BalloonWriter User's Guide

030-8076-A

Apple Computer, Inc.

© Apple Computer, Inc., 1991
20525 Mariani Avenue
Cupertino, CA 95014-6299
(408) 996-1010

Simultaneously published in the United States and Canada.

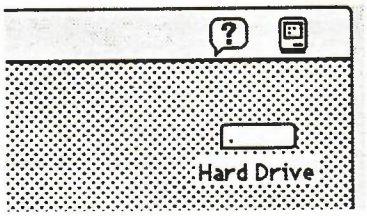
Contents

Preface: Welcome to BalloonWriter

BalloonWriter is a tool for writing Balloon Help for Macintosh application programs. BalloonWriter gives writers an easy, intuitive way to create balloons for menus, windows, and dialog boxes. BalloonWriter makes it possible to install Balloon Help in applications without any knowledge of programming.

Overview of Balloon Help

In System 7, a Help menu, represented by a question mark enclosed in a balloon, appears in the menu bar to the left of the Application menu.



The Help menu provides a standard place for users to access help in any application. It's easy to recognize, and it provides a consistent function. Users no longer need to look for help commands in the Apple menu, the File menu, or other locations.

The Help menu can contain whatever help-related items a program designer wants to include. It always includes the Show Balloons command, which turns on Balloon Help. Balloon Help acts as an extension of the user's cursor, offering information without interfering with the user's work. Balloon Help provides users with immediate information at the focus of their attention. The design of the balloon provides a visual link between the help information and the item to which it points.

Balloon Help answers these questions for the user:

- What is this?
- What does this do?
- What happens when I click this?

Balloon Help makes it easy for users to learn and use application programs, and can complement both printed documentation and other types of on-line help.

Overview of BalloonWriter

BalloonWriter allows you to write Balloon Help for an application program, and to install working balloons for many parts of an application, without doing any programming. With BalloonWriter, you can include Balloon Help as part of the documentation and support materials you create for Macintosh software products.

There are two ways to use BalloonWriter:

- You can work within the BalloonWriter environment and write a number of balloons all at once.
- You can work within the environment of your application program, selecting individual menu items, dialog box items, or windows and writing a balloon for each item you select.

While BalloonWriter allows you to write and install balloons for any item in an application program, only "standard balloons" (balloons for menus, dialog boxes, and windows) will work automatically. You will need a programmer's assistance to add Balloon Help to any element of the program that is not a menu item, a dialog item, or a window.

About this manual

This manual provides you with what you need to learn and use BalloonWriter. To get the most out of this manual, you should already know the basics of using a Macintosh, including how to use the mouse, choose menu commands, select menus, choose items in dialog boxes, and so on.

The manual consists of five chapters; three deal exclusively with BalloonWriter itself, and two discuss some of the implementation and technical issues of Balloon Help.

- Chapter 1, "Getting Started With BalloonWriter," helps you get BalloonWriter installed on your hard disk, and includes a tutorial that takes you through the basic steps of creating, installing, and testing your own help balloons.

- Chapter 2, "Using BalloonWriter," gives you more detailed information about using BalloonWriter to include Balloon Help in an application program.
- Chapter 3, "How to Write Balloons," explains how to write balloons that are easy for users to understand, useful, and consistent with the Apple style guidelines.
- Chapter 4, "Technical Information," contains reference information about how Balloon Help works, and offers some technical details about BalloonWriter.
- Chapter 5, "Troubleshooting," identifies problems you may encounter in working with BalloonWriter, explains their probable causes, and suggests solutions.

Chapter 1: Getting Started With BalloonWriter

This chapter helps you install and start using BalloonWriter. It includes the system requirements for BalloonWriter, instructions for installation, and a tutorial that guides you through the process of creating and installing balloons in an application.

System requirements

BalloonWriter runs on all Macintosh computers with at least 2 megabytes (MB) of random-access memory (4MB are recommended), a hard disk drive, and system software version 7.0 or later. This includes the Macintosh Plus and all subsequent models.

Installing BalloonWriter

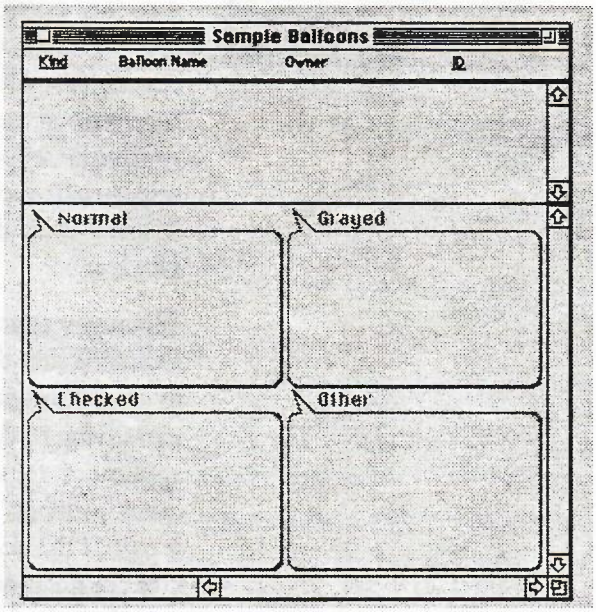
To install BalloonWriter, follow these steps:

1. Drag the file named BalloonWriter INIT to the icon of your System Folder.
2. Drag the file named BalloonWriter onto the icon of your hard disk.
3. If you plan to follow the Quick Start Tutorial, copy the version of TeachText supplied with BalloonWriter (called TeachText 7.0) onto your hard disk.
4. Restart your Macintosh.

Quick Start Tutorial

This tutorial introduces the major features of BalloonWriter and gives you the opportunity to create, install, and test help balloons in TeachText. By following the steps in this tutorial, you learn how to define, save, and install balloons, and see your balloons in action.

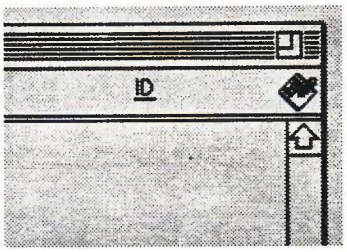
1 Open BalloonWriter by double-clicking its icon.



2 Create a new BalloonWriter document.

Choose New from the File menu. A directory dialog box appears. Name the document "Sample Balloons" and click New. A BalloonWriter document window like the one at left appears.

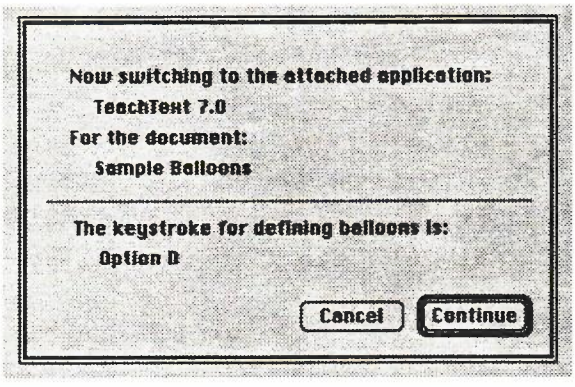
The bottom part of the document window has four areas, labeled "Normal," "Grayed," "Checked," and "Other." You use those areas to type text for the balloons you create.



3 Attach the document to an application.

Choose Attach to Application from the Balloons menu. A directory dialog box appears. Select TeachText 7.0 and click Attach. A small icon representing the attached application appears in the upper-right corner of the BalloonWriter document window.

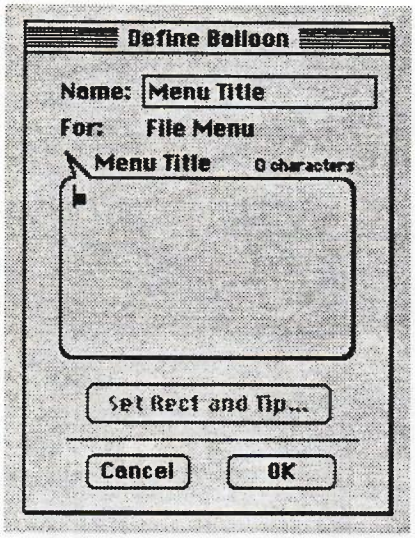
When you attach a BalloonWriter document to an application, you are identifying to BalloonWriter the application for which you will create balloons. The attachment is part of the document; if you save the document, close it, and open it again, it will remain attached to the same application.



4 Select the Define Balloons command.

Choose Define Balloons from the Balloons menu. The dialog box shown at left appears. Click Continue, and TeachText becomes the active application.

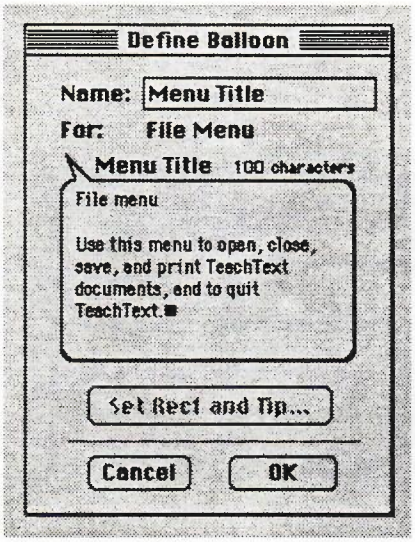
You will use the Option-D key combination to create balloons in the application.



5 Define a balloon for a menu title.

Position the pointer on the File menu title. Hold down the mouse button to pull down the menu and press Option-D while the pointer is still over the word File. Release the mouse button. The dialog box to the left appears.

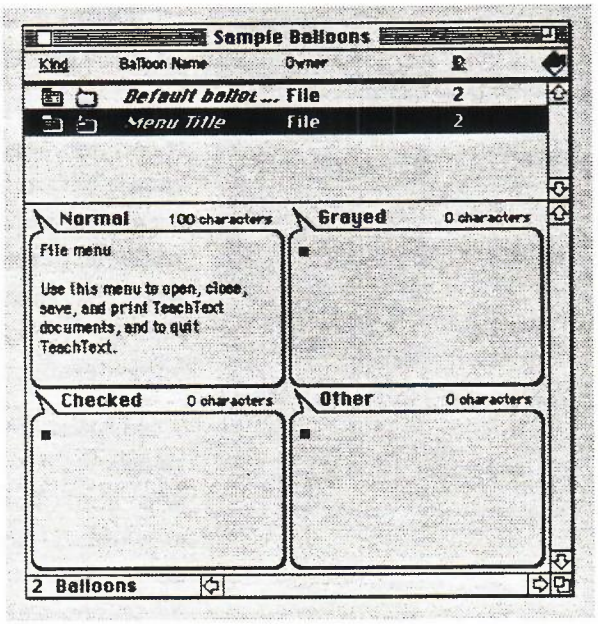
The Option-D key combination signals BalloonWriter to find out what kind of object in the application the pointer is over, and to create an empty balloon for that object.



6 Type text for the balloon.

Type the text shown at left. The small black square marks the end of the text. Just above the typing area, the number of characters you have typed is displayed.

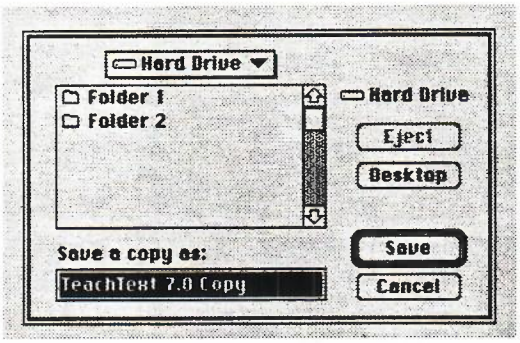
When you are finished typing, click OK to copy your text to the BalloonWriter document and close the dialog box.



7 Switch back to BalloonWriter.

Use the Application menu (at the right side of the menu bar) to switch back to BalloonWriter. Your document contains the new File menu title balloon, and the text you typed appears in the "Normal" area. The other areas can contain text for the balloon when the application object you select (in this case the File menu) appears grayed, checked, or in some other state.

BalloonWriter has also automatically created a Default balloon for the File menu. You don't need to do anything with the Default balloon for this tutorial.

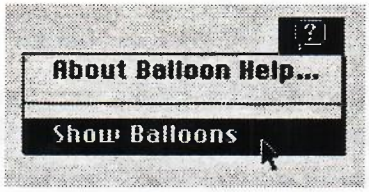


8 Install your balloon into TeachText 7.0.

Choose Install Balloons from the Balloons menu. In the dialog box that appears, click Save to save a copy of TeachText 7.0. (BalloonWriter always makes a copy of the attached application before installing balloons.)

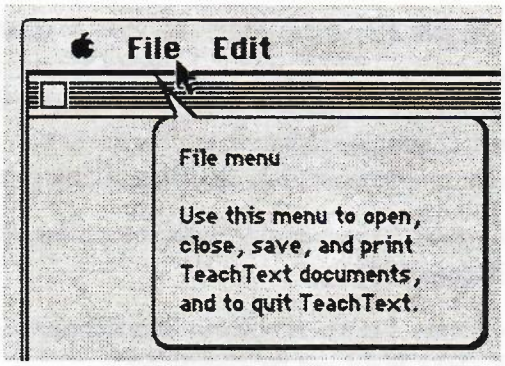
9 Open TeachText 7.0 Copy.

Use the Application menu to switch to the Finder. Double-click the TeachText 7.0 Copy icon.



10 Turn on Balloon Help.

Choose Show Balloons from the Help menu.



11 Display your new balloon.

Position the pointer on the title of the File menu. The balloon you installed appears.

Congratulations! You have reached the end of the Quick Start Tutorial, and you are now a Balloon Writer! You have learned how to:

- create a new BalloonWriter document
- attach the document to an application
- define a balloon
- install a balloon and test it

With these basic skills, you can use BalloonWriter to add Balloon Help to the menus, dialog boxes, and windows of nearly any application program.

Where to go from here

Chapter 2 describes additional ways to create balloons and ways to create custom balloons — balloons for any item other than a window, a menu, or a dialog box. Chapter 2 also explains how to use BalloonWriter's different file formats to save balloons in a form that programmers can use when building an application program.

Chapter 3 presents guidelines for writing effective, user-friendly Balloon Help.

