely does not know what to do next, does not understand what is being requested, or does not know how the requested information needs to be formatted. A user requesting help is much like a ship in distress, sending an SOS. In Table 8-12, we provide our SOS guidelines for the design of system help: Simplify, Organize, and Show. Our first

TABLE 8-11: Examples of Poor and Improved Error Messages

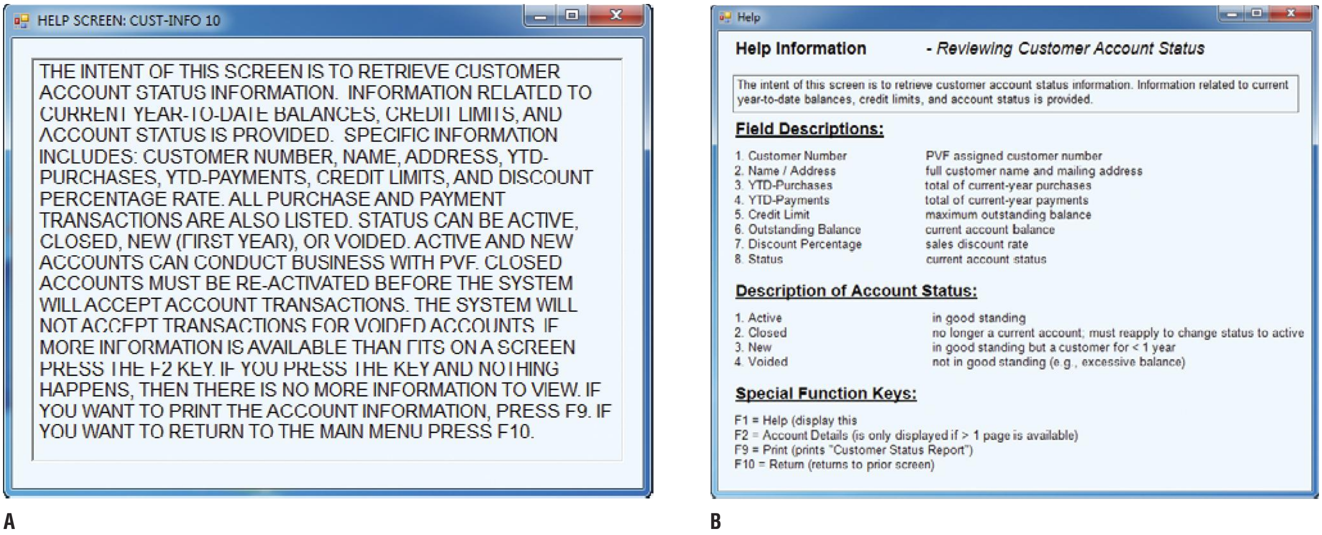
Poor Error Messages	Improved Error Messages
ERROR 56 OPENING FILE	The file name you typed was not found. Press F2 to list valid file names.
WRONG CHOICE	Please enter an option from the menu.
DATA ENTRY ERROR	The prior entry contains a value outside the range of acceptable values. Press F9 for list of acceptable values.
FILE CREATION ERROR	The file name you entered already exists. Press F10 if you want to overwrite it. Press F2 if you want to save it with a new name.

TABLE 8-12: Guidelines for Designing System Help

Guideline	Explanation
Simplify	Use short, simple wording, common spelling, and complete sentences. Give users only what they need to know, with ability to find additional information.
Organize	Use lists to break information into manageable pieces.
Show	Provide examples of proper use and the outcomes of such use.

guideline, *simplify*, suggests that help messages should be short, to the point, and use words that users can understand. The second guideline, *organize*, means that the information in help messages should be easy for users to absorb. Long paragraphs of text are often difficult for people to understand. A better design organizes lengthy information in a manner easier for users to digest through the use of bulleted and ordered lists. Finally, it is often useful to explicitly *show* users how to perform an operation and the outcomes of procedural steps. Figure 8-16 contrasts the designs of two help screens, one that employs our guidelines and one that does not.

Many commercially available systems provide extensive system help. For example, Table 8-13 lists the range of help available in a popular electronic spreadsheet. Many systems are also designed so that users can vary the level of detail provided. Help may be provided at the system level, screen or form level, and individual field level. The ability to provide field-level help is often referred to as *context-sensitive* help. For some applications, providing context-sensitive help for all system options is a tremendous undertaking that is virtually a project in itself. If you do decide to design an extensive help system with many levels of detail, you must be sure that you know exactly what the user needs help with, or your efforts may confuse users more than help them. After leaving a help screen, users should always return back to where they were prior to



THE INTENT OF THIS SCREEN IS TO RETRIEVE CUSTOMER ACCOUNT STATUS INFORMATION. INFORMATION RELATED TO CURRENT YEAR-TO-DATE BALANCES, CREDIT LIMITS, AND ACCOUNT STATUS IS PROVIDED. SPECIFIC INFORMATION INCLUDES: CUSTOMER NUMBER, NAME, ADDRESS, YTD-PURCHASES, YTD-PAYMENTS, CREDIT LIMITS, AND DISCOUNT PERCENTAGE RATE. ALL PURCHASE AND PAYMENT TRANSACTIONS ARE ALSO LISTED. STATUS CAN BE ACTIVE, CLOSED, NEW (FIRST YEAR), OR VOIDED. ACTIVE AND NEW ACCOUNTS CAN CONDUCT BUSINESS WITH PVF. CLOSED ACCOUNTS MUST BE RE-ACTIVATED BEFORE THE SYSTEM WILL ACCEPT ACCOUNT TRANSACTIONS. THE SYSTEM WILL NOT ACCEPT TRANSACTIONS FOR VOIDED ACCOUNTS. IF MORE INFORMATION IS AVAILABLE THAN FITS ON A SCREEN PRESS THE F2 KEY. IF YOU PRESS THE KEY AND NOTHING HAPPENS, THEN THERE IS NO MORE INFORMATION TO VIEW. IF YOU WANT TO PRINT THE ACCOUNT INFORMATION, PRESS F9. IF YOU WANT TO RETURN TO THE MAIN MENU PRESS F10.

Help Information *- Reviewing Customer Account Status*

The intent of this screen is to retrieve customer account status information. Information related to current year-to-date balances, credit limits, and account status is provided.

