

This chapter introduces the basic features of printing extensions and printer drivers and provides you with an overview of how printing works with QuickDraw GX. This chapter describes the printing process from the perspective of a printing-extension or printer-driver developer. To understand how printing works from the perspective of an application developer, you need to read *Inside Macintosh: QuickDraw GX Printing*.

Before reading this chapter, you should be familiar with the QuickDraw GX environment and QuickDraw GX objects. These features are described in *Inside Macintosh: QuickDraw GX Environment and Utilities* and *Inside Macintosh: QuickDraw GX Objects*.

This chapter begins with an overview of QuickDraw GX printing extensions and printer drivers. Next, this chapter describes how QuickDraw GX

- uses a message-passing architecture to accomplish printing
- supports three print imaging systems
- performs printing in several phases
- allows you to add panels to print dialog boxes
- uses collections for data storage and access
- provides messages that you can override to create drivers and extensions
- allows you to use resources to define a large portion of the functionality of extensions and drivers

Note

This chapter and book use the term **printer driver** to refer to the driver of any printing device, such as a printer, plotter, or imagesetter. ♦

About QuickDraw GX Printing Extensions and Printer Drivers

Printing extensions are add-on software components that you can create to extend the printing capabilities of applications. Printing extensions are used for tasks such as supporting hardware additions and modifying the appearance of printed pages and allow you to provide these capabilities without having to write an entire printer driver. When a user places a printing extension in the Extensions folder (which is in the System Folder), the extension is available for use.

Printer drivers translate the instructions that compose QuickDraw GX shapes and pictures into printed output on a specific output device. Each printer driver sends data and instructions in a form specific to the device that it drives and manages the physical communications with that device. You need to develop a separate driver for each hardware device that has different characteristics. Whenever you create a desktop printer, the Macintosh system software loads and uses the driver for that printer.

QuickDraw GX printing extensions and printer drivers are largely data-driven and take advantage of the QuickDraw GX message-based printing architecture, which is briefly described in the next section, “QuickDraw GX Printing and Messages.”

QuickDraw GX Printing and Messages

Printing with QuickDraw GX is based on a **message-passing architecture**. During the printing process, certain printing-related tasks often need to be done, or certain printing-related conditions arise. QuickDraw GX sends a large number of messages to notify such programs as an application, printing extension, and printer driver about these tasks and conditions. These messages are called **printing messages**.

Your printing extension or printer driver might need to respond to these printing messages to make available its features or functionality. For example, your printing extension could respond to a particular printing message called `GXDespoolPage` to add a date-time stamp to each printed page. Or your printer driver could respond to the `GXOpenConnection` message to verify that the corresponding printing device is working properly. This response of intercepting a specific printing message and taking some action is called a **message override**, which is performed by an override function that you define in your extension or driver.

Before you learn more about how message overrides work, it's important to have an overview of what happens when QuickDraw GX sends a printing message. Because your printing extension or printer driver may respond to a printing message, it is called a **message handler**. A number of message handlers, including your extension or driver, an application, and QuickDraw GX itself, can respond to printing messages.

Each message handler is part of the **message chain**, which links the handlers in a hierarchical sequence. Each handler in the hierarchy receives the message and decides whether to respond to the message and whether to forward the message to the handler below it in the hierarchy. A message handler provides a function to respond to any message that the handler wants to act upon.

QuickDraw GX provides a function to handle each message that it sends. These functions are called the **default implementations** of the messages and are activated whenever a message is not completely handled by another message. When you provide a function to handle a message in your printing extension or printer driver, you implement an **override function**, which can add to or replace the actions provided by other handlers of the message.