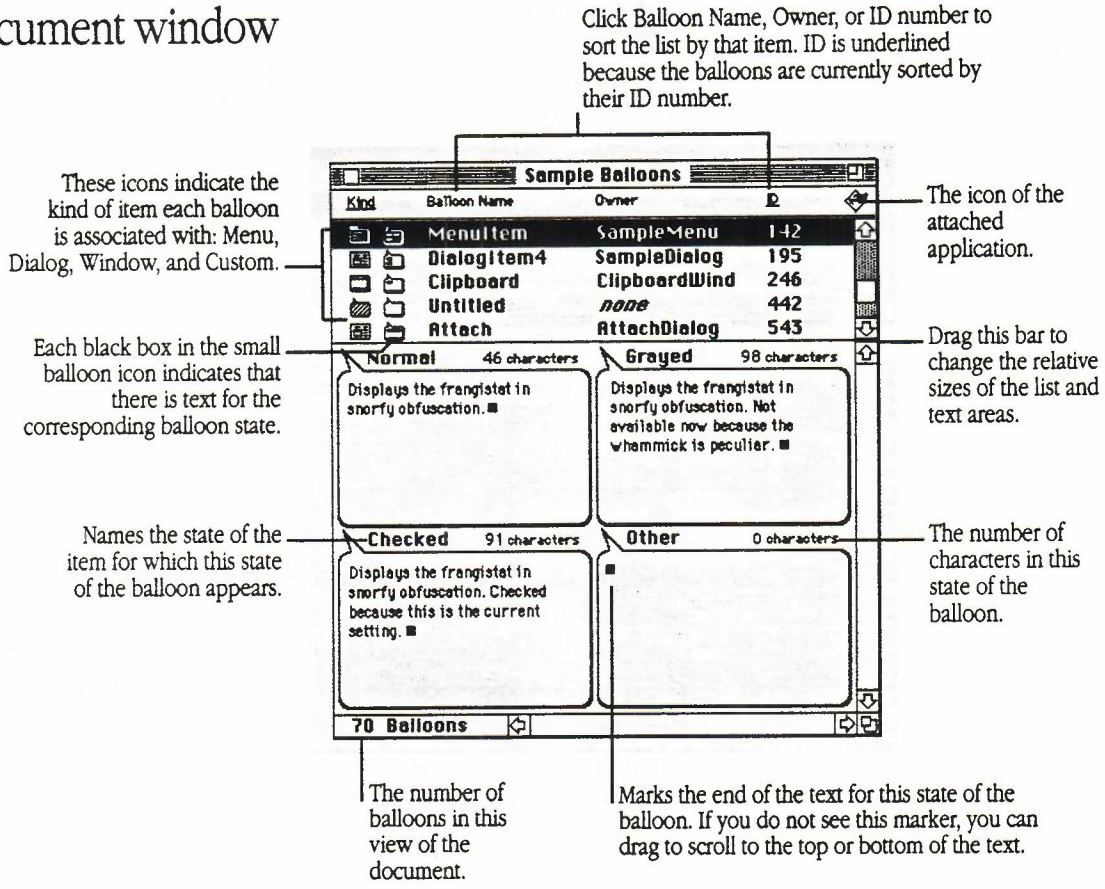
Chapter 4 is where you can find technical details about Balloon Help and BalloonWriter.

Chapter 5 describes some problems you may encounter in using BalloonWriter, and what you can do to solve them.

Chapter 2:
Using BalloonWriter

This chapter presents an overview of the BalloonWriter document window, a brief guide to terms and concepts you may find useful in using BalloonWriter, and detailed discussion of the typical tasks and operations you'll use in writing Balloon Help. Because there are many ways to approach the process of writing Balloon Help, you don't need to read those sections of this chapter in order. Instead, find the description that matches what you want to do, and follow the steps for accomplishing it.

The BalloonWriter document window



BalloonWriter terms and concepts

This is a brief introduction to the terms and concepts used in this chapter and elsewhere in this user's guide. Use this section as your foundation to get the most out of BalloonWriter and this guide.

The four states of a balloon

There are four panels in the lower, or text, area of the BalloonWriter document window. The four panels, taken together, represent the four possible states of a single balloon: normal, grayed, checked, and other. These states correspond to the possible states of an object, such as a menu item, in an application program. For example, depending on the user's actions, a menu item can be available (normal state), unavailable (grayed state), marked with a check (checked state), or marked with another symbol (other state).

The message presented in a balloon should specify the current state of an object and what that means to the user. You can use the four states of a balloon to create similar but distinct Balloon Help messages for an object that can be in different states.

Standard and custom balloons

Generally speaking, there are two types of balloons you can create using BalloonWriter: standard balloons and custom balloons. The balloons that BalloonWriter can create automatically and attach to "owners"—menus, menu items, windows, and items in dialog boxes in an application program—are standard balloons. Balloons for other objects in an application— cells, special controls, rulers; in fact, anything but a menu, window, or dialog box item—are custom balloons. You can use BalloonWriter to create custom balloons, but BalloonWriter cannot make them work. To make custom balloons work requires some special programming.

What BalloonWriter can do

With BalloonWriter, you can create all the help balloons you need for an application program, and you can even install working standard balloons. BalloonWriter cannot install working balloons in desk accessories or control panels, however.

Creating balloons for your application

BalloonWriter offers two methods for creating balloons. You can create and write balloons one at a time within the context of the application program, or you can create a batch of balloons all at once and write text for them within BalloonWriter. To create balloons from within your application program, use the Define Balloons command. To create balloons within BalloonWriter, use the Create Standard Balloons command.

Creating balloons individually

This method is recommended when your application is finished, or nearly finished. You can create menu, window, and dialog balloons with the Define Balloons command. You cannot use this method to create custom balloons.

To create balloons within your application using the Define Balloons command, follow these steps:

1. Open BalloonWriter.
2. Choose New from the File menu to open a new document.
3. Choose Attach to Application from the Balloons menu, and select your application in the dialog box that appears.
4. Choose Define Balloon from the Balloons menu (or click the icon of the attached application in the BalloonWriter document window) to switch to your application.

   You define a balloon in your application by using the Define Balloon key combination (hot key). The default key combination is Option-D. (You can change the key combination by choosing Preferences from the File menu and pressing a new key combination.)

To define a menu balloon

Point to the menu title or menu command and hold the mouse button down. Press the key combination, and then release the mouse button. Type the text for the balloon in the dialog box that appears.

To define a dialog item balloon

Display the dialog box, point to the item for which you want to define a balloon, and press the key combination. Type the balloon text in the dialog box that appears.

To define a window balloon

Point to the window area you want to write a balloon for, and press the key combination.

You can specify the area the pointer must enter for the balloon to be displayed (the rect) and the place to which the tip of the balloon points (the tip) by clicking the Set Rect and Tip button.

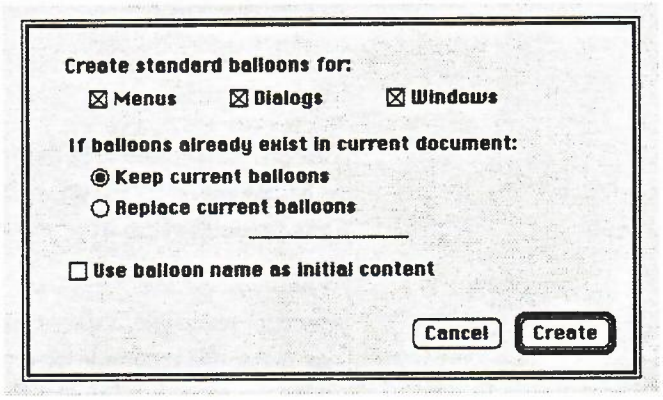
When you click the Set Rect and Tip button, an animated selection rectangle is displayed in your application's window. You can move the rectangle by dragging it from the middle. You can resize the rectangle by dragging the handles at its corners. The thick crosshair within the selection rectangle represents the location to which the balloon will point (the tip). You can drag the tip anywhere within the rectangle, but not outside it.

Creating a batch of balloons

If you prefer to create a number of balloons all at once, and write them inside BalloonWriter rather than switching to the application program for which you're writing balloons, you can use BalloonWriter to create a batch of balloons.

To create a batch of balloons for your application, follow these steps:

1. Open BalloonWriter.
2. Choose New from the File menu to open a new BalloonWriter document.
3. Choose Attach to Application from the Balloons menu, and select your application in the dialog box that appears.
4. Choose Create Standard Balloons from the Balloons menu. You use the dialog box that appears to choose whether to create menu balloons, window balloons, dialog balloons, or any combination of the three. Select the types of standard balloons you want to create.

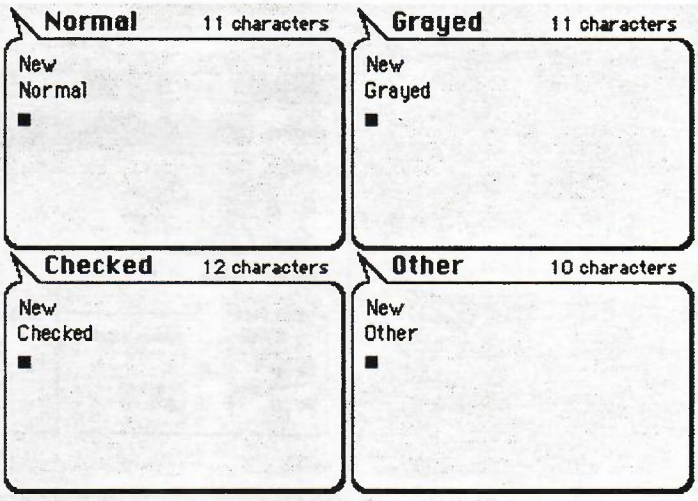


Keeping or replacing current balloons

If the active BalloonWriter document already contains some balloons, the Create Standard Balloons command may replace them with new balloons. This will occur if the current balloons are associated with the same menus, menu items, dialog items, or windows as new balloons created by the Define Balloons command. You can choose to keep the current balloons, or to replace them with the new balloons. Click the corresponding radio button in the dialog box to make your choice.

Inserting the balloon name as text

You can instruct BalloonWriter to automatically insert text into newly created standard balloons by selecting the "Use balloon name as initial content" checkbox. The text inserted identifies the owner and state of each balloon, as shown below. It can be useful as a placeholder, but should not be used as the actual content of Balloon Help.



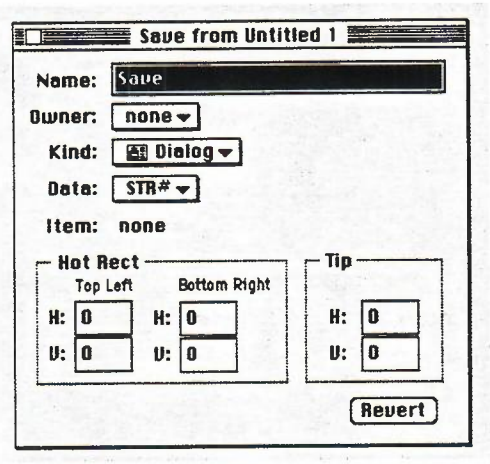
5. Click Create. BalloonWriter creates a balloon for each menu, menu item, dialog box item, and window in the application. You can write the text for those balloons without switching to the attached application.

   The Create Standard Balloons command often creates balloons you do not need. In a dialog box, for example, BalloonWriter creates balloons for separator lines and boxes. To delete an unwanted balloon, select it in the list and choose the Clear Balloon command or press the Delete key.

Changing the properties of balloons

You can change the name, owner, kind, data type, and, if appropriate, the item number, hot rect, and tip of a balloon with the Get Balloon Info command.

To change the properties of a balloon, select the balloon in the list and choose Get Balloon Info from the Balloons menu. A window similar to the one below appears. You can use the pop-up menus and text areas to specify new properties for the balloon. The changes you make are saved when you close the window.



The information displayed in the Get Balloon Info window depends on the kind of balloon you select. The window shown above displays information about a dialog balloon.

Changing default and menu title balloons

The names of default balloons and menu title balloons are italicized in the list. You can change the properties of these balloons, but you cannot change these balloons entirely. A default balloon, for example, remains a default balloon even if you change its owner. Similarly, a menu title balloon cannot be changed to a menu item balloon.

Reusing balloons

If you need to write Balloon Help for more than one application, you may find that some Balloon Help text can be shared among the applications. BalloonWriter provides a number of ways for you to reuse Balloon Help text.

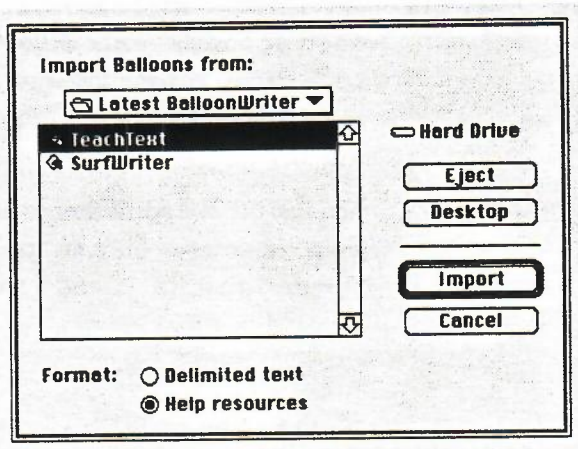
Importing balloons

You can use BalloonWriter to import balloons, complete with text, from two sources: from an application or file in which balloons have already been installed, and from a specially delimited text file.

Importing Help resources from an application or other file

If you have an application or other file in which balloons have already been installed, follow these steps to import the balloons into a BalloonWriter document:

1. Open BalloonWriter.
2. Choose Import Balloons from the File menu.
3. Click the "Help resources" radio button.
4. Select the application or file containing the balloons and click Import.
5. A dialog box appears in which you can name the new document that will contain the imported balloons. Type a name and click New. BalloonWriter reads the information, converts it to BalloonWriter format, and places it in the new document.



Importing text from a text file

If you have a text file containing balloon text in the format BalloonWriter can use (see Chapter 4 for a description of the format), follow these steps to copy the balloons into a BalloonWriter document:

1. Open BalloonWriter.
2. Choose Import Balloons from the File menu.
3. Click the "Delimited text" radio button.
4. Select the text file and click the Import button.
5. A dialog box appears in which you can name the new document that will contain the imported balloons. Type a name and click New. BalloonWriter reads the information, converts it to BalloonWriter format, and places it in the new document.

Copying and pasting balloons

You can copy and paste both text and entire balloons. If you open two BalloonWriter documents at once, balloons (and groups of balloons) can be copied and pasted from one document to the other. To copy balloons, select one or more in the list and choose Copy Balloons from the Edit menu. To paste balloons, choose Paste Balloons from the Edit menu.

Installing duplicate window and dialog balloons

When copying and pasting balloons, it is possible to end up with "duplicate balloons"—more than one balloon for the same menu item, dialog item, or window. BalloonWriter allows you to have any number of duplicate balloons in a document. However, when you install duplicate window or dialog balloons in an application, only the last balloon of a set of duplicate balloons will appear. Installing duplicate window or dialog balloons has no appreciable effect on your application.

▲ **Warning:** Do not install duplicate default balloons for dialog boxes. When duplicate default balloons for a dialog box are installed, the balloons for that dialog box may be displayed incorrectly or not at all. ▲

Installing duplicate menu balloons

In some menus, a given position on the menu may not always display the same command. For example, the first item on BalloonWriter's Edit menu can be "Undo," "Undo Cut Text," or any of several other versions of Undo. Each of these commands is a different menu item.Because each appears in the same position on the menu, however, they share the same menu item number. You can provide balloons for menu items sharing the same number by installing balloons that also share an item number.

Installing duplicate menu item balloons, however, can cause

balloons to be displayed more slowly. When two or more menu balloons share the same item number, the Help Manager must find the right balloon by name instead of number. Comparing names is considerably slower than finding an item number.