Field Descriptions:

1. Customer Number	PVF assigned customer number
2. Name / Address	full customer name and mailing address
3. YTD-Purchases	total of current-year purchases
4. YTD-Payments	total of current-year payments
5. Credit Limit	maximum outstanding balance
6. Outstanding Balance	current account balance
7. Discount Percentage	sales discount rate
8. Status	current account status

Description of Account Status:

1. Active	in good standing
2. Closed	no longer a current account; must reapply to change status to active
3. New	in good standing but a customer for < 1 year
4. Voided	not in good standing (e.g., excessive balance)

Special Function Keys:

F1 = Help (display this
F2 = Account Details (is only displayed if > 1 page is available)
F9 = Print (prints "Customer Status Report")
F10 = Return (returns to prior screen)

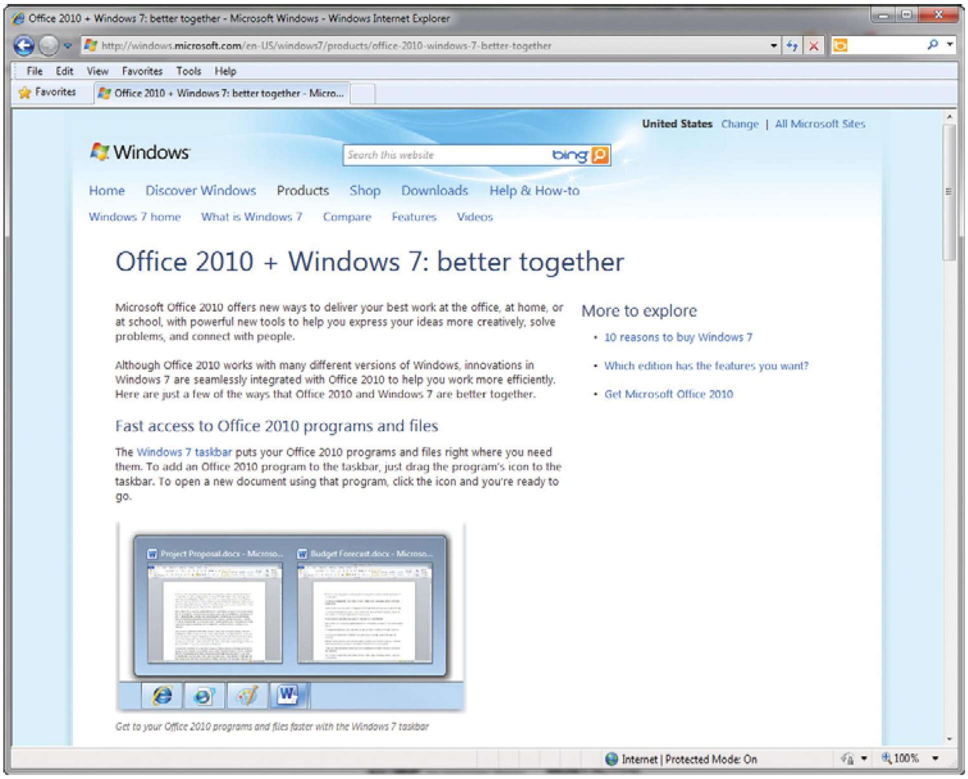
FIGURE 8-16
Contrasting help screens: (A) A poorly designed help screen, (B) An improved design for a help screen.

TABLE 8-13: Types of Help

Type of Help	Example of Question
Help on help	How do I get help?
Help on concepts	What is a customer record?
Help on procedures	How do I update a record?
Help on messages	What does "Invalid File Name" mean?
Help on menus	What does "Graphics" mean?
Help on function keys	What does each function key do?
Help on commands	How do I use the "Cut" and "Paste" commands?
Help on words	What do "merge" and "sort" mean?

requesting help. If you follow these simple guidelines, you will likely design a highly usable help system.

As with the construction of menus, many programming environments provide powerful tools for designing system help. For example, Microsoft's HTML Help SDK allows you to construct hypertext-based help systems quickly. In this environment, you use a text editor to construct help pages that can be easily linked to other pages containing related or more specific information. Linkages are created by embedding special characters into the text document that make words hypertext buttons—that is, direct linkages—to additional information. The HTML Help SDK transforms the text document into a hypertext document. For example, Figure 8-17 shows a hypertext-based help screen from Microsoft.


Office 2010 + Windows 7: better together - Microsoft Windows - Windows Internet Explorer
http://windows.microsoft.com/en-US/windows7/products/office-2010-windows-7-better-together
File Edit View Favorites Tools Help
Favorites
Office 2010 + Windows 7: better together - Micro...
United States Change | All Microsoft Sites
Windows
Search this website
bing
Home Discover Windows Products Shop Downloads Help & How-to
Windows 7 home What is Windows 7 Compare Features Videos
Project Proposal.docx - Microso...
Budget Forecast.docx - Microso...
Internet | Protected Mode: On
100%

Office 2010 + Windows 7: better together

Microsoft Office 2010 offers new ways to deliver your best work at the office, at home, or at school, with powerful new tools to help you express your ideas more creatively, solve problems, and connect with people.

Although Office 2010 works with many different versions of Windows, innovations in Windows 7 are seamlessly integrated with Office 2010 to help you work more efficiently. Here are just a few of the ways that Office 2010 and Windows 7 are better together.

Fast access to Office 2010 programs and files

The Windows 7 taskbar puts your Office 2010 programs and files right where you need them. To add an Office 2010 program to the taskbar, just drag the program's icon to the taskbar. To open a new document using that program, click the icon and you're ready to go.

Get to your Office 2010 programs and files faster with the Windows 7 taskbar

More to explore

- 10 reasons to buy Windows 7
- Which edition has the features you want?
- Get Microsoft Office 2010

FIGURE 8-17
Hypertext-based help system from Microsoft.

Source: Copyright © 2011 Microsoft Corporation. All rights reserved. Protected by the copyright laws of the United States and international treaties.

Hypertext-based help systems have become the standard environment for most commercial operating environments for two primary reasons. First, standardizing system help across applications eases user training. Second, hypertext allows users to selectively access the level of help they need, making it easier to provide effective help for both novice and experienced users within the same system.

Designing Dialogues

Dialogue
The sequence of interaction between a user and a system.

The process of designing the overall sequences that users follow to interact with an information system is called *dialogue design*. A **dialogue** is the sequence in which information is displayed to and obtained from a user. As with other design processes, designing dialogues is a three-step process:

1. Designing the dialogue sequence
2. Building a prototype
3. Assessing usability

The primary design guideline for designing dialogues is consistency; dialogues need to be consistent in sequence of actions, keystrokes, and terminology. In other words, use the same labels for the same operations on all screens and the same location of the same information on all displays.