The declaration of each override function must exactly match the declaration of the message that it is overriding. You can use any name that you want for the override function, but you must declare the same return value and parameter types as are used for the message declaration. For example, one of the printing messages is named `GXCountPages` and has the following declaration:

```
OSErr GXCountPages(gxSpoolFile aSpoolFile, long *numPages);
```

If you override this message, you need to declare your override function with a matching header. For example,

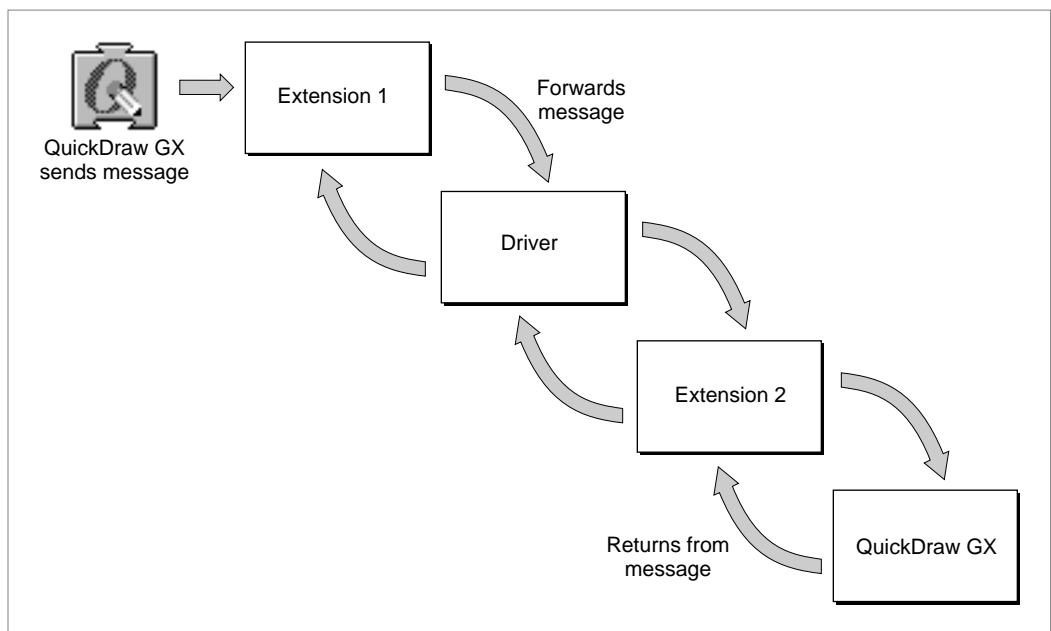
```
OSErr MyCountPages(gxSpoolFile myFile, long *numPages);
```

In most cases, when your printing extension or printer driver overrides a printing message, it performs some operations and forwards the message to the next handler

in the message chain. This is called a **partial override**. (A **total override** is when the message is not forwarded.) Although some of the default implementations of the printing messages provided by QuickDraw GX are empty and simply return, most of the default implementations do provide significant functionality. In many cases, you must allow the default implementation of a message to provide its actions; otherwise, a vital operation might be neglected, potentially resulting in serious errors.

A typical message chain, in which several handlers respond to and forward a message, is shown in Figure 1-1.

Figure 1-1 A printing message chain



Each printing message is described in the chapter “Printing Messages” in this book. Each description includes information about whether you must forward the message to allow the default implementation to provide its actions. Each description also specifies whether, for a partial override of a message, you need to forward the message before adding your own actions or after adding your own actions.

The timing of when you forward a message, relative to when you perform your actions in an override function, can result in significant differences. For example, if you are creating a printing extension that combines eight pages of a document into one page of thumbnail sketches, you might override the `GXCountPages` message, which QuickDraw GX sends to count the pages in the spool file. You would forward this message to allow the default implementation to count the pages, and your override would then modify that value. On the other hand, if you are creating an extension that adds a confidential stamp to each page as it gets spooled, you might override the

GXSpoolPage message. You would add your stamp to the page before forwarding the message to the default implementation, which would spool the page to file.

QuickDraw GX also provides a number of printing functions that you can only call from within the message overrides that you implement in your extension or driver. These functions perform a variety of operations, including

- displaying status information and the printing alert boxes to the user
- interfacing with the paper trays
- communicating imaging options between a driver and an application
- gracefully handling errors that arise during the processing of certain printing messages

The printing functions that you can call from within your message overrides are described in the chapter “Printing Functions for Message Overrides” in this book.