❖ *Technical Note:* When BalloonWriter installs duplicate menu balloons, it includes additional information (called `HMCompareItems`) in the help resources. For more information about the format of help resources, see the Help Manager chapter of *Inside Macintosh*, Volume VI. ❖

The Help Manager identifies balloons by name instead of number whenever you install more than one balloon for a given menu item, even if that item does not change. To minimize the reduction in performance this causes, you should install duplicate menu balloons only when necessary to support menu items that change.

You should always keep groups of duplicate menu balloons together in the BalloonWriter document. If you install duplicate menu balloons that are not in sequence in the BalloonWriter document, the balloons may not work correctly.

▲ **Warning:** Do not install duplicate menu title balloons or duplicate default balloons for a menu. When more than one balloon is installed for a menu title, or a menu has more than one default balloon, the balloons for that menu will be displayed incorrectly or not at all. ▲

Creating custom balloons

Custom balloons are balloons without owners—that is, they are not attached to menu items, dialog box items, or windows. Custom balloons can be displayed only if the application program calls the Help Manager directly. In some applications, the majority of balloons may be custom balloons. You can use BalloonWriter to create and install custom balloons, but the balloons will not work unless the application is programmed to make them work. Installing custom balloons requires exporting balloons to a *Rez* format file for use by a programmer (see the section "Creating a File for Programmers" in this chapter, and the section "Working With Your Development Team" in Chapter 3).

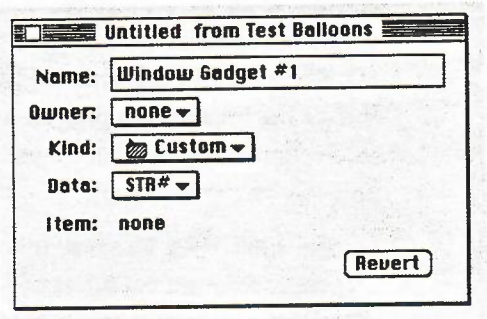
❖ *Technical Note:* Rez is a *resource compiler* often used by programmers creating Macintosh applications. It allows programmers to combine resource files from various sources into the final application. A file in Rez format is simply a file that can be used by Rez. Rez is part of Apple's *Macintosh Programmer's Workshop*, and is also provided with programming environments from some other vendors. ❖

To create custom balloons, follow these steps:

1. Compile a list of the custom balloons needed by your application. Your development team may be able to provide this list, or you can look through the application's user interface to find objects that need balloons. Each balloon needs a name, which may also be provided by your development team.
2. Open BalloonWriter and create a new document or open an existing document.
3. Choose New Custom Balloon from the Balloons menu.

   If New Custom Balloon is not available, choose All Balloons from the View menu. You cannot create a custom balloon while viewing other types of balloons.
4. Write the text for the balloon.
5. Choose Get Balloon Info from the Balloons menu.

6. Enter the balloon's name in the Get Balloon Info window, and make any other changes specified by your development team. An example of the Get Balloon Info window for a custom balloon is shown below.



7. Repeat steps 3 through 6 for each custom balloon you need.
8. Export the custom balloons into a Rez file (see the section "Creating a File for Programmers" in this chapter).
9. Deliver the file to your programming team for use in building the application program.

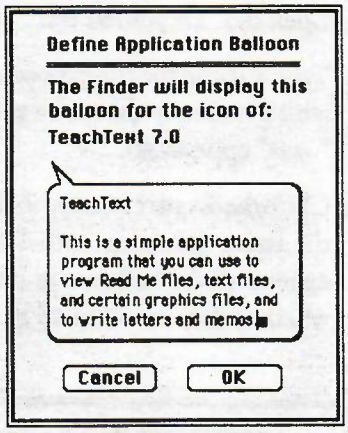
❖ *Technical Note:* For custom balloons to work, your application program must include the correct resources, and may need special software routines. For instructions and examples of how to make custom balloons work in your application, see the Help Manager chapter of *Inside Macintosh,* Volume VI. ❖

Creating an application icon balloon

You can use BalloonWriter to create a balloon linked to the icon of your application program. The application icon balloon is displayed in the Finder when the user points to the icon of your application.

To create an application icon balloon, follow these steps:

1. Open a BalloonWriter document that is attached to an application.
2. Choose Define Application Balloon from the Balloons menu.
3. In the dialog box that appears, type the text for the balloon.



4. Click OK. The application balloon is stored as part of the BalloonWriter document.

   Although the application balloon does not appear in the BalloonWriter document window, BalloonWriter includes it when you print, install, or export balloons.

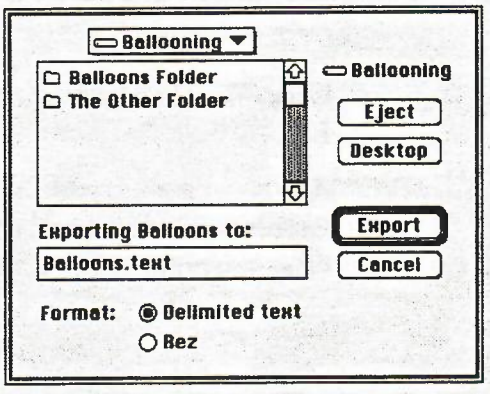
Checking spelling of balloons

BalloonWriter cannot check the spelling of balloon text. However, you can export balloon text into a text file, use other software to check and correct spelling, and import the text back into a BalloonWriter document. Follow the directions for exporting text files in the next section.

Exporting text files

To export balloon text into a text file, follow these steps:

1. Open the BalloonWriter document.
2. Select Export Balloons from the File menu.
3. Click the "Delimited text" radio button.
4. Type a name for the exported file (or use the default name, which is simply the name of the BalloonWriter document with ".text" appended).
5. Click the Export button. BalloonWriter copies the balloons in the active document into delimited text format. BalloonWriter exports all the balloons in the active document, regardless of whether they are all listed in the current view.



Importing text files

If you have a delimited text file containing balloon text in the format BalloonWriter can use, follow the instructions in the section "Importing Text From a Text File," earlier in this chapter. See Chapter 4 for a detailed description of the text format BalloonWriter can use.

Creating a file for programmers

In some cases, for example, when an application program needs custom balloons, it may be preferable for a programmer to include Balloon Help in a program as part of the programming process. The writer, in such a situation, would write Balloon Help text, but would export the text into a Rez file rather than installing the balloons directly into the application. This is the only way that custom balloons can be included in an application.

❖ *Technical Note:* Rez is a *resource compiler* often used by programmers creating Macintosh applications. It allows programmers to combine resource files from various sources into the final application. A file in Rez format is simply a file that can be used by Rez. Rez is part of Apple's *Macintosh Programmer's Workshop*, and is also provided with programming environments from some other vendors. ❖

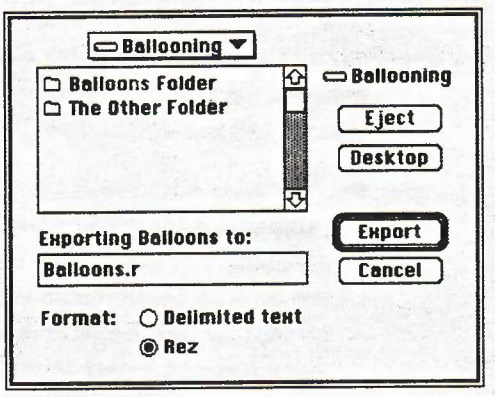
Exporting a Rez file

To export a BalloonWriter document as a Rez file, follow these steps:

1. Open the BalloonWriter document.
2. Select Export Balloons from the File menu.
3. Click the Rez radio button.
4. Type a name for the exported file (or use the default name, which is simply the name of the BalloonWriter document with ".r" appended).
5. Click the Export button. BalloonWriter copies the balloons in the active document into a Rez file. BalloonWriter exports all the

balloons in the active document, regardless of whether they are all listed in the current view.

△ **Important:** Balloons for dialog boxes may not work properly if installed through a Rez file. See the section "Making Sure Your Dialog Balloons Appear," later in this chapter, for information on making balloons for dialog boxes work when exported to a Rez file. △



Installing balloons

You can install balloons into an application program or another file by using the Install Balloons command from the Balloons menu. You can use Balloon Help in an application immediately after installing balloon.

To install balloons into an application:

1. Open a BalloonWriter document that is attached to an application and that contains balloons for that application.
2. Choose Install Balloons from the Balloons menu.
3. In the dialog box that appears, type a name for a copy of the

attached application. (BalloonWriter always makes a copy of an application before installing balloons.)

4. Click Save. The balloons are installed and ready for use.

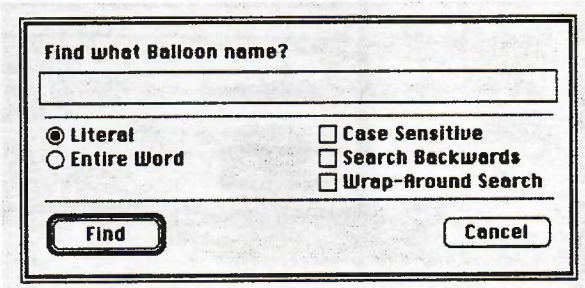
Finding balloons and balloon text

BalloonWriter allows you to search the active BalloonWriter document for balloon names and for balloon text.

Searching for a balloon name

To search for a balloon name, follow these steps:

1. Select Find Balloon Name from the Go menu.
2. In the dialog box that appears, type the whole or partial name of the balloon or balloons you need to find.



3. Click one of the radio buttons on the left to specify whether the characters you typed were a portion of a word (a literal) or the entire word.
4. You can use the checkboxes on the right to customize the search. When Case-Sensitive is selected, the search will match capital and lowercase letters exactly. Search Backwards determines whether the search should proceed upward (backward) or downward (forward) through the list of balloons. When Wrap-Around Search is selected, the whole document is searched, even

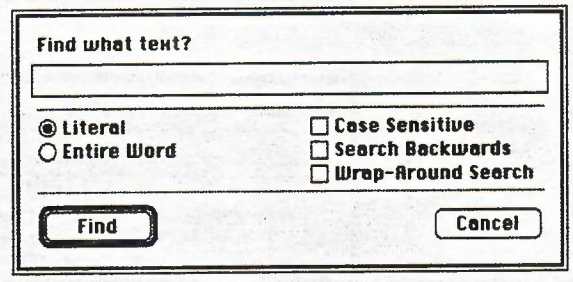
if you have not selected the first balloon in the list.

5. Click the Find button.
6. The first balloon to be found matching the characteristics you chose is selected. To continue to the next balloon matching the characteristics, choose Find Again from the Go menu.

Searching for a word or phrase

To search for a word or phrase in the text of a balloon, follow these steps:

1. Choose Find Text from the Go menu.
2. In the dialog box that appears, type part or all of the word or phrase you need to find.



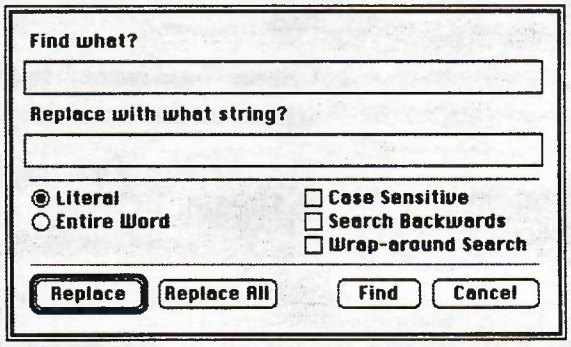
3. Click one of the radio buttons on the left to specify whether the characters you typed in the text box were a portion of a word (a literal) or the entire word.
4. You can use the checkboxes on the right to customize the search. When Case-Sensitive is selected, the search will match capital and lowercase letters exactly. Search Backwards determines whether the search should proceed upward (backward) or downward (forward) through the list of balloons. When Wrap-Around Search is selected, the whole document is searched, even if you have not selected the first balloon in the list.
5. Click the Find button.

6. The first piece of text to be found matching the characteristics you chose is displayed and highlighted. The balloon containing the text is selected. To continue to the next piece of text matching the characteristics, choose Find Again from the Go menu.

Searching for and replacing text

To search for and automatically replace one piece of text with another, follow these steps:

1. Choose Replace from the Go menu.
2. In the dialog box that appears, under "Find What?", type the characters you want to change.
3. Under "Replace with what string?", type the characters you want to use as a replacement. For example, you might want to replace all occurrences of "System 6" with "System 7".



4. Click one of the radio buttons on the left to specify whether the characters you typed in the text box were a portion of a word (a literal) or the entire word.
5. You can use the checkboxes on the right to customize the search. When Case-Sensitive is selected, the search will match capital and lowercase letters exactly. Search Backwards determines whether the search should proceed upward (backward) or

downward (forward) through the list of balloons. When Wrap-Around Search is selected, the whole document is searched, even if you have not selected the first balloon in the list.

- Click the Replace button to replace the first instance of the search string with the replacement string.
- Click the Replace All button to replace all instances of the search string with the replacement string.
- Click the Find button to find and highlight (but not replace) the first instance of the search string.

Finding selected text

There is a quicker way to find a piece of text that matches text in a balloon. You can select a piece of text in a balloon, and use the "Find Selected" command to search for matching text. To find selected text, follow these steps:

1. In a balloon, select the text (for example, "system") you want to find.
2. Choose "Find system" from the Go menu (the text you select appears on the menu along with the word "Find").

   The next piece of text matching your selection is displayed and highlighted.

Printing balloons

You can print any or all of the balloons in a BalloonWriter document. BalloonWriter prints the text of the balloons, but does not print any graphic representation of the balloons.

- To print all the balloons in a BalloonWriter document, choose Print from the File menu.
- To print some of the balloons in a BalloonWriter document, select the balloons you want to print in the list, and choose Print Selected Balloons from the File menu.

Special considerations for menu balloons

The following section addresses the unique aspects of creating Balloon Help for menus.

Creating balloons for menus that are not available

A menu that is not available can still be pulled down, but the title of the menu is dimmed, as are all the items. A menu may be unavailable for one of two reasons: because the application has made it so, or because there is a dialog box on the screen. You can use BalloonWriter to write balloons for each of these cases.

To write a balloon for a menu title or item that has been made unavailable by the application, simply use the "grayed" state of the balloon for the title or item. Each menu item has its own grayed state balloon.

To write balloons for the title and items of a menu that is unavailable because a dialog box is on the screen, use the "checked" and "other" states of the menu title balloon, as shown below. Note that in this case the "other" state applies to menu items even though this is a menu title balloon.

Menu Title Balloon States

Normal This balloon is for the title of a menu when the menu is in its normal, available state.	**Grayed** This balloon is for the title of a menu that is unavailable because the application program has made it unavailable.
Checked This balloon is for the title of a menu that is unavailable because a dialog box is on the screen.	**Other** This balloon is for the items in a menu that is unavailable because a dialog box is on the screen.