One example of these guidelines concerns removing data from a database or file (see the Reversal entry in Table 8-14). It is good practice to display the information that will be deleted before making a permanent change to the file. For

TABLE 8-14: Guidelines for the Design of Human-Computer Dialogues

Guideline	Explanation
Consistency	Dialogues should be consistent in sequence of actions, keystrokes, and terminology (e.g., use the same labels for the same operations on all screens and the same location of the same information on all displays).
Shortcuts and sequence	Allow advanced users to take shortcuts using special keys (e.g., CTRL-C to copy highlighted text). A natural sequence of steps should be followed (e.g., enter first name before last name, if appropriate).
Feedback	Feedback should be provided for every user action (e.g., confirm that a record has been added, rather than simply putting another blank form on the screen).
Closure	Dialogues should be logically grouped and have a beginning, middle, and end (e.g., the last in the sequence of screens should indicate that there are no more screens).
Error handling	All errors should be detected and reported; suggestions on how to proceed should be made (e.g., suggest why such errors occur and what the user can do to correct the error). Synonyms for certain responses should be accepted (e.g., accept either "t," "T," or "TRUE").
Reversal	Dialogues should, when possible, allow the user to reverse actions (e.g., undo a deletion); data should not be deleted without confirmation (e.g., display all the data for a record the user has indicated is to be deleted).
Control	Dialogues should make the user (especially an experienced user) feel in control of the system (e.g., provide a consistent response time at a pace acceptable to the user).
Ease	Dialogues should provide simple means for users to enter information and navigate between screens (e.g., provide means to move forward, backward, and to specific screens, such as first and last).

Source: Based on B. Shneiderman, C. Plaisant, M. Cohen, and S. Jacobs (2009). *Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th Edition*. Reading, MA: Addison-Wesley.

example, if the customer service representative wanted to remove a customer from the database, the system should ask only for the customer ID in order to retrieve the correct customer account. Once found, and before allowing the confirmation of the deletion, the system should display the account information. For actions making permanent changes to system data files and when the action is not commonly performed, many system designers use the double-confirmation technique by which the users must confirm their intention twice before being allowed to proceed.

Designing the Dialogue Sequence Your first step in dialogue design is to define the sequence. In other words, you must have a clear understanding of the user, task, technological, and environmental characteristics when designing dialogues. Suppose that the marketing manager at Pine Valley Furniture (PVF) wants sales and marketing personnel to be able to review the year-to-date transaction activity for any PVF customer. After talking with the manager, you both agree that a typical dialogue between a user and the Customer Information System for obtaining this information might proceed as follows:

PINE
VALLEY
FURNITURE

1. Request to view individual customer information
2. Specify the customer of interest
3. Select the year-to-date transaction summary display
4. Review customer information
5. Leave system

As a designer, once you understand how a user wishes to use a system, you can then transform these activities into a formal dialogue specification.

A method for designing and representing dialogues is **dialogue diagramming.** Dialogue diagrams, illustrated in Figure 8-18, have only one symbol, a box with three sections; each box represents one display (which might be a full screen or a specific form or window) within a dialogue. The three sections of the box are used as follows:

Dialogue diagramming
A formal method for designing and representing human-computer dialogues using box-and-line diagrams.

1. Top: Contains a unique display reference number used by other displays for referencing it
2. Middle: Contains the name or description of the display
3. Bottom: Contains display reference numbers that can be accessed from the current display

All lines connecting the boxes within dialogue diagrams are assumed to be bidirectional and, thus, do not need arrowheads to indicate direction. With this capability, users are allowed to always move forward and backward between adjacent displays. If you desire only unidirectional flows within a dialogue, arrowheads should be placed at one end of the line. Within a dialogue diagram, you can easily represent the sequencing of displays, the selection of one display over another, or the repeated use of a single display (e.g., a data-entry display). These three concepts—sequence, selection, and iteration—are illustrated in Figure 8-19.

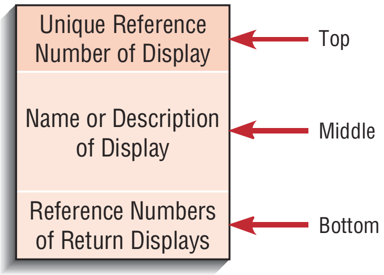
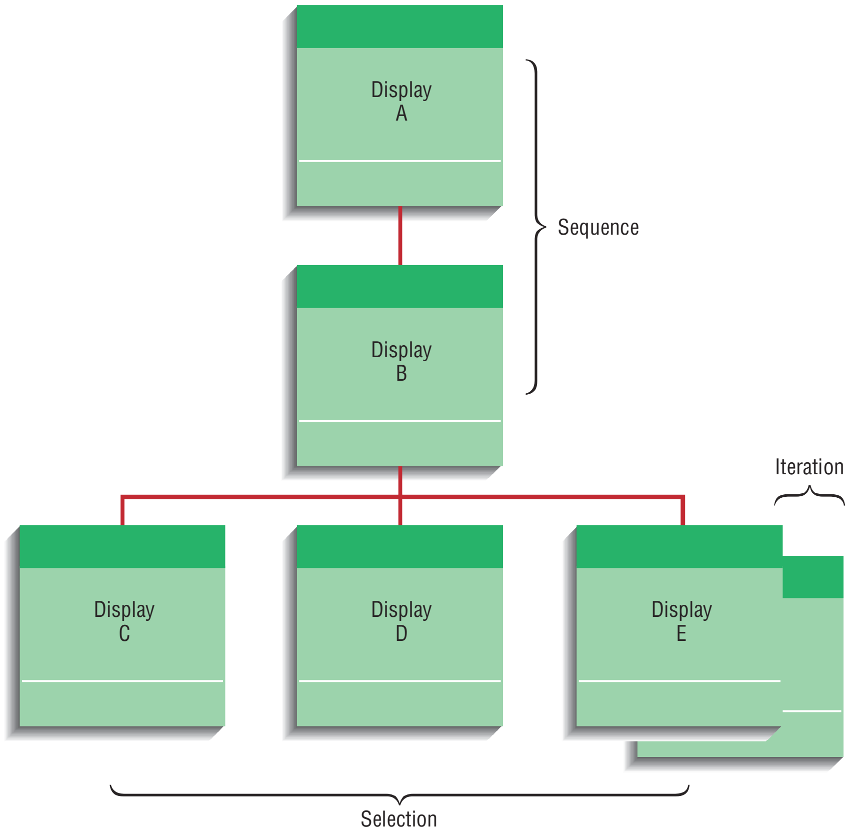

Unique Reference Number of Display
Name or Description of Display
Reference Numbers of Return Displays
Top
Middle
Bottom

FIGURE 8-18
A dialogue-diagramming box has three sections.

FIGURE 8-19
Dialogue diagram illustrating sequence, selection, and iteration.