The message-passing architecture used for QuickDraw GX printing is supported by the Message Manager, which is a general-purpose software component of QuickDraw GX that you can use for message-passing in your own programs. The Message Manager and the concepts and terminology of message-based programming are described in *Inside Macintosh: QuickDraw GX Environment and Utilities*.

Print Imaging Systems

Some of the printing messages that QuickDraw GX sends are specific to the print imaging system that a printing device uses. A **print imaging system** is a part of the QuickDraw GX printing software that manages the conversion of QuickDraw GX shapes into data for a specific type of output device, such as a printer. QuickDraw GX supports four print imaging systems for printing documents, as shown in Table 1-1.

Table 1-1 Print imaging systems that QuickDraw GX supports

Imaging system	Explanation
Raster	For use with a raster output device such as an ImageWriter printer. Raster bitmap data and escape sequences are sent to accomplish the printing of each page.
PostScript™	For use with a PostScript output device such as an Apple LaserWriter printer. PostScript printing instructions are sent to accomplish the printing of each page.

Table 1-1 Print imaging systems that QuickDraw GX supports (continued)

Imaging system	Explanation
Vector	For use with a vector output device that uses a plotting language such as HPGL. QuickDraw GX shapes are converted into vectors, and the vector data and pen information are sent to the device to accomplish the printing of each page.
Portable digital document	For the creation of a portable digital document (PDD).

NOTE If you are writing a driver that creates portable digital documents for transportation to other types of computing systems, you specify that your driver works with the portable digital document (PDD) imaging system, which is described in *Inside Macintosh: QuickDraw GX Environment and Utilities*.

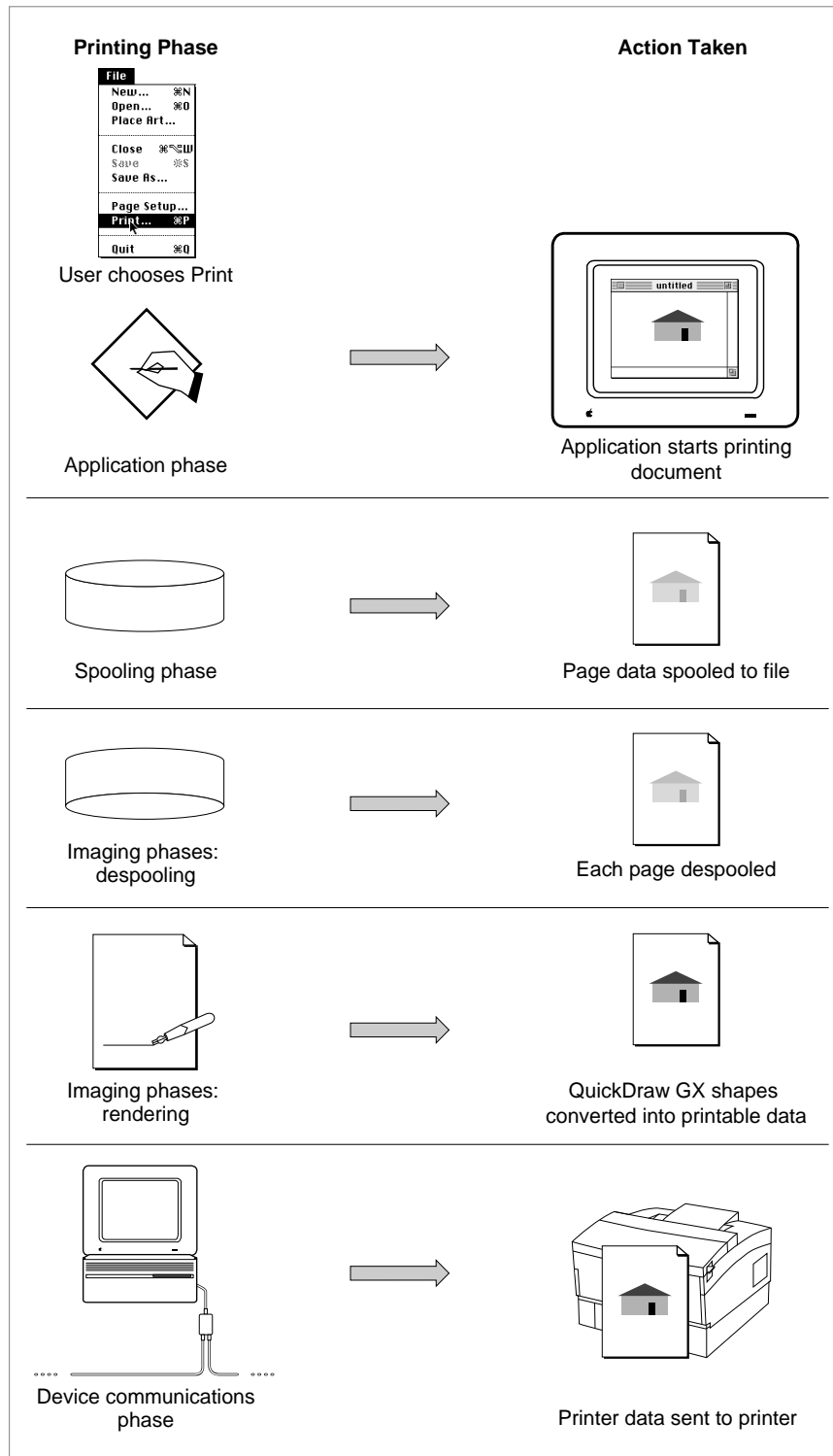
QuickDraw GX defines a few imaging messages for the raster and vector imaging systems and a large number of imaging messages for the PostScript imaging system. For each message, QuickDraw GX provides a default implementation that performs the basic task for which the message is intended. You can override any of these messages to customize the behavior for your extension or driver. In the resources that you provide, you specify which imaging system your extension or driver uses and which messages you are overriding.