Creating balloons for menu items that change

Many applications have menus that can display, at different times, different commands in the same position. For example, the Undo command in BalloonWriter changes to reflect the action that can be undone (for example, "Undo Typing Text"). To the user, the menu item itself appears to change.

Each command appearing in the same menu position can have its own unique balloon. When a set of commands sharing the same menu position all have balloons, the balloons share an item number (listed in the Get Balloon Info window for each balloon). The names of the balloons correspond to the names of the commands. You can use either the Create Standard Balloons command or the Define Balloons command to create balloons for menu items that change.

Create Standard Balloons may, depending on how the application program is written, create a separate balloon for each command that can appear in the same menu position. If it does, you can simply write the text for each balloon.

If Create Standard Balloons does not create balloons for each possible command, it will create at least one balloon for that menu position. Since this balloon has the correct item number and owner ID, the best way to create balloons for the other possible commands in that menu position is to follow these steps:

1. Select the balloon, copy it, and paste it repeatedly until there are as many balloons for that menu item as there are commands that can be displayed there.
2. Give each of the pasted balloons a name corresponding exactly to one of the commands that can be displayed at the menu position. (To change the name of a balloon, select it, choose the Get Balloon Info command from the Balloons menu, and type the new name.) Do not change the item number of the pasted balloons. The item number represents the position on the menu at which the command appears, and it will be the same for the whole group of commands that can appear there.

△ **Important:** To ensure that your menu balloons work correctly, make sure the names of the balloons are exactly the same as the commands, and make sure that you have a balloon for each possible state of the menu. △

To use the Define Balloons command to create balloons for a set of menu commands that share the same menu position:

1. In BalloonWriter, use the Attach to Application and Define Balloons commands to prepare to define balloons. The Define Balloons command will make your application program active.
2. Perform the appropriate actions in your application program to display a particular command in the menu.
3. Highlight the menu item, press the Define Balloons key combination (hot key), and write the text for the balloon.
4. Perform the appropriate actions to display the next command in the same menu position, and define a balloon for that command, as in step 3.
5. Keep repeating step 4 until you have defined a balloon for every possible command that can be displayed in the same position in the menu.

Creating balloons for hierarchical menus

BalloonWriter treats items on hierarchical menus the same as normal menu items. If you use the Create Standard Balloons command to create menu balloons, BalloonWriter creates balloons for hierarchical menu items as well as for items in normal menus. Simply write the text for the balloons as you normally would. If you use the Define Balloons command, select an item on a hierarchical menu and use the Define Balloons key combination just as you would for a normal menu.

△ **Important:** Do not write any text for the "other" state of the menu item from which the hierarchical menu extends. The Help Manager

may interpret the appearance of the hierarchical menu as an "other" state of that menu item. If there is a balloon for the "other" state of that item, it may appear for an instant before the hierarchical menu is displayed. △

Creating balloons for dynamic menus

Dynamic menus are menus that can acquire additional items. For example, a menu might list all available fonts, or all open application windows. You cannot know for sure how many items there will be in such a menu, and you may not know the names of the items. To ensure that every item added to a dynamic menu will have a balloon, you can use the default balloon for the menu. The default balloon for a menu will appear for any item in that menu for which there is no specific balloon.

The Create Standard Balloons command automatically creates a default balloon for each menu.

The Define Balloons command automatically creates a default balloon for a menu when you define the first balloon for an item in the menu.

Creating balloons for pop-up menus

Creating balloons for pop-up menus in dialog boxes requires two types of balloons. The dialog button from which the menu pops up should have a dialog balloon, and the menu itself should have one or more menu balloons.

If you use the Create Standard Balloons command to create dialog and menu balloons, BalloonWriter creates balloons for the dialog buttons from which the pop-up menu appears, and for the items in the pop-up menu. Write the text for the balloons as you normally would.

If you use the Define Balloons command, select an item on a pop-up menu and use the Define Balloons key combination to create menu balloons. Place the pointer over the button from which the menu pops up and use the Define Balloons key combination to

create a dialog balloon. Pop-up menus do not have menu titles, and so do not need to write menu title balloons. Do not delete the menu title balloon for a popup menu, however. BalloonWriter requires a menu title balloon in order to properly install balloons, even if the menu title balloon has no text.

Special considerations for window balloons

The following section addresses the unique aspects of creating Balloon Help for windows.

Window names and balloons

Window balloons are linked to the name of the window. If your application displays one or more windows that always have the same name, you can use BalloonWriter to create and install working balloons for those windows.

To create a window balloon that will appear for a type of window regardless of the name of the window (for example, a document window named by the user), you must use custom balloons, and your application must be written with software routines that display the custom balloons at the appropriate times. See the Help Manager chapter of *Inside Macintosh,* Volume VI.

Creating balloons for items in windows

Many applications use windows to display items with which users can perform tasks. Buttons, tools, formats, and other application-specific objects can appear in windows. Some items in windows can have standard window balloons, while others cannot be detected by BalloonWriter, and therefore require custom balloons and additional programming to enable Balloon Help support.

You can use standard window balloons for items in windows only if

- The window containing the item has a name that does not change. For example, you cannot use standard window balloons for document windows named by the user.

- The item has a fixed location relative to the top left corner of the window.

The only way to provide Balloon Help for items not meeting these criteria is to create custom balloons and make sure the necessary programming is done to enable the application to control the display of the balloons.

The Create Standard Balloons command creates balloons for any named windows in an application program. You can type text for these balloons as you would for any other balloon.

To create a standard window balloon using the Define Balloons command:

1. In BalloonWriter, use the Attach to Application and Define Balloons commands to prepare to define balloons. The Define Balloons command will make your application program active.
2. In your application program, perform the appropriate actions to display the window.
3. Point at the portion of the window for which you want to define a balloon, and press the Define Balloon key combination (hot key). Write the text for the balloon.
4. If necessary, click the Set Rect and Tip button. A selection rectangle appears, representing the hot rect of the balloon (the area in which the pointer must be for the balloon to appear). Inside the selection rectangle is a crosshair, representing the position of the tip of the balloon.

   You can drag the middle of the selection rectangle to move it, and you can drag the handles at the corners to resize it. You can drag the tip anywhere within the selection rectangle.
5. Keep repeating step 3 until you have defined a balloon for every necessary portion of the window.

   ❖ *Note:* If you try to define a window balloon when your pointer is inside the hot rect of a previously defined balloon,

the text of the previous balloon will appear in the Define Balloon dialog box. This occurs even though you cannot see the hot rects of previously defined balloons. ❖

To create a custom balloon for an item appearing in a window:

1. Choose New Custom Balloon from the Balloons menu to create a custom balloon.
2. Choose Get Balloon Info from the Balloons menu, and give the balloon a name. You will use the name of the balloon with your development team to make sure the program displays the correct balloons.
3. Close the Get Balloon Info window.
4. Type the balloon text for each state of the object.

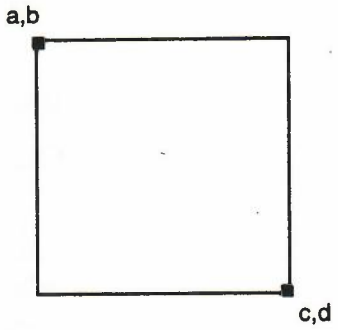
Special considerations for dialog balloons

The following section addresses the unique aspects of creating Balloon Help for dialog boxes.

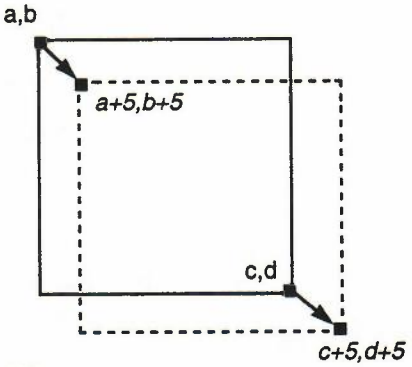
How to interpret the rect and tip coordinates

There are two rects associated with a dialog balloon. The area of the dialog item represents the rect in which the pointer must be for the balloon to appear. You cannot change this in BalloonWriter. The rect you can change in BalloonWriter is the area in which the balloon itself appears. The rect and tip coordinates shown in the Get Info window for a dialog balloon represent values that are added to the dialog item rect to create the area in which the balloon appears (the *balloon alt rect*).

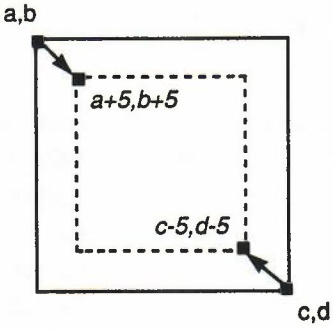
The dialog item rect a,b,c,d; a balloon alt rect with coordinates of 0,0,0,0 will be the same size and position as the dialog item rect.



The dialog item rect a,b,c,d and balloon alt rect coordinates of 5,5,5,5.



The dialog item rect a,b,c,d and balloon alt rect coordinates of 5,5,-5,-5.



As shown in this figure, if you set the rect coordinates of a dialog balloon to 0,0,0,0 the balloon alt rect will be the same as the dialog item rect. You can use positive numbers to move the corners of the balloon alt rect down and to the right. You can use negative numbers to move the corners of the balloon alt rect up and to the left.

The tip coordinates represent the location to which the tip of the balloon points. You can allow the Help Manager to determine the tip by specifying tip coordinates of 0,0, or you can choose any coordinates within the balloon alt rect for the tip. The Help Manager uses the tip coordinates you specify unless those coordinates would place the balloon off the screen.

△ **Important:** If you specify tip coordinates outside the balloon alt rect, the balloon will not appear. △

Creating a balloon for a group of dialog items

It can be useful to define a single help balloon for a group of buttons or other related items in a dialog box. BalloonWriter does not currently support this type of balloon. To create a single balloon for a group of dialog items, your development team must add software routines to the program to control the display of balloons in the dialog box. That dialog box then requires a custom balloon for the group of items.

Making sure your dialog balloons appear

Dialog balloons must be linked to their dialog boxes in order to work properly. When you use BalloonWriter to install your balloons in an application program, BalloonWriter provides the necessary links. If you export balloons to a Rez file to be compiled with your application, however, BalloonWriter is unable to link dialog balloons to their dialog boxes.

To ensure that your dialog balloons work properly when they are compiled from a Rez file, follow these steps:

1. Use BalloonWriter to install your balloons in a copy of the application program.
2. Use BalloonWriter to export your balloons to a Rez file.
3. Give your programming team the copy of the application program in which you installed the balloons, as well as the Rez file you exported.

The programmers must modify the Rez file used to create the

'DITL' resources in the application. The modified Rez file includes links between the 'DITL' resources and the corresponding balloons. To modify the Rez file properly, they should follow these steps:

1. Use the DeRez tool to obtain a Rez source code file for the 'DITL' resources in the copy of the application containing the balloons.

   The Rez code contains help items for each 'DITL' resource. The ID numbers of each help item points to the 'hdlg' resource associated with the 'DITL'.

2. Copy and paste the help item code from the Rez file produced by the DeRez tool to the Rez file used to create the 'DITL' resources.

   The help item code provides the linkage between the dialog items and the corresponding balloons. The following is a sample of Rez source code for a help item in which the ID number is 1005:

```
/* [8] */
  {0,0,0,0},
  HelpItem {
    disabled,
    HMScanhdlg {
      1005
    }
  }
```

△ **Important:** If you change the number of balloons in the BalloonWriter document, the number of dialog boxes in the application program, or the number of items in a dialog box, the ID numbers in the help item code will no longer be correct. If you make any such changes, you must repeat the above procedure to ensure that the balloons work properly. △

Items for which you may not need to write balloons

The Macintosh system software provides default balloons for several interface items. If your application program is written according to Apple guidelines, you may not need to write balloons for these items. If you do write balloons for these items, your development team must provide support to make the balloons you write take precedence over the default balloons provided by system software.

The following items have default balloons provided by system software:

Interface item	Remarks
Document icon in the Finder	You cannot install a unique balloon for a document.
Directory dialog boxes	You can install balloons for any nonstandard items in directory dialog boxes.
Window title bar, close box, and zoom box	Scroll bars and the size box do not have default balloons. You must use custom balloons for these items.
Apple menu title, Help menu title, and Application menu title	
Application icon in the Finder	You can install a unique balloon using the Define Application Balloon command.

Chapter 3
How to Write Balloons

This chapter provides guidelines for writing effective, informative Balloon Help. It also includes suggestions for writers working with programmers to create Balloon Help.

What Balloon Help can and can't do

Balloons are on-screen descriptions of items. The balloon for an item appears when the user moves the pointer to an item. This is a powerful method of providing information, because the user knows exactly what the text is referring to. But the method has some limitations. There are some kinds of information that balloons cannot display effectively.

- Balloon Help can show users what they will accomplish by using the objects on their screen, including menu commands, dialog boxes, and tool palettes.
- Balloon Help can help experienced Macintosh users who prefer to learn a program by using it, rather than by reading a manual.
- Balloon Help can't help a user who doesn't know what he or she wants to do, or a user who doesn't know where to look. (Any more than a dictionary can help someone spell a word when they don't know what letter it begins with.)
- Balloon Help can't teach your program by itself. It can't substitute for task-oriented paper or electronic documentation or training.
- Balloon Help can't teach novice Macintosh users the concepts they need to know in order to use the Macintosh.

Balloon Help works best when you keep your audience in mind as you write. Ask yourself these questions when you are planning balloons for your program:

- What people will be using your program?
- What aspects of your program are users not familiar with?
- What terminology are your users likely to know?

Unless your application program has a specialized audience, it's best to write for users who already know something about using the Macintosh (although they may not be experts) but who don't know much about your program.

Working with your development team

It takes programmers as well as writers to make the most of Balloon Help. As detailed in Chapter 2, balloons can be attached to any object or area your application program displays on the screen. You can use BalloonWriter to write all of these balloons, but you need the help of your application development team — the programmers — to make any custom balloons work.

The development team and the writer need to reach agreement on four things:

- A master list of all the balloons to be included in the application, the items the balloons will describe, and the possible states of the items (normal, grayed, checked, and other). The reasons for the states of the items may be important; you may, for example, want your balloons to explain to the user why an item is not available.
- Which balloons are standard and which are custom, requiring programming support.
- The resource types, names, and ID numbers for the custom balloons.
- The format in which the writer will provide the text for Balloon Help — for example, in a text file, a Rez file, or a BalloonWriter file.

General guidelines for writing balloons

These guidelines are presented as general rules, followed by explanations and examples. Each guideline includes one or more examples. For each example, an illustration presents an effective way to word the balloon and a table shows you the effective wording, a less effective wording, and an explanation of the difference.

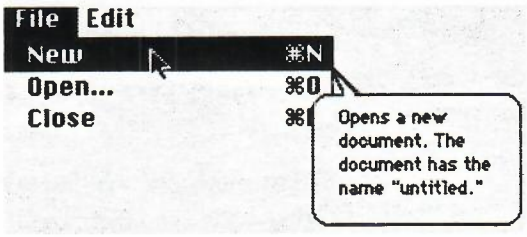
Many of the example balloons are written for fictional application programs called Widget Maker and Animals.