Continuing with our PVF example, Figure 8-20 shows a partial dialogue diagram for processing the marketing manager's request. In this diagram, the analyst placed the request to view year-to-date customer information within the context of the overall Customer Information System. The user must first gain access to the system through a log-on procedure (item 0). If log-on is successful, a main menu is displayed that has four items (item 1). Once the user selects the Individual Customer Information display (item 2), control is transferred to the Select Customer display (item 2.1). After a customer is selected, the user is presented with an option to view customer information four different ways (item 2.1.1). Once the user views the customer's year-to-date transaction activity (item 2.1.1.2), the system will allow the user to back up to select a different customer or back up to the main menu (see bottom of item 2.1.1.2).

Building Prototypes and Assessing Usability Building dialogue prototypes and assessing usability are often optional activities. Some systems may be simple and straightforward. Others may be more complex but are extensions to existing systems where dialogue and display standards have already been established. In either case, you may not be required to build prototypes and do a formal assessment. However, for many other systems, it is critical that you build prototype displays and then assess the dialogue; developing a prototype can pay numerous dividends later in the systems development life cycle (e.g., it may be easier to implement a system or train users on a system they have already seen and used).

Building prototype displays is often a relatively easy activity if you use graphical development environments such as Microsoft's Visual Basic.Net. Some systems development environments include easy-to-use input and output (form, report, or window) design utilities. Also several tools called "Prototypers" or "Demo Builders" allow you to design displays quickly and show how an

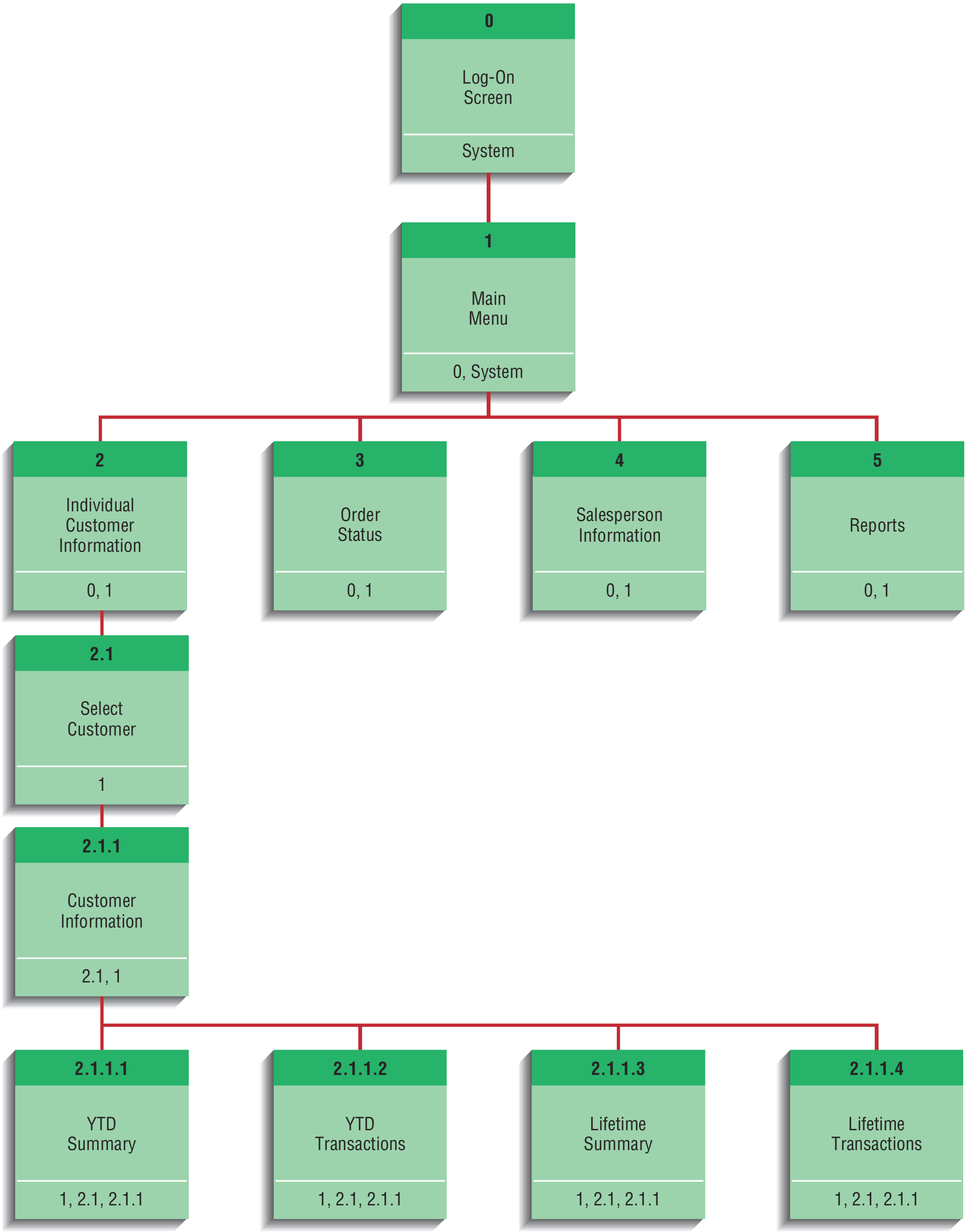


FIGURE 8-20
Dialogue diagram for the Customer Information System at Pine Valley Furniture.

interface will work within a full system. These demo systems allow users to enter data and move through displays as if they were using the actual system. Such activities are useful not only for showing how an interface will look and feel but also for assessing usability and performing user training long before actual systems are completed.

Pine Valley Furniture WebStore: Designing the Human Interface

PINE
VALLEY
FURNITURE

Designing the human interface for an Internet-based electronic commerce application is a central and critical design activity. Because customers will interact with a company at this point, much care must be put into its design. Like the process followed when designing the interface for other types of systems, a prototyping design process is most appropriate when designing the human interface for an Internet electronic commerce system. Although the techniques and technology for building the human interface for Internet sites are rapidly evolving, several general design guidelines have emerged. In this section, we examine some of these as they apply to the design of Pine Valley Furniture's WebStore.

General Guidelines for Designing Web Interfaces

Over the years, interaction standards have emerged for virtually all of the commonly used desktop computing environments such as Windows or Mac OS. However, some interface design experts believe that the growth of the Web has resulted in a big step backward for interface design. One problem is that countless nonprofessional developers are designing commercial Web applications. In addition, four other important factors contribute to a lack of standards (Johnson, 2007):

- Web's single "click-to-act" method of loading static hypertext documents (i.e., most buttons on the Web do not provide click feedback)
- Limited capabilities of most Web browsers to support finely grained user interactivity
- Limited agreed-upon standards for encoding Web content and control mechanisms
- Lack of maturity of Web scripting and programming languages as well as limitations in commonly used Web GUI component libraries

In addition to these contributing factors, designers of Web interfaces and dialogues are often guilty of many design errors. Although not inclusive of all possible errors, Table 8-15 summarizes those errors that are particularly troublesome.