When you develop a printing extension, you can choose to make the extension available for any or all of the imaging systems, depending on what tasks the extension needs to perform. When you develop a printer driver, you specify the appropriate imaging system for that printer. For example, if you are writing a raster printer driver, you specify that your driver works with the raster imaging system. You can then use the raster package controls ('ropt') resource to define the escape sequences used for performing line feeds on the printer. You can also override the GXRasterLineFeed message to implement how a line feed is performed.

If you are writing a PostScript printer driver, you specify that your driver works with the PostScript imaging system. You can then use the PostScript preferences ('pdip') resource to specify such things as which level of PostScript and which color space your driver supports. You can also override any of the numerous PostScript printing messages to exercise control over your device.

Printing Phases

QuickDraw GX sends specific printing messages during each of the four phases of printing. Figure 1-2 shows what happens to document data in these phases of printing.

Figure 1-2 The phases of QuickDraw GX printing

The four phases of printing are:

- The **application phase** of printing, during which the application calls QuickDraw GX and interacts with the user by displaying dialog boxes to establish printing parameters, such as page orientation and paper type. Examples of messages sent during this phase are `GXStartJob`, `GXPrintPage`, and `GXFinishJob`.
- The **spooling phase** of printing, during which the application spools the document pages to disk, in preparation for printing. QuickDraw GX sends messages during this phase to notify you when each page is about to be spooled. Examples of messages sent during this phase are `GXCreateSpoolFile`, `GXSpoolPage`, and `GXSpoolData`.
- The **imaging phase** of printing, during which each previously spooled page is rendered into a form that can be printed on the output device.

The imaging phase is actually composed of two processes:

- **Despooling**, during which each previously spooled page is read from the spool file. Examples of the despooling messages are `GXCountPages`, `GXDespoolPage`, `GXDespoolData`, `GXDespoolResource`, and `GXCloseSpoolFile`.
- **Rendering**, during which each despoiled page is converted into image data that can be printed by the output device. Some rendering messages, known as universal imaging messages, are sent for all imaging systems. Examples of these are the `GXImageJob`, `GXImagePage`, and `GXRenderPage` messages. Other messages sent during rendering depend on which imaging system is in use.