First describe what the user will accomplish

What the user wants most to know should come first in the balloon. The user wants to know what task he or she will accomplish by using the item.

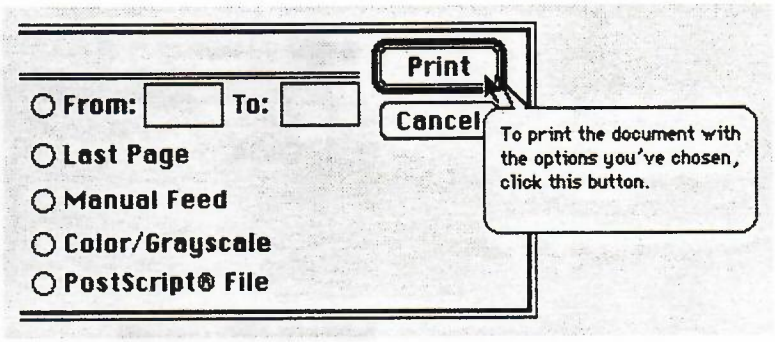
For menu commands, the best way to accomplish this is usually to begin the balloon with a verb describing what the menu command does. For other interface elements, a good construction is "To [perform action], click this [name of item]." (For more information, see the section "Wording for Specific Types of Balloons.")

After you've described what the user will accomplish by using the item the balloon points to, then, if you wish, you can describe how the item works, or other things the user might want to know about the item.



Balloon for a New command

This	Not this	Why
Opens a new document. The document has the name "untitled."	This is a menu command. Choose this command to open a new document.	The user most wants to know what he or she will accomplish by choosing the command.



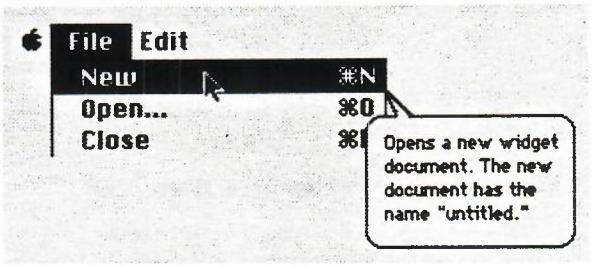
Balloon for a Print button

This	Not this	Why
To print the document with the options you've chosen, click this button.	After you've selected options, click this button to print your document.	The user most wants to know what he or she will accomplish (printing). Other things should be described later in the balloon.

Use the fewest words possible

Users are most likely to read and understand your balloons if you use the fewest words possible. Keep sentences short. Present only one concept per sentence. Present only concepts that are directly related to the item in question.

If your balloons will be translated into another language, you have another reason to keep balloons short—translated text is often 20-30 percent longer than English text. (See "Guidelines for Balloons That Will Be Translated," later in this section.)



Balloon for a New command

This	Not this	Why
Opens a new widget document. The new document has the name "untitled."	This menu command lets you open a new widget document. The new document is named "untitled."	The user already knows it's a menu command. "Lets you" doesn't tell the user anything. Everything in the program lets you do something.
	A new widget document is opened when you choose this menu command. The new document has the name "untitled."	Uses more words than necessary; uses passive voice.
	Opens a new widget document that includes standard widget-designing controls. The document has the name "untitled."	The balloon for the Open command is not an appropriate place to discuss the widget-designing controls.

Write separate balloons for checked, selected, or dimmed items

Whenever possible, write a separate balloon for each condition that a menu item, button, tool, or checkbox might be in. (This might require special programming.)

Conditions include:

- Menu items: available, dimmed, checked (in rare cases a menu item might have a diamond)

- Radio buttons or checkboxes: selected, unselected, dimmed
- Buttons in a dialog box: plain, outlined, dimmed

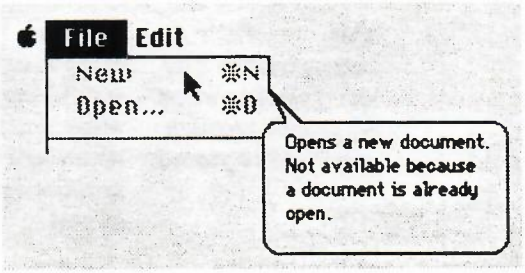
The beginning of the balloon should explain what the item usually does. The end of the balloon should describe the special condition (why it is checked or dimmed).

If there are many reasons why an item might be dimmed (or in some other special state), don't name them all. Describe one or two of the most likely reasons.

Whenever possible, use the same wording for the various balloons belonging to a single item. Don't vary the wording unnecessarily.

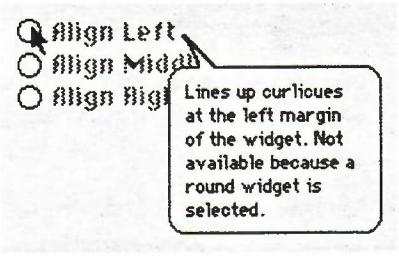
However, when describing an item that's not available, don't use the "To [perform action], click this [item]" construction, because the user can't click an item that's not available.

Use "not available" to describe a dimmed menu item or button. Use "dimmed" to describe an open icon. (Additional wording guidelines appear in the section "Guidelines for Phrasing Inside Balloons.")



Balloon for a dimmed New command

This	Not this	Why
Opens a new document. Not available because a document is already open.	Opens a new document. Grayed because a document is already open.	It is more helpful to the user to know what the grayed appearance means.
	Opens a new document. Not available because a document is already open, there is not enough memory, a dialog box is on the screen, or a desk accessory is open.	Too many explanations; might be confusing to the user. It's best to put only the most likely explanation in the balloon.

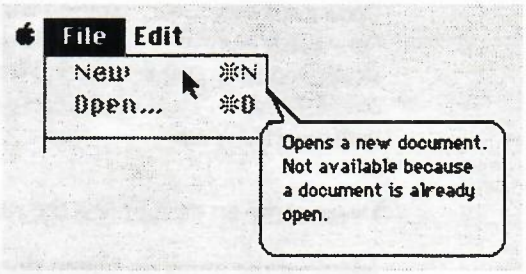


Balloon for a dimmed radio button

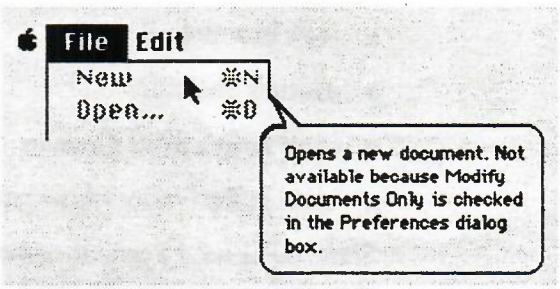
This	Not this	Why
Lines up curlicues at the left margin of the widget. Not available because a round widget is selected..	To line up curlicues, at the left margin of the widget, click this button. Not available because a round widget is selected.	The "to [perform action], click this [item]" construction isn't appropriate when an item is not available, because the user can't click the item.

Write extra balloons for hidden conditions.

The more specific your balloons are, the easier it is for the user to figure out how your application program works. It's especially important to write a specific balloon for a situation the user might find difficult to figure out. For example, if a selection in a Preferences dialog box causes some menu commands to be dimmed, a special balloon should appear when that selection is on.



Regular balloon for a dimmed command



Balloon for a command that's dimmed because of a selection elsewhere

This	Not this	Why
Regular balloon	*Single balloon for all cases*	
Opens a new document. Not available because a document is open.	Opens a new document. Not available because a document is open, a dialog box is on your screen, the program is out of memory, a desk accessory is open, or Modify Documents Only is checked in the Preferences dialog box.	The single balloon has too many explanations. The user may not read the whole list of items on the screen.
Special balloon		It's best to provide the most likely possibility in the regular balloon, and the obscure one in another balloon that appears when the option is checked.
Opens a new document. Not available because Modify Documents Only is checked in the Preferences dialog box.		

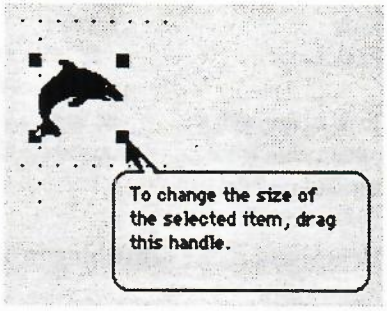
Don't name an item unless the name helps the user

Many of the items on screen don't need names. An item needs a name only if the name helps the user remember how to use the program. The following items are likely to need names:

- icons that don't already have names on the screen
- tools in a tool palette
- controls on a ruler
- controls in a paint program
- icons in the Finder whose names can be changed

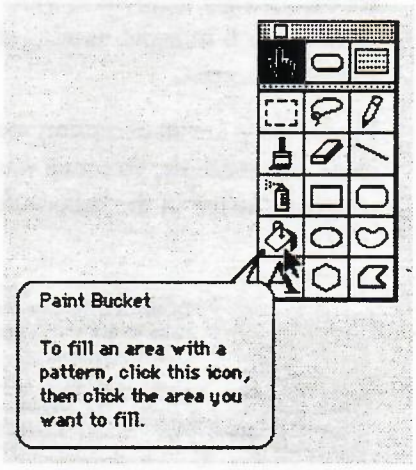
Some items need a generic name, indicating how they are used, but they don't need a specific name.

If you decide to name an item, make sure that the name you use in the balloon matches the name used in other documentation.



Balloon for a selection handle

This	Not this	Why
To change the size of the selected item, drag this handle.	Selection Handle To change the size of the selected item, drag this handle.	The user doesn't need to know the term *selection handle* in order to use the object. The generic term *handle* is sufficient.



Balloon for a paint tool

This	Not this	Why
Paint Bucket To fill an area with a pattern, click this icon and then click the area you want to fill.	To fill an area with a pattern, click this icon and then click the area you want to fill.	It's helpful to the user to know the name of the icon, because that helps him or her remember how to use the tool.

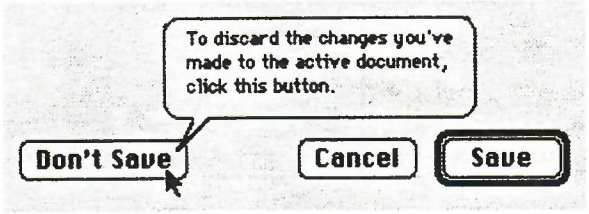
Before you name something in a balloon, ask yourself these questions:

- Will this name help the user remember how to use the program?
- Does this item need to have a specific name? Or would a generic name do?
- Is the item already named on the screen? If so, it's not necessary to name it in a balloon.

Don't name an item that's already named on screen

It's best to keep balloons as short as possible, and one way to avoid extra words is to avoid naming things in balloons if they're already named on screen.

Finder icons are an exception, because the user can rename Finder icons. For example, you may want to include your application program's name in the balloon for its icon.

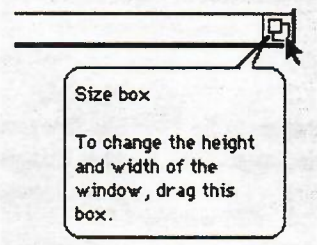


Balloon for a button in a dialog box

This	Not this	Why
To discard the changes you've made to the active document, click this button.	To discard the changes you've made to the active document, click this "Don't Save" button.	The words "Don't Save" are already on the screen, so it's not necessary to use them in the balloon.

Use active voice

Active voice usually uses fewer words and is easier to read than passive voice.



Balloon for a size box

This	Not this	Why
To change the height and width of the window, drag this box.	The window is made bigger or smaller by dragging this box.	Passive voice uses more words and is more difficult to read.

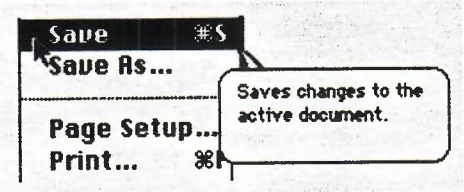
You can use a sentence fragment beginning with a verb

It's OK to use sentence fragments in a balloon—in other words, it's OK to leave out the subject of the sentence if the item is named on screen. Sentence fragments make balloons shorter, and the thing that the user really wants to know (what the item does) comes first in the balloon.

Use sentence fragments with:

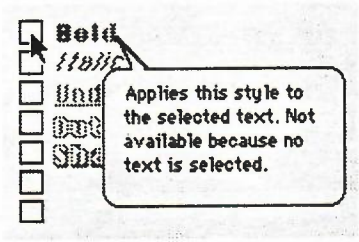
- menu commands

- checkboxes and radio buttons that are not available
- radio buttons that are selected



Balloon for a Save command

This	Not this	Why
Saves changes to the active document.	The Save command saves changes to the active document.	The user already knows it's a command, and "Save" is already on the screen.

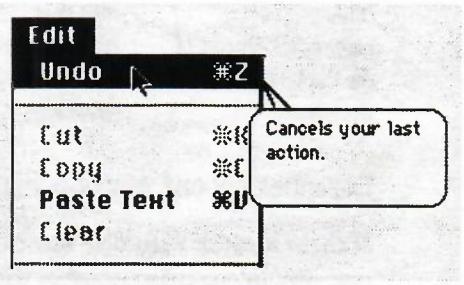


Balloon for a dimmed Bold style checkbox

This	Not this	Why
Applies this style to the selected text. Not available because no because no text is selected.	The Bold checkbox applies the Bold style to the selected text. Not available because no text is selected.	Beginning the balloon with a verb is appropriate because "Bold" is already on the screen. Also, it is not necessary to use the name "checkbox" here.
	To apply the Bold style to the selected text, click this box. Not available because no text is selected.	It's not appropriate to use the "To [perform action], click this [item]" construction if the user can't perform the action right now (because the item is not available).

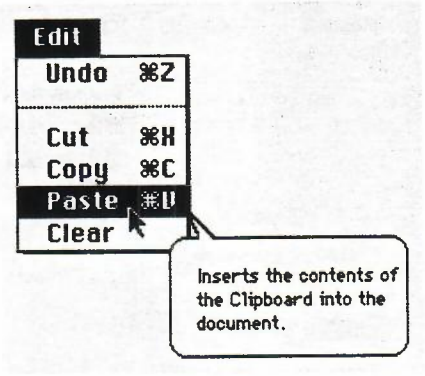
Define unfamiliar words

When you're describing a menu item or a button, try to use a word that's different from the one that appears on screen. This helps users who aren't sure what the item means.



Balloon for an Undo command

This	Not this	Why
Cancels your last action.	Undoes your last action.	It's better to use "Cancels," because that explains what the Undo command does.