General Guidelines for Web Layouts

As previously mentioned, the rapid deployment of Internet Web sites has resulted in having countless people design sites who, arguably, have limited ability to do so. To put this into perspective, consider the following quote from Web design guru, Jakob Nielsen (1999a, pp. 65–66):

> If the [Web's] growth rate does not slow down, the Web will reach 200 million sites sometime during 2003. . . . The world has about 20,000 user interface [UI] professionals. If all sites were to be professionally designed by a single UI professional, we can conclude that every UI professional in the world would need to design one Web site every working hour from now on to meet demand. This is obviously not going to happen. . . .

Continued growth in the number of unique Web sites, estimated to exceed 250 million in early 2011, makes this problem increasingly dire. Three possible solutions to the problem include the following:

- Make it possible to design reasonably usable sites without having UI expertise
- Train more people in good Web design
- Live with poorly designed sites that are hard to use

TABLE 8-15: Common Errors When Designing the Interface and Dialogues of Web Sites

Error	Description
Opening new browser window	Avoid opening a new browser window when a user clicks on a link unless it is clearly marked that a new window will be opened; users may not see that a new window has been opened, which will complicate navigation, especially when moving backward.
Breaking or slowing down the back button	Make sure users can use the back button to return to prior pages. Avoid opening new browser window; using immediate redirect where and when a user clicks the back button, they are pushed forward to an undesired location; or prevent caching such that each click of the back button requires a new trip to the server.
Complex URLs	Avoid overly long and complex URLs that make it more difficult for users to understand where they are and can cause problems if users want to e-mail page locations to others.
Orphan pages	Avoid having pages with no "parent" that can be reached by using a back button; requires users to "hack" the end of the URL to get back to a prior page.
Scrolling navigation pages	Avoid placing navigational links below where a page opens, because many users may miss these important options that are not immediately visible.
Lack of navigation support	Make sure your pages conform to user expectations by providing commonly used icon links, such as a site logo at the top of major elements. Also place these elements on pages in a consistent manner.
Hidden links	Make sure you leave a border around images that are links, don't change link colors from normal defaults, and avoid embedding links within long blocks of text.
Links that don't provide enough information	Avoid not turning off link-marking borders so that links clearly show which links users have clicked and which they have not. Make sure users know which links are internal anchor points versus external links, and indicate if a link brings up a separate browser window from those that do not. Finally, make sure link images and text provide enough information to the user so that they understand the meaning of the link.
Buttons that provide no click feedback	Avoid using image buttons that don't clearly change when being clicked; use Web GUI toolkit button, HTML from-submit buttons, or simple textual links.

Designing forms and reports may lead to errors that are specific to Web site design. It is unfortunately beyond the scope of this book to critically examine all possible design problems with contemporary Web sites. Here, we will simply summarize those errors that commonly occur and are particularly detrimental to the user's experience (see Table 8-16). Fortunately, numerous excellent sources are available for learning more about designing useful Web sites (Ash, 2008; Loveday and Niehaus, 2007; Nielsen and Loranger, 2006; Veeny, 2008; www.useit.com; www.webpagesthatsuck.com).

Designing the Human Interface at Pine Valley Furniture

The first design activity that Jim Woo and the PVF development team focused on was the human-computer interface. To begin, they reviewed many popular electronic commerce Web sites and established the following design guidelines:

- Menu-driven navigation with cookie crumbs
- Lightweight graphics
- Forms and data integrity rules
- Template-based HTML

In order to ensure that all team members understood what was meant by each guideline, Jim organized a design briefing to explain how each would be incorporated into the WebStore interface design.

TABLE 8-16: Common Errors When Designing the Layout of Web Pages

Error	Recommendation
Nonstandard use of GUI widgets	Make sure that when using standard design items, that they behave in accordance to major interface design standards. For example, the rules for radio buttons state that they are used to select one item among a set of items that is, not confirmed until "OK'ed" by a user. In many Web sites, selecting radio buttons are used as both *selection* and *action*.
Anything that looks like advertising	Because research on Web traffic has shown that many users have learned to stop paying attention to Web advertisement, make sure that you avoid designing any legitimate information in a manner that resembles advertising (e.g., banners, animations, pop-ups).
Bleeding-edge technology	Make sure that users don't need the latest browsers or plug-ins to view your site.
Scrolling text and looping animators	Avoid scrolling text and animations because they are both hard to read and often equated by users with advertising.
Nonstandard link colors	Avoid using nonstandard colors to show links and for showing links that users have already used; nonstandard colors will confuse the user and reduce ease of use.
Outdated information	Make sure that your site is continuously updated so that users "feel" that the site is regularly maintained and updated. Outdated content is a sure way to lose credibility.
Slow download times	Avoid using large images, lots of images, unnecessary animations, or other time-consuming content that will slow the downloading time of a page.
Fixed-formatted text	Avoid fixed-formatted text that requires users to scroll horizontally to view contents or links.
Displaying long lists as long pages	Avoid requiring users to scroll down a page to view information, especially navigational controls. Manage information by showing only *N* items at a time, using multiple pages, or by using a scrolling container within the window.

Menu-Driven Navigation with Cookie Crumbs

After reviewing several sites, the team concluded that menus should stay in the exact same place throughout the entire site. They concluded that placing a menu in the same location on every page will help customers to become familiar with the site more quickly and therefore to navigate through the site more rapidly. Experienced Web developers know that the quicker customers can reach a specific destination at a site, the quicker they can purchase the product they are looking for or get the information they set out to find. Jim emphasized this point by stating, "These details may seem silly, but the second users find themselves 'lost' in our site, they're gone. One mouse click and they're no longer shopping at Pine Valley Furniture but at one of our competitor's sites."