Examples of raster rendering messages are `GXRasterDataIn`, `GXRasterLineFeed`, and `GXRasterPackageBitmap`. Examples of PostScript rendering messages are `GXPostScriptQueryPrinter`, `GXPostScriptInitializePrinter`, `GXPostScriptStreamFont`, and `GXPostScriptDoPageSetup`. Examples of vector rendering messages are `GXVectorPackageShape`, `GXVectorLoadPens`, and `GXVectorVectorizeShape`.

- During the **device communications phase** of printing, the data that represents the rendered form of each page is sent to the output device. Messages sent during this phase include `GXOpenConnection`, `GXStartSendPage`, `GXWriteData`, and `GXCheckStatus`. You can only communicate with the printing device during this phase of printing, which means that you must override the device communications messages to communicate with the user regarding printer supplies such as paper.

The four phases of printing can be, but are not necessarily, sequential. In some cases, these phases are interleaved. For example, when the same computer is performing both formatting and output of a document, the application and spooling phases are interleaved, and the imaging and device communications phases are interleaved. The application initiates the printing of each page, and that output is written to a spool file. When the printer is ready, each page is despoiled, rendered, and sent to the device.

Printing extensions and printer drivers can override any of the messages that QuickDraw GX sends during each of these printing phases, allowing you a tremendous degree of flexibility in controlling printing on a specific device.

Note

The spooling, imaging, and device communications phases of printing can occur on different devices. For example, an application can spool a document to a printer server on a network, which might then image the document to a disk that is taken to a printing service and printed on a high-resolution printer. ♦

Extensions, Drivers, and the User Interface

QuickDraw GX sends printing messages when the user chooses commands that display such print dialog boxes as the Page Setup, Custom Page Setup, and Print dialog boxes. Your printing extension or printer driver can override the appropriate printing message to add one or more panels to one of these print dialog boxes. These panels become available to the user when the user clicks the More Choices button in the dialog box. When the user opens a panel that you added, your extension or driver can continue to receive and respond to messages as the user manipulates the panels, including messages that notify you if a user closes a panel or if a user confirms or cancels the print dialog box.