Balloon for a Paste command

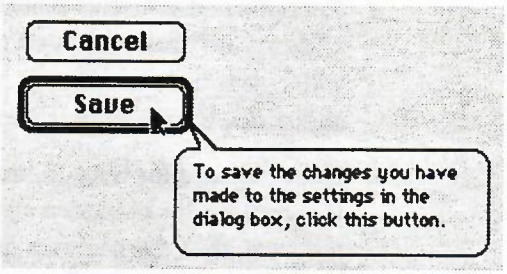
This	Not this	Why
Inserts the contents of the Clipboard into the document.	Pastes the contents of the Clipboard into the document.	It's better to use "Inserts," to explain what the Paste command does.

Describe only one way of doing something

If there is more than one way of doing something, only mention one way in the main body of the balloon.

Describe the method that involves the item the user is pointing at. In other words, if the user is pointing at a button, the balloon should instruct the user how to use the button, not how to use a keyboard shortcut for that button.

If there is more than one method involving the item the user is pointing at, describe the method that's simplest to describe and understand.



Balloon for a default Save button

This	Not this	Why
To save the changes you have made to the settings in this dialog box, click this button.	To save the changes you have made to these settings, click this button or press the Return key.	The extra information makes the sentence harder to understand.
	To save the changes you have made to these settings, press the Return key.	The balloon should describe how to use the item the user is pointing at (in this case, the button).

Go easy on hints

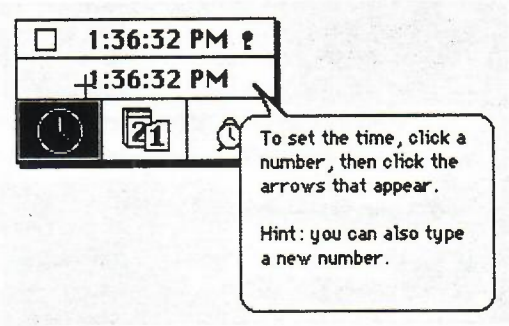
If there are just a few interesting features in your program that would be difficult to discover, then it's appropriate to use balloons to call those features to users' attention.

But if you want to give a hint or shortcut in a balloon, ask yourself these questions:

- Is the balloon reasonably short, even with the hint?
- How often will users need the information? If it's a very obscure feature that few people would need, it probably doesn't belong in the balloon.

- Are hints and shortcuts available somewhere else, for example, in a "shortcuts" dialog box or a quick-reference card? Not all users will look at balloons. If your program includes many shortcuts and tricks, be sure to list them in other documentation as well.
- Does the need for hints indicate the need for a different design? If there are a lot of hidden shortcuts and features in your program, then the program may need to be redesigned to make these features more easily accessible to users.

Put the hint or shortcut on a separate line at the end of the balloon.



Balloon for setting a clock

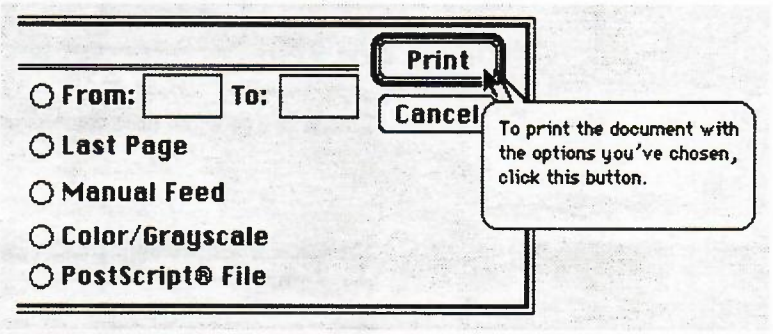
This	Not this	Why
To set the time, click a number, then click the arrows that appear. Hint: you can also type a new number.	To set the time, click a number, then click the arrows that appear or type to insert a new number.	The hint is short and simple enough to fit into the balloon, but it's easier for the user to comprehend two simple sentences than one sentence with several different options.

Describe only the item the user is pointing to

In a dialog box or a control panel, in most cases, you should try to describe only the item the balloon is pointing to. It may be

tempting to discuss the relationships among items, but if you overdo it, the balloons can become complex and difficult to read. Remember that the user can point at other items to find out what they are.

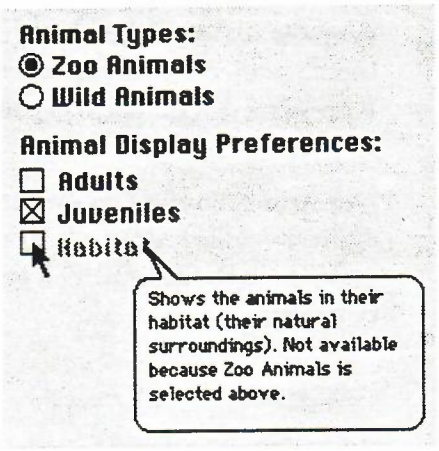
Direction indicators ("to the left," "at bottom," "below") can be difficult to follow, especially if there is more than one direction indicator in the balloon. This is another reason to avoid describing other items.



Balloon for Print button

This	Not this	Why
To print the document with the options you've chosen, click this button.	To print the number of copies of the document that you've selected to the left, using the printer named at the top of this dialog box, click this button.	The longer balloon is more difficult to read. Discussing other options is not necessary in this balloon — the options are discussed in other balloons. Multiple direction indicators ("to the left," "at the top") are difficult to understand.

It's OK to refer to other items and use direction indicators when it is necessary, for example, when the user's choices for an item are limited because of a selection elsewhere in a dialog box.

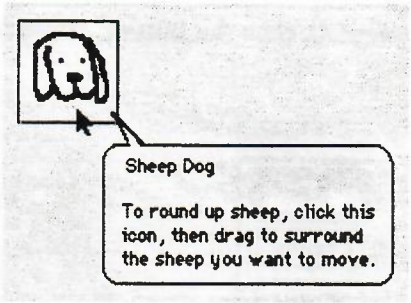


Balloon for a checkbox that's dimmed because of a selection elsewhere

Describe multiple-step procedures only if they are simple to remember

Balloons are not usually appropriate for describing multiple-step procedures, because a balloon does not stay on the screen while the user performs the various steps. The user may begin a procedure described in a balloon, and then become confused when the information disappears.

It's OK to describe a very simple two-step procedure in a balloon. For example, it's OK to describe two steps in a balloon for a tool in a tools palette.



Balloon for a tool in a palette

Guidelines for balloons that will be translated

If your balloons will be translated, don't append subphrases to a balloon when an item is in a special condition. (For example, a Finder icon balloon contains the following subphrase when an icon is open: "The icon is dimmed because the disk is open.") Write separate balloons for the special conditions instead. The grammar of a foreign language may change according to the context, so you can't assume that the same phrase will fit into several different sentences.

Expect the text to expand 20-30 percent after translation. In order to keep foreign-language balloons to 255 characters, limit the English balloons to approximately 180 characters.

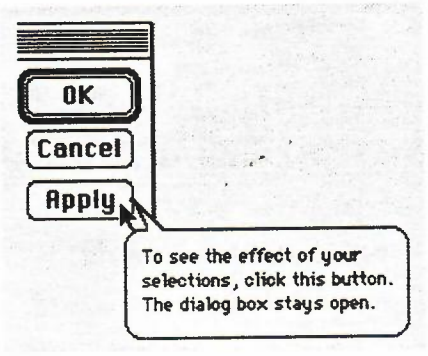
Wording for specific types of balloons

It's easiest for the user to read balloons if similar balloons use similar wording. Unnecessary variations in wording are distracting. Here are Apple's choices for phrasing in the System 7 balloons. If you use similar phrasing in your balloons, users are likely to find your balloons "familiar" and easier to understand.

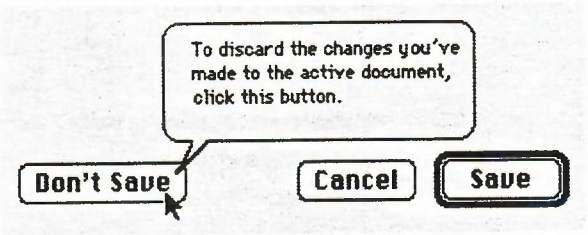
Buttons with words

For buttons that appear in dialog boxes, use the "To [perform

action], click this button" construction.



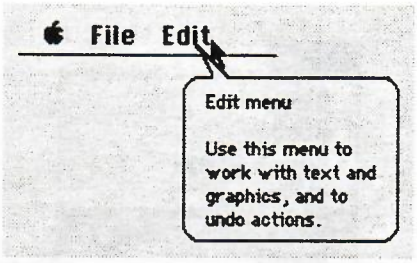
Balloon for an Apply button



Balloon for a Don't Save button

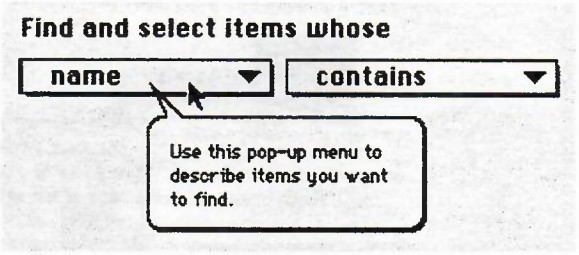
Menu titles

For menu-bar menus, give the title of the menu and then describe what kinds of commands are in the menu. You give the title of the menu because some menus on the menu bar are icons, not words.



Balloon for a menu in a menu bar

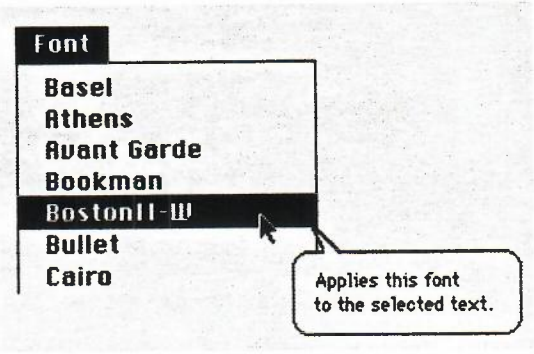
For pop-up menus, describe what to do with the menu. Don't give the menu a name.



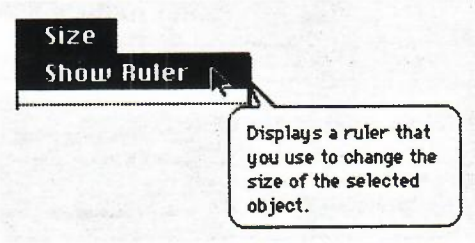
Balloon for a pop-up menu

Menu items

Don't name the menu item. Begin with a verb describing what happens when you choose the item.



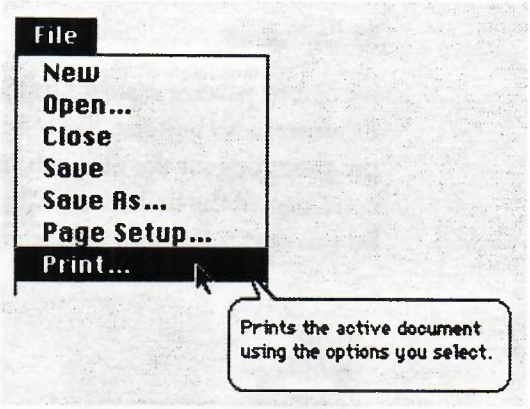
Balloon for a Font menu item



Balloon for a menu command

Menu items that display dialog boxes

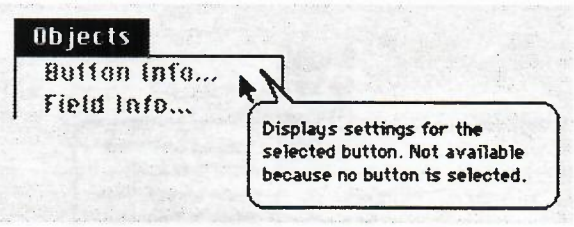
It is often not necessary to say in the balloon that a menu item displays a dialog box. The fact that a menu item displays a dialog box may not be the thing the user most wants to know. The user wants to know what choosing the menu item ultimately accomplishes.



Balloon for a menu command that displays a dialog box

Items that aren't available

After you explain what the item usually does, explain why it is dimmed or otherwise unavailable, beginning with the phrase "Not available because..."

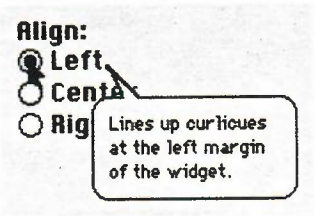


Balloon for a dimmed menu command

If there are many conditions that might result in an item's not being available, don't list them all. List one or two of the most likely only, or create separate balloons for each condition, especially obscure ones. (For more guidelines, see the sections "General Guidelines for Writing Balloons," "Radio Buttons," and "Checkboxes.")

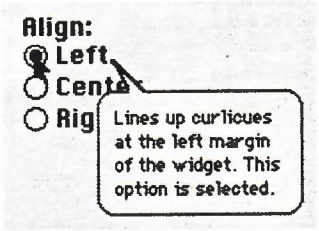
Radio buttons

It's best to provide separate balloons for selected, unselected, and dimmed radio buttons, but in some cases this might require special programming, or the program might not be able to tell the condition of the button. If that's the case, then describe what the button does when selected, beginning with a verb.



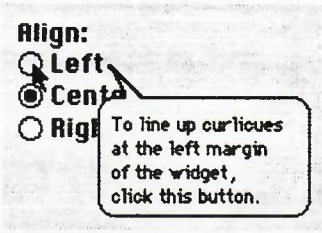
Balloon for a radio button in any state

For selected radio buttons, describe what the selected button does, beginning with a verb. At the end of the balloon, say that the button is selected.



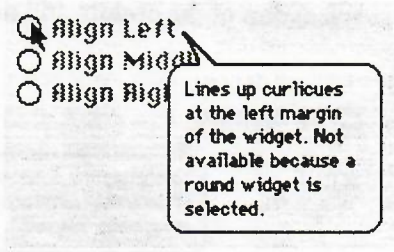
Balloon for a selected radio button

For an unselected radio button, describe what happens when you select the button.



Balloon for an unselected radio button

For a button that's not available, describe what the button does when selected, using a sentence fragment beginning with a verb. Then explain why it is not available.



Balloon for a radio button that's not available

Checkboxes

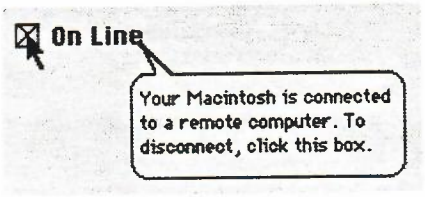
For checkboxes, provide the following information:

- The current state of the system (what the system does now, given the options that are selected or not selected)
- An explanation or description of the option provided by the checkbox (if necessary)
- How to turn the option on or off.

Do not describe the current state of the system if it's obvious or if it

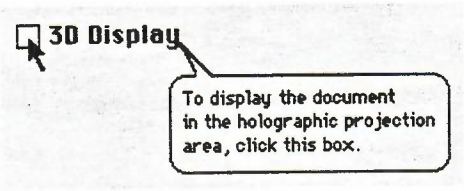
would involve saying merely "this option is not on."

Do not provide an explanation of the option if your users don't need one.