A second design feature, and one that is being used on many electronic commerce sites, is cookie crumbs. **Cookie crumbs** are a technique for showing users where they are in the site by placing "tabs" on a Web page that remind users where they are and where they have been. These tabs are hypertext links that can allow users to move backward quickly in the site. For example, suppose that a site is four levels deep, with the top level called "Entrance," the second "Products," the third "Options," and the fourth "Order." As the user moves deeper into the site, a tab is displayed across the top of the page showing the user where she is and giving her the ability to jump backward quickly one or more levels. In other words, when first entering the store, a tab is displayed at the top (or some other standard place) of the screen with the word "Entrance." After moving down a level, two tabs are displayed, "Entrance" and "Products." After selecting a product on the second level, a third level is displayed where a user can choose product options. When this level is displayed, a third tab is

Cookie crumbs
A technique for showing users where they are in a Web site by placing a series of "tabs" on a Web page that shows users where they are and where they have been.

produced with the label "Options." Finally, if the customer decides to place an order and selects this option, a fourth-level screen is displayed and a fourth tab displayed with the label "Order." In summary:

Level 1: Entrance
Level 2: Entrance → Products
Level 3: Entrance → Products → Options
Level 4: Entrance → Products → Options → Order

By using cookie crumbs, users know exactly how far they have wandered from "home." If each tab is a link, users can quickly jump back to a broader part of the store should they not find exactly what they are looking for. Cookie crumbs serve two important purposes. First, they allow users to navigate to a point previously visited and will ensure that they are not lost. Second, it clearly shows users where they have been and how far they have gone from home.

Lightweight Graphics

In addition to easy menu and page navigation, the PVF development team wants a system where Web pages load quickly. A technique to assist in making pages load quickly is **lightweight graphics.** Lightweight graphics are the use of small simple images that allow a page to load as quickly as possible. "Using lightweight graphics allows pages to load quickly and helps users to reach their final location in the site—hopefully the point-of-purchase area—as quickly as possible. Large color images will only be used for displaying detailed product pictures that customers explicitly request to view," Jim explained. Experienced Web designers have found that customers are not willing to wait at each hop of navigation for a page to load, just so they have to click and wait again. The quick feedback that a Web site with lightweight graphics can provide will help to keep customers at the WebStore longer.

Lightweight graphics
The use of small simple images to allow a Web page to be displayed more quickly.

Forms and Data Integrity

Because the goal of the WebStore is to have users place orders for products, all forms that request information should be clearly labeled and provide adequate room for input. If a specific field requires a specific input format such as a date of birth or phone number, it must provide a clear example for the user so that data errors can be reduced. Additionally, the site must clearly designate which fields are optional, which are required, and which have a range of values.

Jim emphasized, "All of this to me seems a bit like overkill, but it makes processing the data much simpler. Our site checks all data before submitting it to the server for processing. This allows us to provide quicker feedback to the user on any data-entry error and eliminate the possibility of writing erroneous data into the permanent database. Additionally, we want to provide a disclaimer to reassure our customers that the data will be used only for processing orders, will never be sold to marketers, and will be kept strictly confidential."

Template-Based HTML

When Jim talked with the consultants about the WebStore during the analysis phase, they emphasized the advantages of using **template-based HTML.** He was told that when displaying individual products, it would be advantageous to try to have a few "templates" that could be used to display the entire product line. In other words, not every product needs its own page; the development time for that would be far too great. Jim explained, "We need to look for ways to write a module once and reuse it. This way, a change requires modifying one page, not seven hundred. Using HTML templates will help us create an interface that is easy to maintain. For example, a desk and a filing cabinet are two

Template-based HTML
Templates to display and process common attributes of higher-level, more abstract items.

completely different products. Yet, both have an array of finishes to choose from. Logically, each item requires the same function—namely: 'display all finishes.' If designed correctly, this function can be applied to all products in the store. On the other hand, if we write a separate module for each product, it would require us to change each and every module every time we make a product change, like adding a new finish. But a function such as 'display all finishes,' written once and associated with all appropriate products, will require the modification of one generic or 'abstract' function, not hundreds."

Key Points Review

1. **Explain the process of designing forms and reports, and the deliverables for their creation.**

   Forms and reports are created through a prototyping process. Once created, designs may be stand-alone or integrated into actual working systems. The purpose of the prototyping process, however, is to show users what a form or report will look like when the system is implemented. The outcome of this activity is the creation of a specification document where characteristics of the users, tasks, system, and environment are outlined along with each form and report design. Performance testing and usability assessments may also be included in the design specification.

2. **Apply the general guidelines for formatting forms and reports.**

   Guidelines should be followed when designing forms and reports. These guidelines, proven over years of experience with human-computer interaction, help to create professional, usable systems. Guidelines are available for the use of titles, layout of fields, navigation between pages or screens, highlighting information, format of text, and the appropriate use and layout of tables and lists.

3. **Format text, tables, and lists effectively.**

   Textual output is becoming increasingly important as text-based applications such as electronic mail, bulletin boards, and information services become more popular. Text should be displayed using common writing conventions such as mixed uppercase and lowercase, appropriate punctuation, left-justified, and a minimal amount of obscure abbreviations. Words should not be hyphenated between lines, and blocks of text should be double-spaced or, minimally, a blank line should be placed between each paragraph. Tables and lists should have meaningful labels that clearly stand out. Information should be sorted and arranged in a meaningful way. Numeric data should be right-justified.

4. **Explain the process of designing interfaces and dialogues, and the deliverables for their creation.**

   Designing interfaces and dialogues is a user-focused activity that follows a prototyping methodology of iteratively collecting information, constructing a prototype, assessing usability, and making refinements. The deliverable and outcome from interface and dialogue design is the creation of a specification that can be used to implement the design.

5. **Describe and apply the general guidelines for interface design, including guidelines for layout design, structuring data-entry-fields, providing feedback, and system help.**

   To have a usable interface, users must be able to move the cursor position, edit data, exit with different consequences, and obtain help. Numerous techniques for structuring and controlling data entry, as well as providing feedback, prompting, error messages, and a well-organized help function can be used to enhance usability.

6. **Design human-computer dialogues, including the use of dialogue diagramming.**

   Human-computer dialogues should be consistent in design, allowing for shortcuts, providing feedback and closure on tasks, handling errors, allowing for action reversal, and giving the user a sense of control and ease of navigation. Dialogue diagramming is a technique for representing human-computer dialogues. The technique uses boxes to represent screens, forms, or reports and lines to show the flow between each.

7. **Discuss interface design guidelines unique to the design of Internet-based electronic commerce systems.**

   The human-computer interface is a central and critical aspect of any Internet-based electronic commerce system. Using menu-driven navigation with cookie crumbs ensures that users can easily understand and navigate a system. Using lightweight graphics ensures that Web pages load quickly. Ensuring data integrity means that customer information is processed quickly, accurately, and securely. Using common templates ensures a consistent interface that is easy to maintain.

Key Terms Checkpoint

Here are the key terms from the chapter. The page where each term is first explained is in parentheses after the term.