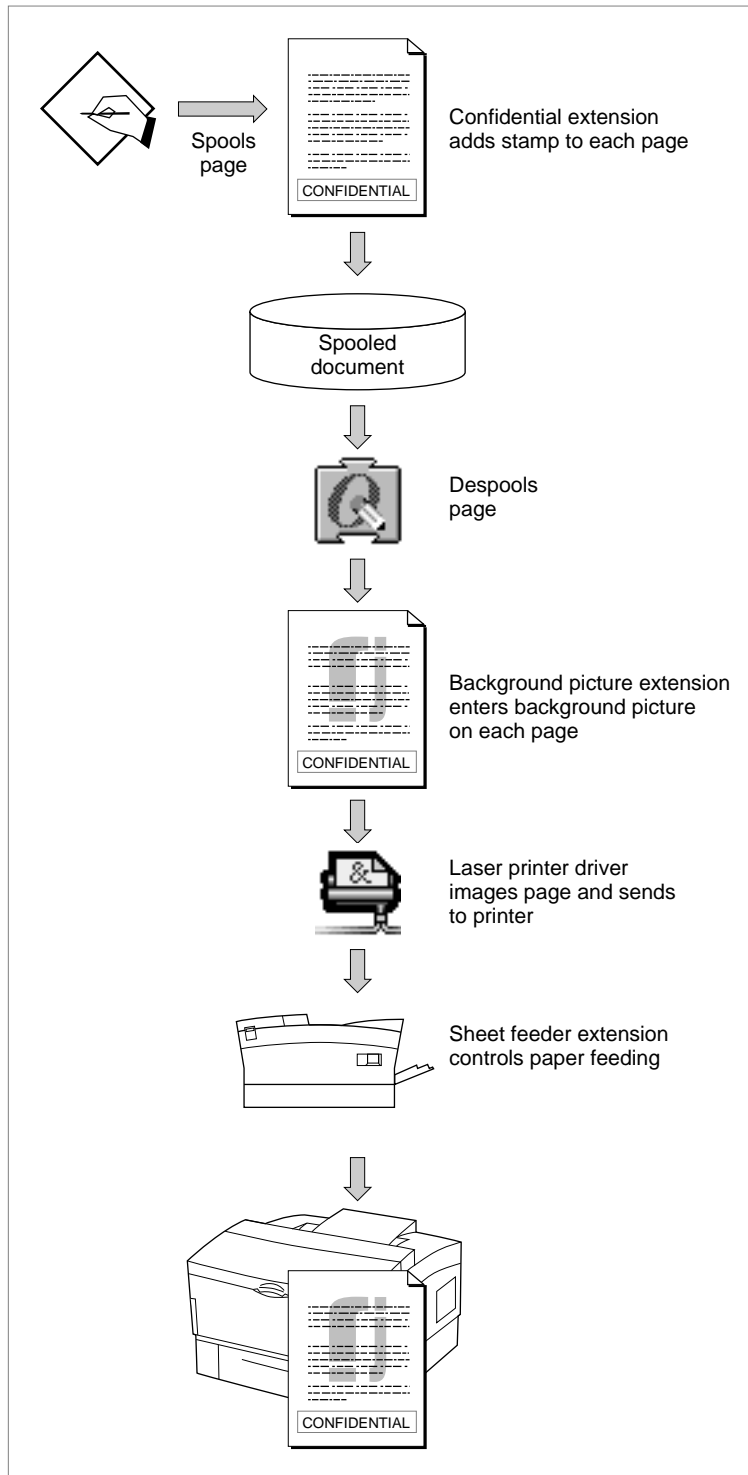
Much of the information that a user specifies in a print dialog box is stored in one of the collections that QuickDraw GX provides: the job collection, the format collection, or the paper-type collection. These collections add to the printing information already available in the corresponding print objects: the job object, the format object, and the paper-type object. Collections also provide data extensibility for printing extensions and printer drivers. For example, you can use the Collection Manager to create your own collection items for storing data related to the options a user chooses in a panel that your extension or driver adds to a dialog box. An example of how an extension uses a collection is in the chapter “Printing Extensions” in this book.

Adding panels to a dialog box is described in *Inside Macintosh: QuickDraw GX Printing*, and the messages that QuickDraw GX sends when a print dialog box is displayed or an event happens in a panel are in the section “Dialog Box Messages” beginning on page 4-81 in the chapter “Printing Messages.” Information about manipulating the job, format, and paper-type collections is in *Inside Macintosh: QuickDraw GX Printing*. Information about using the Collection Manager is in *Inside Macintosh: QuickDraw GX Environment and Utilities*.

Using Printing Extensions and Printer Drivers Together

While printer drivers are device dependent by definition, printing extensions can be device independent, which allows you to develop printing features that can be used with any number of devices.

The printing of a document always involves exactly one or two printer drivers: the formatting driver and the output driver, as described in *Inside Macintosh: QuickDraw GX Printing*. In many cases, formatting and output are handled by the same printer driver. However, any number of different printing extensions can be active at the same time. For example, you could create one extension that controls a sheet feeder that can be connected to numerous printers, another extension that draws a background picture on each printed page, and another extension that prints a confidential stamp on each page. These extensions could all be active at the same time as one printer driver, as shown in Figure 1-3.