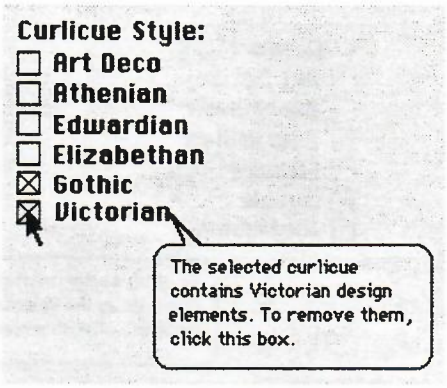
Balloon for a selected checkbox

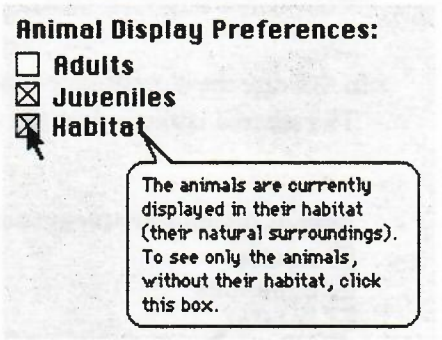
This balloon describes the current state of system, provides an explanation of the option, and describes how to turn off the option.



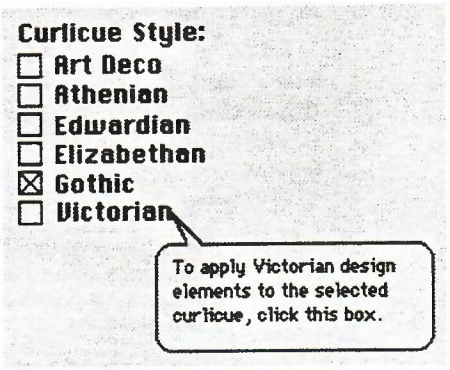
Balloon for an unselected checkbox

The balloon does not describe the current state of the system ("The document is not being displayed in the holographic projection area"), because it's obvious. It provides an explanation of the option and describes how to turn on the option.



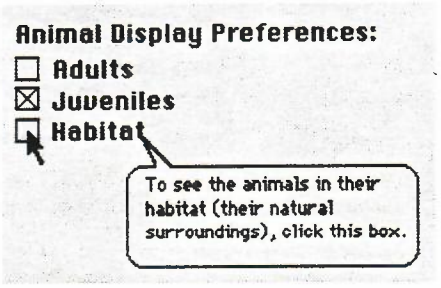


More balloons for selected checkboxes



Balloon for an unselected checkbox

In this case the current state is not described; it's not necessary to say "The selected curlicue does not contain Victorian design elements."

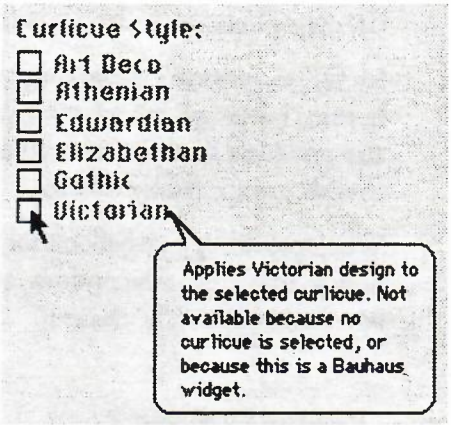


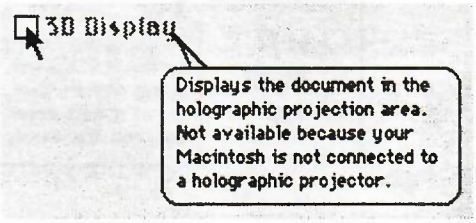
Balloon for an unselected checkbox

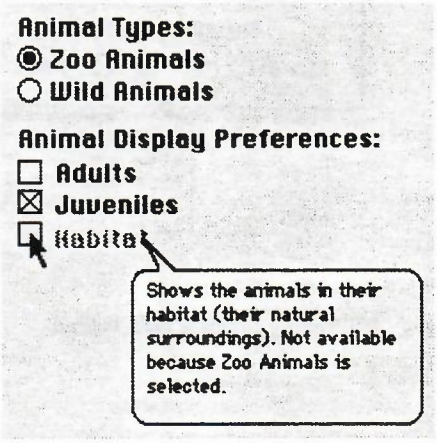
The current state ("You see the animals without their habitat") is not described, because it's obvious from the screen.

Balloons for checkboxes that aren't available

Describe what the box does when it's selected, and then explain why it is not available.


Curlicue Style:
Art Deco
Athenian
Edwardian
Elizabethan
Gothic
Victorian
Applies Victorian design to the selected curlicue. Not available because no curlicue is selected, or because this is a Bauhaus widget.

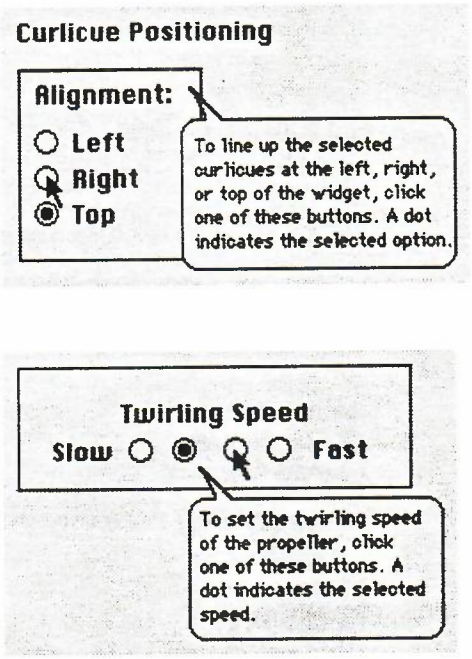

3D Display
Displays the document in the holographic projection area. Not available because your Macintosh is not connected to a holographic projector.


Animal Types:
Zoo Animals
Wild Animals
Animal Display Preferences:
Adults
Juveniles
Habitat
Shows the animals in their habitat (their natural surroundings). Not available because Zoo Animals is selected.

Groups of buttons or checkboxes

It's OK to provide a single balloon for an entire group of radio buttons (although this may require special programming). You can also provide a single balloon for a group of checkboxes, if the checkboxes are closely related.

When providing one balloon for a group of options, describe first how to implement the options, and then describe how you can tell whether an option is selected.



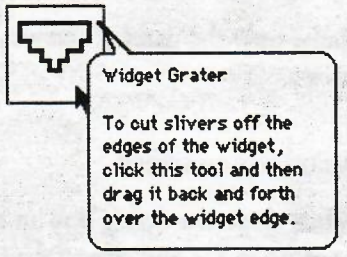
Balloons for groups of radio buttons

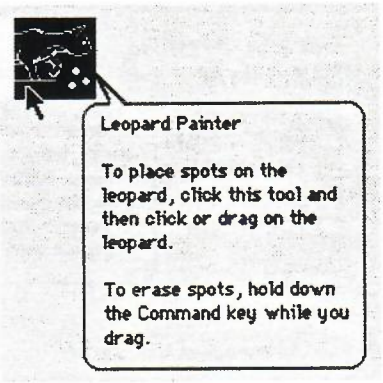
Balloon for a group of checkboxes

Tools in palettes

It's a good idea to name tools, because the name of a tool is often a clue to its use; it helps the user figure out what the tool is for.

After naming the tool, describe one or two likely ways to use it. Don't describe every shortcut or trick you can do with the modifier keys.





Balloons for design tools

Window parts

Use Apple's standard balloons for these. If your windows have non-standard parts, consult the section "General Guidelines for Writing Balloons" when writing balloons. (See Apple's *Human Interface Guidelines*, available from Addison-Wesley Publishing Company, Inc., for recommendations on window design.)

Name only the parts of the window that it's necessary for the user to know.

Dialog box on screen

When there is a dialog box on the screen, you can add the following wording to the end of each balloon that refers to an unavailable item:

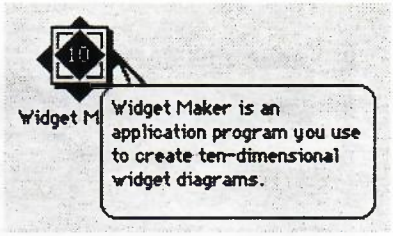
Not available because a dialog box is on the screen.

Icons

Apple provides standard balloons for icons. If you wish, you can provide your own balloons for a application program and its

associated special files (but not for your program's document icons).

Don't explain how to open icons; you can assume Macintosh users know how.

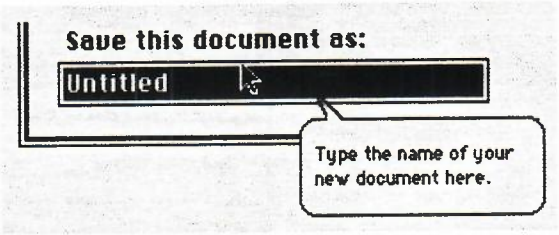


Balloon for an application program icon

Balloon for a Preferences file icon

Text entry areas

Use "here" to describe the area. You don't have to describe standard Macintosh editing procedures.



Balloon for a text box

Guidelines for phrasing inside balloons

Apple writers use the following guidelines for wording inside balloons. It's best to follow these guidelines, because consistent wording is easier to read and understand. (Just as consistent interfaces are easier to learn.)

For more guidelines, consult the *Apple Publications Style Guide*, available from APDA.

This	Not this	Why
display information about	display Info window	Avoids jargon, more meaningful to the user. It's better to describe what a feature does than to mention its name.
on the Clipboard	in the Clipboard	Preserves the metaphor. You don't put papers inside a real clipboard.
dialog box	dialog	If you must use a name for something, use its full name, not an abbreviation.

This	Not this	Why
click	click on	"On" is unnecessary.
not available	dimmed, grayed	It's more informative to the user if you describe the state of the system, rather than the appearance of the system.
active	frontmost, current	Apple style. In cases where terminology is necessary, it's best if the user sees consistent terminology in all documentation.

Word or Phrase	Guideline
this	To avoid many occurrences of the word "this" in a balloon, use "this" only when you're referring to the thing you're manipulating with the mouse (such as a button or menu item). Otherwise use "the" or "a."
close the dialog box	To avoid extra words, use this wording only in the balloon for a Cancel button. ("To close the dialog box without saving any changes, click this button"). In balloons for buttons that close dialog boxes and also perform other actions, just mention the other action ("To save the changes you made, click this button").

Chapter 4: Technical Information

This chapter provides technical information about BalloonWriter and related topics. It includes information about BalloonWriter preferences, file formats recognized by BalloonWriter, and resources. In addition, it lists sources of information about developing Macintosh application programs.

About resources

One of the ways in which the Macintosh is different from other computers is in the way it stores files. Macintosh files can consist of two parts: a data fork and a resource fork. The data fork usually contains information entered by a user, while the resource fork usually contains information needed by the computer. A file may include just one of these forks. For example, a word-processing document might contain only a data fork; there is nothing in that file except what the user has typed. An application program might contain only a resource fork.

The information in the resource fork is divided into a variety of types. The system software makes use of standard types of resources to display menus, windows, dialog boxes, icons, and other items. Each resource type is identified by a four-letter name (for example, `'MENU'`, `'DLOG'`, and so on).

Resources and Balloon Help

Menus, windows, and dialog boxes are defined in resources. The Help Manager uses these resources, as well as its own help resources, to display balloons. To provide help balloons for menus, dialog boxes, and windows, you need only create the help resources to go along with the menu, dialog box, and window resources. When the user turns on Balloon Help, the Help Manager automatically displays the help balloon you have created for each object. BalloonWriter identifies the menu, dialog box, or window resource associated with a help balloon as the "owner" of the balloon.

Application programs also display objects that are not menus, dialog boxes, or windows. These objects may be defined in unique types of resources designed by the development team, or they may not have anything to do with resources. The Help Manager cannot automatically display help balloons for these objects. An application program can, however, be designed to use the Help Manager to display help balloons for any type of object.

When a help balloon is controlled by an application program, it is not associated with (or "owned by") a menu, dialog, or window resource. These are custom balloons—you can use BalloonWriter to create them, but your development team must enable your application program to tell the Help Manager when to display them. Discussion and examples of how to include this capability in a program appear in the Help Manager chapter of *Inside Macintosh,* Volume VI.

Resource types supported by BalloonWriter

There are many different types of resources, but BalloonWriter only recognizes resources related to windows, menus, items in dialog boxes, and the resource types used for Balloon Help. The following table lists the resource types used and recognized by BalloonWriter:

Resource Type	Purpose
`'MENU'`	Used by BalloonWriter in creating standard menu balloons
`'DITL'`	Used by BalloonWriter in creating standard dialog balloons
`'WIND'`	Used by BalloonWriter in creating standard window balloons
`'hmnu'`	Help for menus, installed by BalloonWriter
`'hdlg'`	Help for dialog boxes, installed by BalloonWriter
`'hrct'`	Rectangle for dialog or window balloons, installed by BalloonWriter
`'hwin'`	Window for dialog or window balloons, installed by BalloonWriter
`'STR '`	Text for a balloon when the string is 255 characters or fewer
`'STR#'`	List of text strings for balloons
`'TEXT'`	Text for a balloon when the string is longer than 255 characters

Setting document preferences

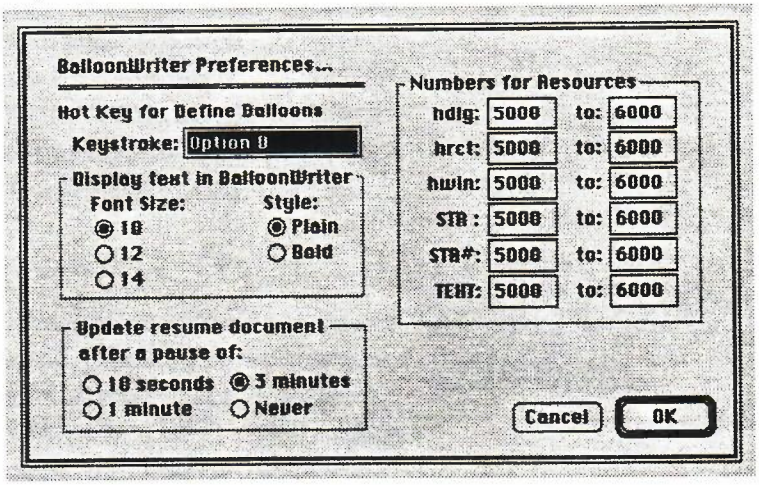
You use the Document Preferences command in the File menu to choose the key combination to use in defining balloons, and to choose the font size and style for displaying balloon text in BalloonWriter. (This setting has no effect on Balloon Help; it controls text displayed only while using BalloonWriter.) The settings in the Document Preferences dialog box apply to the current

BalloonWriter document. When you open a new document, the document preferences are always set to the defaults.

Numbers for resources

When a file contains more than one resource of the same type, each resource must be identified by a unique number. For example, if an application program has three menus, it would have three resources of type `'MENU'`, numbered 128, 129, and 130. BalloonWriter automatically assigns unique numbers to the resources it creates. You can set the range of numbers for each resource type in the Document Preferences dialog box. BalloonWriter assigns numbers consecutively within each resource type.