1. **Audit trail (p. 253)**
2. **Cookie crumbs (p. 264)**
3. **Dialogue (p. 258)**
4. **Dialogue diagramming (p. 259)**
5. **Form (p. 234)**
6. **Lightweight graphics (p. 265)**
7. **Report (p. 234)**
8. **Template-based HTML (p. 265)**

Match each of the key terms above with the definition that best fits it.

____ 1. Templates to display and process common attributes of higher-level, more abstract items.

____ 2. A formal method for designing and representing human-computer dialogues using box and line diagrams.

____ 3. A business document that contains only predefined data; it is a passive document used only for reading or viewing; typically contains data from many unrelated records or transactions.

____ 4. A technique for showing users where they are in a Web site by placing a series of "tabs" on a Web page that shows users where they are and where they have been.

____ 5. The sequence of interaction between a user and a system.

____ 6. A business document that contains some predefined data and may include some areas where additional data are to be filled in; typically based on one database record.

____ 7. A record of the sequence of data entries and the date of those entries.

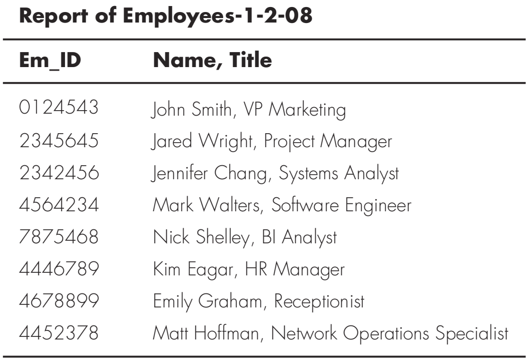
____ 8. The use of small simple images to allow a Web page to be displayed more quickly.

Review Questions

1. Describe the prototyping process of designing forms and reports. What deliverables are produced from this process? Are these deliverables the same for all types of system projects? Why or why not?
2. To which initial questions must the analyst gain answers in order to build an initial prototype of a system output?
3. How should textual information be formatted on a help screen?
4. What type of labeling can you use in a table or list to improve its usability?
5. What column, row, and text formatting issues are important when designing tables and lists?
6. Describe how numeric, textual, and alphanumeric data should be formatted in a table or list.
7. Provide some examples where variations in user, task, system, and environmental characteristics might impact the design of system forms and reports.
8. Describe the process of designing interfaces and dialogues. What deliverables are produced from this process? Are these deliverables the same for all types of system projects? Why or why not?
9. List and describe the functional capabilities needed in an interface for effective entry and navigation. Which capabilities are most important? Why? Will this be the same for all systems? Why or why not?
10. Describe the general guidelines for structuring data-entry fields. Can you think of any instances when it would be appropriate to violate these guidelines?
11. Describe four types of data errors.
12. Describe the types of system feedback. Is any form of feedback more important than the others? Why or why not?
13. Describe the general guidelines for designing usable help. Can you think of any instances when it would be appropriate to violate these guidelines?
14. What steps do you need to follow when designing a dialogue? Of the guidelines for designing a dialogue, which is most important? Why?
15. Describe what is meant by a cookie crumb. How do these help prevent users from getting lost?
16. Describe why you might want to use lightweight graphics on some Web pages and large detailed graphics on others.
17. Why is it especially important to eliminate data-entry errors on an electronic commerce Web site?
18. How can template-based HTML help to make a large electronic commerce site more maintainable?

Problems and Exercises

1. Imagine that you are to design a budget report for a colleague at work using a spreadsheet package. Following the prototyping discussed in the chapter (see also Figure 1-12), describe the steps you would take to design a prototype of this report.
2. Consider a system that produces inventory reports at a local retailer. Alternatively, consider a system that produces student academic records for the records office at a university. For whichever system you choose, answer the following design questions: Who will use the output? What is the purpose of the output? When is the output needed, and when is the information that will be used within the output available? Where does the output need to be delivered? How many people need to view the output?
3. Imagine the worst possible reports from a system. What is wrong with them? List as many problems as you can. What are the consequences of such reports? What could go wrong as a result? How does the prototyping process help guard against each problem?
4. Given the guidelines presented in this chapter, identify flaws in the design of the Report of Employees shown below. What assumptions about users and tasks did you make in order to assess this design? Redesign this report to correct these flaws.



Report of Employees-1-2-08

Em_ID	Name, Title
0124543	John Smith, VP Marketing
2345645	Jared Wright, Project Manager
2342456	Jennifer Chang, Systems Analyst
4564234	Mark Walters, Software Engineer
7875468	Nick Shelley, BI Analyst
4446789	Kim Eagar, HR Manager
4678899	Emily Graham, Receptionist
4452378	Matt Hoffman, Network Operations Specialist

5. Consider the design of a registration system for a hotel. Following design specification items in Figure 8-11, briefly describe the relevant users, tasks, and displays involved in such a system.
6. Obtain a report of some information, either from your employer (e.g., a budget or project report) or from your school (e.g., your student academic record). Evaluate the design of the report using the general guidelines in Table 8-2.
7. Design one sample data-entry screen for a hotel registration system using the data-entry guidelines provided in this chapter (see Table 8-7). Support your design with arguments for each of the design choices you made.
8. Describe some typical dialogue scenarios between users and a hotel registration system. For hints, reread the section in this chapter that provides sample dialogue between users and the Customer Information System at Pine Valley Furniture.
9. Represent the dialogues from the previous question through the use of dialogue diagrams.
10. Think of an online retailer you've recently used or considered using for a purchase. Why is good design of that retailer's interface important for the retailer? Visit the online retailer and evaluate the interface, highlighting several good things and several bad things.

Discussion Questions

1. Discuss the differences between a form and a report. What characteristics make a form or report good (bad) and effective (ineffective)?
2. Discuss the various ways that information can be highlighted on a computer display. Which methods are most effective? Are some methods better than others? If so, why and when?
3. What problems can occur if a system fails to provide clear feedback and error messages to users?
4. Use a search engine to find recommendations for good design of Web interfaces. How are these recommendations similar to those discussed in this chapter? How do they differ?

Case Problems

1. Pine Valley Furniture

Pine Valley Furniture's Customer Tracking System project is now ready to move into the systems design phase. You are excited because this phase involves designing the new system's forms, reports, and databases. During this morning's meeting with Jim Woo, he asked you to design several forms and reports for the new Customer Tracking System.

During the requirements determination phase, Jackie Judson requested that a customer profile be created for each customer. The customer profile is established when new customers place their first order. Customers will have the option of not completing a profile; however, to encourage customer participation, a 10 percent discount on the customer's total order will be given to each customer who completes a profile. In the beginning, existing customers will also be given the opportunity to participate in the customer profiling process. Customer profile information will be collected via a Customer Profile Form.