Figure 1-3 Using multiple extensions concurrently

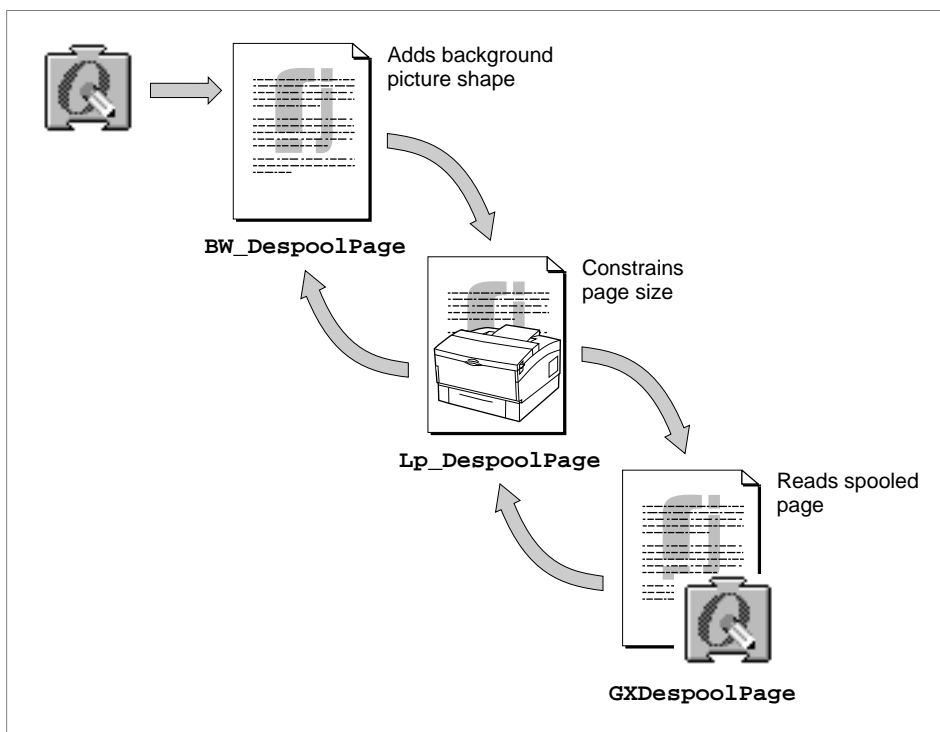
In Figure 1-3, the confidential stamp extension adds a confidential stamp at the bottom of the page. This extension overrides the `GXSpoolPage` message and adds the shape object that represents the confidential stamp to the shapes that compose the page when the page is sent to the spool file.

The background picture extension mentioned in Figure 1-3 adds a panel to the Print dialog box to allow the user to select a picture file to use. The user can choose not to have a background picture or can select a file. If the user selects a picture file, this extension overrides the `GXDespoolPage` message so that it can draw the background picture after the page has been despoiled and before it has been sent to the printer. The background picture extension is described extensively in the chapter “Printing Extensions” in this book.

The sheet feeder extension mentioned in Figure 1-3 works with a sheet-feeding device that can be added to laser printers produced by a number of manufacturers. This extension adds a panel to the Print dialog box to allow the user control over the sheet feeder and overrides several paper-handling messages to interface with the sheet feeder.

The laser printer driver that is mentioned in Figure 1-3 overrides a number of printing messages, including the `GXDespoolPage` message, which the background printing extension also overrides. The flow of control for the `GXDespoolPage` message in this case is shown in Figure 1-4, which includes only those message handlers that override the message.

Figure 1-4 Several functions handling the `GXDespoolPage` message



Whenever QuickDraw GX sends the `GXDespoolPage` message, as in this example, the following flow of control results.

1. The confidential stamp extension doesn't override this message, so QuickDraw GX doesn't send it to this extension.
2. The background picture extension overrides `GXDespoolPage` with the `BW_DespoolPage` function, which forwards the message before modifying the page.
3. The sheet feeder extension doesn't override this message, so QuickDraw GX doesn't forward it to this extension.
4. The laser printer driver overrides `GXDespoolPage` with the `Lp_DespoolPage` function, which forwards the message before modifying the page.
5. The QuickDraw GX default implementation of `GXDespoolPage` reads the graphics and formatting data for the page.
6. The laser printer driver override regains control and performs its operations to restrict the page size to the size supported by the device.
7. The background picture extension override regains control and performs its operations to add the background picture to the picture shape that represents the page.

One of the important issues that you have to decide when overriding a message in a printing extension or printer driver is whether to forward the message before adding your own code or after adding your own code. The timing of when your operations are performed can be critical, as is the case in the previous example. The chapter "Printing Messages" contains information about each printing message, which can help you to decide when timing might be an issue.

Any applications that use QuickDraw GX printing features can take advantage of extensions without any coding: the user controls whether extensions are applied to the printing of documents. In the example shown in Figure 1-3, the printed output of any application that prints while the extensions are enabled will use the background picture, confidential stamp, and selected paper-feeding options.