▲ **Warning:** Application programs may store other types of information in `'STR '`, `'STR#'`, and `'TEXT'` resources. When you install balloons, BalloonWriter checks to see if any existing resources have the same numbers you choose for the resources installed by BalloonWriter. You cannot install balloons if BalloonWriter finds existing resources with numbers matching the resources BalloonWriter installs. ▲



Resume documents

BalloonWriter automatically makes a backup copy of your current document if you haven't made any changes to it for an interval you choose. You can turn off this feature by clicking "Never" in the Preferences dialog box.

The resume document is stored in a folder in your System Folder called Resume BalloonWriter. Under normal conditions, the folder is empty because BalloonWriter deletes the latest resume document when you quit. If your system crashes, however, BalloonWriter can use the resume document to recover some or all of your work.

When you start BalloonWriter after your system crashes, a dialog box informs you of the existence of the resume document, and gives you instructions for using it.

To use a resume document, you must drag it out of the Resume BalloonWriter folder and give it a different name. Once you have done this, you can open and use it like any other BalloonWriter document.

△ **Important:** When you want to keep a resume file, make sure you rename it and drag it out of the Resume BalloonWriter folder. If you leave it in the Resume BalloonWriter folder with its original name, it is deleted when you quit BalloonWriter. △

Keyboard shortcuts

The following is a list of the keyboard combinations you can use in BalloonWriter:

Key	Function
File Menu	
⌘-N	Creates a new document.
⌘-O	Opens a document.
⌘-W	Closes the active document.
⌘-S	Saves the active document.
⌘-P	Prints balloons in the active document.
⌘-Q	Quits BalloonWriter.
Edit Menu	
⌘-Z	Undoes your last action.
⌘-X	Cuts the selection and places it on the Clipboard.
⌘-C	Copies the selection to the Clipboard.
⌘-V	Pastes the contents of the Clipboard.
⌘-A	Selects all (balloons or text).
Go Menu	
⌘-1	Goes to the first balloon in the list.
⌘-2	Goes to the next balloon up the list.
⌘-3	Goes to the next balloon down the list.
⌘-4	Goes to the last balloon in the list.
⌘-F	Finds text.
⌘-G	Repeats the previous Find command.
⌘-H	Finds selected text.
⌘-R	Finds and replaces text.
⌘-T	Repeats the previous Replace command.
Balloons Menu	
⌘-D	Defines a new balloon in the attached application.
⌘-E	Creates a new custom balloon.
⌘-I	Displays information about a balloon.

Other shortcuts

To define balloons, click the icon of the attached application in the upper-right corner of the BalloonWriter document window.

To sort balloons in the active BalloonWriter document, click Balloon Name, Owner, or ID.

Delimited text file format

The format of the delimited text files produced by the Export Balloons command is as follows: a list of command lines, blank lines, and balloon content text. No command may cross line boundaries (that is, you may have only one command per line). Each command begins with "##," followed first by a key word denoting the kind of command, then by command arguments. Balloon Help text begins with one of four commands: ##Normal, ##Grayed, ##Checked, or ##Other, and is terminated by the next command line. Here is a list of all of the kinds of commands for each balloon in a file:

`##Title`	Specifies a balloon's title
`##Normal`	The balloon's "Normal" text begins on next line
`##Grayed`	The balloon's "Grayed" text begins on next line
`##Checked`	The balloon's "Checked" text begins on next line
`##Other`	The balloon's "Other" text begins on next line
`##Owner`	Specifies the owner of the balloon (a name)
`##OwnerRes`	Specifies the balloon owner's resource type and ID
`##HelpItemRes`	A dialog balloon's help item resource type and ID
`##Item`	Menu and dialog balloons' item number
`##HotRect`	Dialog, custom, and window balloons' hotrect
`##Tip`	Dialog, custom, and window balloons' tip
`##Verified`	Reserved for future use
`##End`	Terminates specification of a balloon

The following is a list of commands at the beginning of an exported document:

`##Comment`	A comment line, ignored by Import Balloons. Used to record the name

of the exported document, and when it was exported.

##BalloonExportVersion	The version of the delimited text export
##HotKey	Hexadecimal codes for the document's hotkey and modifiers
##TextSize	Decimal value of the document's display text size
##TextStyle	Decimal value of the document's display text style
##HrctID	Range of 'hrct' resource IDs from the document's preferences dialog box
##HdlgID	Range of 'hdlg' resource IDs from the document's preferences dialog box
##STRPID	Range of 'STR#' resource IDs from the document's preferences dialog box
##HwinID	Range of 'hwin' resource IDs from the document's preferences dialog box
##TEXTID	Range of 'TEXT' resource IDs from the document's preferences dialog box
##STRID	Range of 'STR ' resource IDs from the document's preferences dialog box
##AlwaysTEXT	A nonzero value if document's preferences dialog box specifies "always TEXT;" otherwise 0
##LinkAppName	The name of the attached application
##LinkAppCreator	The attached application's 4-character signature
##LinkAppVol	Volume where the attached application is stored
##LinkAppLoc	Directory ID and creation date of the attached application
##AppIcon	Hexadecimal codes of the attached application's icon
##AppSicn	Hexadecimal codes for a 'sicn' format version of the icon
##AppInvertSicn	Hexadecimal codes for an inverted small icon
##ApplBalloon	The text of the application icon balloon
##EndDocData	Indicates the end of the section describing the document; the document's balloons follow

The following is a portion of an exported text file showing the Close and Save commands from the File menu of BalloonWriter:

```
##Title Close
##Owner menu File
```

```
##OwnerRes 129 MENU
##Item 4
##Verified 1
##Normal none -1 0
Closes the active BalloonWriter document.
##Grayed none -1 0
Closes the active BalloonWriter document. Not available now
because no BalloonWriter document is open.
##End Close
##Title Save
##Owner menu File
##OwnerRes 129 MENU
##Item 5
##Verified 1
##Normal none -1 0
Saves changes you have made to the active document.
##Grayed none -1 0
Saves changes you have made to the active document. Not
available now because no BalloonWriter document is active.
##End Save
```

For more information

If you need more information about Macintosh systems, technical products and documentation, or developer programs, several sources are available.

Technical information

Additional technical information about Macintosh systems is available in Apple technical books published by Addison-Wesley, such as *Inside Macintosh*. These are available at technical bookstores worldwide. Books and manuals published by Apple are available through APDA, the Apple Programmers and Developers Association, at the address listed in the next section. Technical notes and other materials of interest to Macintosh application developers are also available from APDA.

About APDA

APDA (Apple Programmers and Developers Association) offers worldwide access to a broad range of programming products, resources, and information for anyone developing on Apple platforms. You'll find the most current versions of Apple and third-party development tools, debuggers, compilers, languages, and technical references for all Apple platforms. To establish an APDA account, obtain additional ordering information, or find out about site licensing and developer training programs, please contact

APDA
Apple Computer, Inc.
20525 Mariani Avenue, M/S 33-G
Cupertino, CA 95014-6299

1-800-282-2732 (United States)
1-800-637-0029 (Canada)
1-408-562-3910 (International)
Fax: 1-408-562-3971
Telex: 171576
AppleLink address: APDA

If you provide commercial products and services, please call 1-408-974-4897 for information on the developer support programs available from Apple.

About Apple Developer Programs

If you plan to develop hardware or software products for sale through retail channels, you may be able to get valuable support from Apple Developer Programs. Write to them at the following address:

Apple Developer Programs
Apple Computer, Inc.
20525 Mariani Avenue
Cupertino, CA 95014-6299

Chapter 5: Troubleshooting

This chapter lists some problems you may encounter in working with BalloonWriter, offers explanations of the probable causes of the problems, and suggests solutions.

Problems, causes, and solutions

When you're creating Balloon Help, you're participating in the software development process. As always in this process, it's a good idea to save your documents often and make backup copies regularly. If you encounter a problem in working with BalloonWriter, this chapter may help you identify the cause and provide a way to solve or minimize your difficulties.

The balloons for a menu do not work

Situation: You have installed balloons into an application, and while most of them work, no balloons appear for the title or items of one menu.

Possible cause: You may have installed more than one menu title balloon or default balloon for that menu.

Solution: Make sure the BalloonWriter document contains only one menu title balloon and one default balloon per menu. Reinstall the balloons.

Dialog balloons do not work

Situation: You have created balloons for an application and exported them for the programming team. Although the menu and window balloons work, no balloons appear for dialog boxes.

Possible cause: The dialog balloons may not be linked to their dialog items.

Solution: Dialog balloons must be linked to their dialog boxes in order to work. A procedure for accomplishing this is detailed in the section "Making Sure Your Dialog Balloons Appear," in Chapter 2.

Balloon text is not imported properly

Situation: You are trying to import balloons from an application, and some of the balloon text is garbled.

Possible cause: BalloonWriter may be close to its memory limit. This is particularly likely if you are trying to import several hundred balloons.

Solution: Increase BalloonWriter's memory allocation by following these steps:

1. Quit BalloonWriter.
2. In the Finder, select the BalloonWriter icon.
3. Choose Get Info from the File menu.
4. In the lower right corner of the window that appears, type a higher number for BalloonWriter's current memory size. For example, BalloonWriter's suggested memory size is 512K, but for larger documents you may need to specify 1000K.
5. Close the Get Info window.
6. Open BalloonWriter and try importing balloons again from the same application.

   If the same problem occurs, try increasing BalloonWriter's memory allocation further.

The Define Balloons command does not work

Situation: The Define Balloons command does not work.

Possible cause: The BalloonWriter INIT has not been installed in the Extensions folder inside your System Folder.

Solution: Insert your original BalloonWriter disk into the computer's floppy disk drive and copy the file called BalloonWriter INIT into the Extensions folder inside your System Folder. Restart your Macintosh to enable the BalloonWriter INIT file to take effect.

A balloon flashes before a hierarchical menu appears

Situation: You have installed balloons into your application, and a balloon flashes for an instant just before a hierarchical menu appears.

Possible cause: You may have created an "other" balloon for the menu item from which a hierarchical menu appears. The Help Manager may interpret the appearance of the hierarchical menu as an "other" state of the menu item, and display the balloon for the "other" state of the menu item.

Solution: Remove all text from the "other" state of the balloon for the menu item and reinstall the balloons.

The hot rect does not appear

Situation: When your are defining a balloon and attempting to set the hot rect and tip, the selection rectangle does not appear.

Possible cause: The selection rectangle may be hidden beneath the Define Balloon dialog box.

Solution: Drag the Define Balloon dialog box to another part of the screen.

Attached application icon is unclear

Situation: The icon for an attached application appears inaccurate or "chunky."

Possible cause: BalloonWriter tries to use the attached application's special "small" icon resource, but not all applications contain that resource. If an application does not contain a small icon, BalloonWriter reduces the size of the regular icon to fit.

Solution: If possible, ensure that your application includes a small icon.

You can't see all the text for a balloon you're writing

Situation: The text you are writing in BalloonWriter is too long to fit in the area allowed for a single balloon state, or in the Define Balloon dialog box, and you can't see all of it.

Solution: You can scroll the text in BalloonWriter text areas by dragging to select text above or below the borders of a text area.

You can't open or use BalloonWriter

Situation: BalloonWriter will not open on your Macintosh, or will not run reliably without crashing.

Possible cause: BalloonWriter or the BalloonWriter INIT may be incompatible with one or more of the extensions you have loaded.

Solution: Remove all the files except BalloonWriter INIT from the Extensions folder inside your System Folder. Try opening BalloonWriter. If it works with no extensions loaded, you may be able to identify the extension causing the problem by replacing your extensions one by one. Restart your Macintosh and try BalloonWriter after replacing each extension. You may find that either a single extension or a combination of extensions causes some incompatibility with BalloonWriter.

You can't write balloons for a desk accessory or control panel

Situation: You cannot get balloons to work in a desk accessory or control panel.

Cause: BalloonWriter is not capable of working with desk accessories or control panels.

Menu balloons are not displayed properly

Situation: You have installed balloons in an application with menus that display different items in the same position (for example, Undo Cut Text and Undo Paste Text), and the menu items have the wrong balloons.

Cause: You may have neglected one or more of the menu items that can be displayed in the same position. In order for balloons to work properly with menus that display different items in the same position, you must provide a balloon for every possible menu item.

Alternative cause: When you installed the balloons, they may not have been in the proper order in the BalloonWriter document. All balloons associated with menu items that share a menu position must be listed together in the BalloonWriter document window when they are installed.

Solution: Define a balloon for every possible menu item, and ensure that they are listed together in the BalloonWriter document window, then reinstall the balloons.

Words in a balloon are broken into two lines

Situation: The text in an installed balloon does not wrap properly; rather than breaking a line at a space, one or more words are broken in the middle.

Cause: The Help Manager is unable to break the lines of text properly within the required size and shape of the balloon.

Solution: Rewrite the balloon so that it includes more words or shorter words.

You can't create a new custom balloon

Situation: You cannot create a new custom balloon.

Cause: You have chosen Menu Balloons, Dialog Balloons, or Window Balloons in the View Menu. When those views are selected, you cannot see custom balloons in the list, and you cannot create new custom balloons.

Solution: Choose All Balloons or Custom Balloons in the View menu.

Dialog balloons are not displayed properly

Situation: Your balloons were added to the application program by your programming team rather than being installed from BalloonWriter, and the balloons for dialog boxes appear for the wrong dialog box items.

Possible cause: A change was made in the number of balloons, in the number of dialog boxes, or in the number of items in a dialog box. The Rez file containing the balloons was not updated to reflect this change.

Solution: Repeat the process of installing balloons, using the DeRez tool, and exporting a Rez file for your programming team. For a description of this procedure, see the section "Making Sure Your Dialog Balloons Appear," in Chapter 2.

A balloon with a long message does not appear

Situation: A balloon with a long message (usually greater than 255 characters) does not appear.

Possible cause: The Help Manager limits the number of pixels that can be displayed in a balloon. A balloon with a long message may exceed the limit, and will not be displayed.

Solution: Rewrite the text for the balloon to shorten it. Reinstall the balloon and test it. Because the Help Manager limits pixels in a balloon, not characters, you may need to repeat this process more than once to get the balloon to work.

Index

The Apple Publishing System

This Apple® manual was written and edited on a desktop publishing system using Apple Macintosh® computers and Microsoft Word. Proof pages were created on Apple LaserWriter® printers. Final pages were produced in QuarkXPress. Screen shots were created with system software, Exposure, and SuperPaint.

Display type is Apple's corporate font, a condensed version of Garamond. Text type is Adobe Garamond. Ornaments are ITC Zapf Dingbats® and custom symbols designed for Apple Computer. Some elements, such as computer voice, are set in Apple Courier, a fixed-width font.

PostScript®, the LaserWriter page description language, was developed by Adobe Systems Incorporated.