Gracie Breshers, a marketing executive, has requested that the Customer Tracking System generate a Products by Demographics Summary Report. This summary report should identify Pine Valley Furniture's major furniture categories, such as business furniture, living room, dining room, home office, and kitchen. Within each furniture category, she would like the total sales by region and customer age reported. She has also requested that several detailed reports be prepared; these reports will associate customer demographics with specific furniture category items.

Thi Hwang, a Pine Valley Furniture sales executive, would like to know, in a Customer Purchasing Frequency Report, how many of Pine Valley Furniture's customers are repeat customers, in terms of percentages, and how often they make purchases. Additionally, he would like to have this information categorized by customer type. For each customer type, he would like to know the frequency of the purchases. For instance, does this type of customer place an order at least once a month, at least every six months, at least once a year, or longer than one year? To be considered a repeat customer, the customer must have made two separate purchases within a two-year period.

a. What data will the Customer Profile Form need to collect? Using the guidelines presented in the chapter, design the Customer Profile Form.
b. Using the guidelines presented in the chapter, design the Products by Demographics Summary Report.
c. Using the guidelines presented in the chapter, design the Customer Purchasing Frequency Report.
d. Modify the dialogue diagram presented in Figure 8-20 to reflect the addition of the Customer Profile Form, Products by Demographics Summary Report, and the Customer Purchasing Frequency Report.

2. Hoosier Burger

As the lead analyst for the Hoosier Burger project, you have worked closely with Bob and Thelma Mellankamp. Having completed the systems analysis phase, you are now ready to begin designing the new Hoosier Burger information system. As the lead analyst on this project, you are responsible for overseeing the development of the forms, reports, and databases required by the new system. Because the inventory system is being automated and a new delivery system is being implemented, the Hoosier Burger system requires the development of several forms and reports.

Using your data-flow diagrams and entity-relationship diagrams, you begin the task of identifying all the necessary forms and reports. You readily identify the need for a Delivery Customer Order Form, a Customer Account Balance Form, a Low-in-Stock Report, and a Daily Delivery Summary Report. The Delivery Customer Order Form will capture order details for those customers placing delivery orders. Bob will use the Customer Account Balance Form to look up a customer's current account balance. The Low-in-Stock Report will be generated daily to identify all food items or supplies that are low in stock. The Daily Delivery Summary Report will summarize each day's delivery sales by menu item sold.

a. What data will the Delivery Customer Order Form need to collect? Using the design guidelines presented in the chapter, design the Delivery Customer Order Form.
b. What data will the Customer Account Balance Form need to show? Using the design guidelines presented in the chapter, design the Customer Account Balance Form.
c. Using the design guidelines presented in the chapter, design the Daily Delivery Summary Report.
d. Using the design guidelines presented in the chapter, design the Low-in-Stock Report.

3. Pet Nanny

Pet owners often have difficulty locating petsitters for their pets, boarding their pets, or just getting the pets to the veterinarian. Recognizing these needs, Gladys Murphy decided to open Pet Nanny, a business providing specialized pet-care services to busy pet owners. The company provides a multitude of services, including pet grooming, massage, day care, home care, aromatherapy, boarding, and pickup and delivery. The company has been experiencing a steady increase in demand for its services.

Initially, when the company was founded, all pet-care records were kept manually. However, Gladys recognized the need to update Pet Nanny's existing systems and hired your consulting firm to design the system changes. Your analyst team has just completed the requirements structuring phase and has selected an alternative design strategy. You are now ready to begin the systems design phase.

During the analysis phase, you determined that several forms and reports were necessary, including a Pet Enrollment Form, Pet Service Form, Pickup and Delivery Schedule Report, and Daily Boarding Report. When a customer wishes to use Pet Nanny's services for a new pet, the customer must provide basic information about the pet. For instance, the customer is asked to provide his or her name, address, phone number, the pet's name, birth date (if known), and special care instructions. When a customer requests a special service for the pet, such as grooming or a massage, a service record is created. Because the pickup and delivery service is one of the most popular services offered by Pet Nanny, Gladys wants to make sure that no pets are forgotten. Each morning a report listing the pet pickups and deliveries is created. She also needs a report listing the pets being boarded, their special needs, and their length of stay.

a. What data should the Pet Enrollment Form collect? Using the guidelines provided in the chapter, design the Pet Enrollment Form.
b. What data should the Pet Service Form collect? Using the guidelines provided in the chapter, design the Pet Service Form.
c. Using the guidelines provided in the chapter, design the Pickup and Delivery Schedule Report.
d. Using the guidelines provided in the chapter, design the Daily Boarding Report.

CASE: PETRIE'S ELECTRONICS

Designing the Human Interface

Jim Watanabe, project director for the "No Customer Escapes" customer loyalty system for Petrie's Electronics, walked into the conference room. Sally Fukuyama, from marketing, and Sanjay Agarwal, from IT, were already there. Also at the meeting was Sam Waterston, one of Petrie's key interface designers.

"Good morning," Jim said. "I'm glad everyone could be here today. I know you are all busy, but we need to make some real progress on the customer account area for 'No Customer Escapes.' We have just awarded the development of the system to XRA, and once all the documents are signed, they will be coming over to brief us on the implementation process and our role in it."

"I'm sorry," Sally said, "I don't understand. If we are licensing their system, what's left for us to do? Don't we just install the system and we're done?" Sally took a big gulp of coffee from her cup.

"I wish it was that easy," Jim said. "While it is true that we are licensing their system, there are many parts of it that we need to customize for our own particular needs. One obvious area where we need to customize is all of the human interfaces. We don't want the system to look generic to our loyal customers—we need to make it unique to Petrie's."

"And we have to integrate the XRA system with our own operations," added Sanjay. "For example, we have to integrate our existing marketing and product databases with the XRA CRM (see PE Figure 6-2). That's just one piece of all the technical work we have to do."

"We've already done some preliminary work on system functionality and the conceptual database," Jim said. "I want to start working on interface issues now. That's why Sam is here. What we want to do today is start work on how the customer account area should look and operate. And Sally, the customer loyalty site is a great opportunity for marketing. We can advertise specials and other promotions to our best customers on this site. Maybe we could use it to show offers that are only good for members of our loyalty program."

"Oh yeah," Sally replied, "that's a great idea. How would that look?"

"I have ideas," said Sam. Using a drawing program on a tablet PC, he started to draw different zones that would be part of the interface. "Here at the top we would have a simple banner that says 'Petrie's' and the name of the program."

"It's not really going to be called 'No Customer Escapes,' is it?" asked Sally.