Using Resources to Create Printing Extensions and Printer Drivers

Printing extensions and printer drivers are very similar entities. Each contains a number of resources and overrides a certain number of the printing messages to provide its functionality. Each printing extension and printing driver includes resources for such diverse purposes as

- displaying extension, driver, and desktop printer icons
- displaying dialog box panels and defining controls
- specifying of which kind of device communications your driver uses, such as serial or PAP
- specifying which messages the extension or driver overrides

- defining when an extension needs to be loaded into memory
- specifying the imaging system (raster, vector, PostScript, or portable digital document) that your printer driver uses
- defining imaging options such as color and grayscale control
- defining status and alert conditions and displaying information about them

All of the resources that you need to define for printing extensions and printer drivers are described in the chapter “Printing Resources” in this book. Some resources are required for both extensions and drivers, some are required for only one, and some are optional.

QuickDraw GX allows you to define much of the functionality of your extensions and drivers in these printing resources. Given that QuickDraw GX also provides default implementations for most of the printing messages, you can develop many printing extensions and printer drivers without writing much code. In fact, some very useful printing extensions (including the confidential stamp extension) and some PostScript printer drivers have been written by overriding only one or two printing messages.

Overriding Printing Messages

When you decide that you need to override a certain printing message in a printing extension or printer driver, you need to follow these three steps:

1. Write the code to implement your override. You can name your override function with whatever name you want. You need to decide whether your override is going to forward the message to other message handlers (partially override) or not forward the message at all (totally override). If you are partially overriding the message, then you must decide at which point in your code to forward the message to other handlers.
2. Add the message name to one of the override (`'over'`) resources that you include with your extension or driver. Each entry in an override resource specifies the QuickDraw GX message ID, the code segment, and the offset of the jump statement to that function in the jump table. The override resource is described in the chapter “Printing Resources” in this book.
3. Add a jump statement for your message override to the assembly-language jump table that you must include in the code for your extension or driver.

The chapters “Printing Extensions” and “Printer Drivers” in this book provide examples of message overrides and override resources and describe the jump table.

Defining Components of the User Interface

You need to manage some user interface features in most printing extensions and all printer drivers. You can define most of these features in resources; however, there are some tasks that you need to implement in code.

When creating a printing extension, you need to define icons for your printing extension, you might need to manage the display of status information or printing alert boxes, and you might need to add panels to one or more of the print dialog boxes.

When creating a printer driver, you need to define icons for your printer driver, you must provide a Chooser interface, and you must manage the reporting of status and alert conditions to the user while printing is in progress. You display status information to which the user need not respond, and you display printing alert boxes to the user when a condition arises that requires a response, such as a paper jam. All of these tasks are described in the chapter “Printer Drivers” in this book.

In addition, you may want to add a panel to a print dialog box to provide options and controls to the user. Most printer drivers add panels to allow the user to specify options such as quality control or paper-type assignments. You can read about adding panels in your extensions and drivers in the chapters “Printing Extensions” and “Printer Drivers” in this book. You can also read about adding panels in *Inside Macintosh: QuickDraw GX Printing*.

Planning How to Write a Printing Extension or Printer Driver

You use similar development strategies for writing both printing extensions and printer drivers, including the following steps:

- Determine which imaging system(s) you are using.
- Determine which printing messages you need to override.
- Write the code to override those messages.
- Define the resources that provide data for QuickDraw GX to use with your extension or driver.

The chapter “Printing Extensions” in this book provides a detailed description of developing a printing extension. The chapter “Printer Drivers” in this book provides a detailed description of developing a printer driver.

The chapter “Printing Messages” in this book provides a complete reference guide to all of the printing messages that you can override to implement your printing extensions and printer drivers. The chapter “Printing Functions for Message Overrides” in this book provides a reference guide to all of the printing functions that you can call from within your message overrides. The chapter “Printing Resources” in this book provides a reference guide to all of the resources that you use to write your printing extensions and printer drivers.