
Carbon Printing Reference

[Carbon > Printing](#)



2007-07-24



Apple Inc.
© 2001, 2007 Apple Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, Carbon, Cocoa, Mac, Mac OS, Quartz, and QuickDraw are trademarks of Apple Inc., registered in the United States and other countries.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY

DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

Carbon Printing Reference 5

Overview	5
Functions by Task	5
Displaying the Page Setup and Print Dialogs	5
Print Loop Functions	6
Creating, Calling, and Deleting Universal Procedure Pointers	6
Customizing the Page Setup and Print Dialogs	7
Legacy Carbon Printing Functions	8
Functions	9
DisposePMSheetDoneUPP	9
InvokePMSheetDoneUPP	9
NewPMSheetDoneUPP	10
PMSessionBeginCGDocument	10
PMSessionBeginPage	11
PMSessionDisablePrinterPresets	12
PMSessionEnablePrinterPresets	13
PMSessionEndDocument	13
PMSessionEndPage	14
PMSessionPageSetupDialog	14
PMSessionPrintDialog	15
PMSessionUseSheets	16
PMShowPageSetupDialogAsSheet	17
PMShowPrintDialogWithOptions	18
PMShowPrintDialogWithOptionsAsSheet	19
Callbacks by Task	20
Sheet Dialog Callback	20
Legacy Callbacks	20
Callbacks	21
PMItemProcPtr	21
PMPageSetupDialogInitProcPtr	22
PMPrintDialogInitProcPtr	23
PMSheetDoneProcPtr	24
Data Types	25
PMItemUPP	25
PMPageSetupDialogInitUPP	25
PMPrintDialogInitUPP	26
PMSheetDoneUPP	26
Constants	26
Print Dialog Options	26

Appendix A Deprecated Carbon Printing Functions 29

Deprecated in Mac OS X v10.4	29
DisposePMItemUPP	29
DisposePMPageSetupDialogInitUPP	29
DisposePMPrintDialogInitUPP	30
InvokePMItemUPP	30
InvokePMPageSetupDialogInitUPP	31
InvokePMPrintDialogInitUPP	31
NewPMItemUPP	32
NewPMPageSetupDialogInitUPP	32
NewPMPrintDialogInitUPP	33
PMBeginDocument	33
PMBeginPage	34
PMEndDocument	34
PMEndPage	35
PMGetDialogAccepted	35
PMGetDialogDone	36
PMGetDialogPtr	36
PMGetItemProc	37
PMGetModalFilterProc	38
PMPageSetupDialog	38
PMPageSetupDialogInit	39
PMPageSetupDialogMain	40
PMPrintDialog	40
PMPrintDialogInit	41
PMPrintDialogInitWithPageFormat	41
PMPrintDialogMain	42
PMSessionPageSetupDialogInit	43
PMSessionPageSetupDialogMain	44
PMSessionPrintDialogInit	44
PMSessionPrintDialogMain	45
PMSetDialogAccepted	46
PMSetDialogDone	47
PMSetItemProc	48
PMSetModalFilterProc	48
Deprecated in Mac OS X v10.5	49
PMSessionBeginDocument	49

Document Revision History 51

Index 53

Carbon Printing Reference

Framework:	Carbon/Carbon.h
Companion guide	Supporting Printing in Your Carbon Application
Declared in	PMApplication.h PMApplicationDeprecated.h

Overview

Carbon Printing is a C API used by Carbon applications to display a user interface for printing. Carbon Printing provides functions to:

- Display the Page Setup dialog, which allows a user to modify the page format of a document to be printed
- Display the Print dialog, which allows a user to modify the default settings for a print job
- Execute a print loop that displays a printing status dialog

Carbon applications also need to use Core Printing to fully implement their printing features. For information about Core Printing, see *Core Printing Reference*.

Note: Carbon Printing and the Carbon Human Interface Toolbox are not available to 64-bit applications. To build a 64-bit application with a user interface, you must use Cocoa. For information about printing in Cocoa applications, see *Printing Programming Topics for Cocoa*.

Functions by Task

Displaying the Page Setup and Print Dialogs

[PMSessionUseSheets](#) (page 16)

Specifies that a printing dialog should be displayed as a sheet and specifies a function to call when the user dismisses the printing dialog.

[PMSessionPageSetupDialog](#) (page 14)

Displays the Page Setup dialog and records the user's selections in a page format object.

[PMSessionPrintDialog](#) (page 15)

Displays the Print dialog and records the user's selections in a print settings object.

- [PMShowPageSetupDialogAsSheet](#) (page 17)
Displays a Page Setup dialog as a sheet attached to a window.
- [PMShowPrintDialogWithOptions](#) (page 18)
Displays a Print dialog with a specified set of controls.
- [PMShowPrintDialogWithOptionsAsSheet](#) (page 19)
Displays a Print dialog with a specified set of controls as a sheet attached to a window.
- [PMSessionEnablePrinterPresets](#) (page 13)
Enables the use of printer presets in the Print dialog.
- [PMSessionDisablePrinterPresets](#) (page 12)
Disables the use of printer presets in the Print dialog.

Print Loop Functions

- [PMSessionBeginCGDocument](#) (page 10)
Begins a print job in which all drawing is to a Quartz graphics context. A printing status dialog informs the user of the job's progress.
- [PMSessionEndDocument](#) (page 13)
Ends a print job started by calling the function [PMSessionBeginCGDocument](#) (page 10) or [PMSessionBeginDocument](#) (page 49).
- [PMSessionBeginPage](#) (page 11)
Starts a new page for printing in the specified printing session.
- [PMSessionEndPage](#) (page 14)
Indicates the end of drawing the current page for the specified printing session.
- [PMSessionBeginDocument](#) (page 49) **Deprecated in Mac OS X v10.5**
Begins a print job in which, by default, all drawing is to a QuickDraw graphics port. A printing status dialog informs the user of the job's progress. (**Deprecated.** Use [PMSessionBeginCGDocument](#) (page 10) instead.)

Creating, Calling, and Deleting Universal Procedure Pointers

- [NewPMSheetDoneUPP](#) (page 10)
Creates a new universal procedure pointer (UPP) to a sheet-done callback.
- [InvokePMSheetDoneUPP](#) (page 9)
Calls a sheet-done callback.
- [DisposePMSheetDoneUPP](#) (page 9)
Disposes of a universal procedure pointer (UPP) to a sheet-done callback.
- [DisposePMenuItemUPP](#) (page 29) **Deprecated in Mac OS X v10.4**
Disposes of a universal procedure pointer (UPP) to a dialog item event handling callback. (**Deprecated.** Use a printing dialog extension instead.)
- [DisposePMPageSetupDialogInitUPP](#) (page 29) **Deprecated in Mac OS X v10.4**
Disposes of a universal procedure pointer (UPP) to a Page Setup dialog initialization callback. (**Deprecated.** Use a printing dialog extension instead.)

[DisposePMPrintDialogInitUPP](#) (page 30) **Deprecated in Mac OS X v10.4**

Disposes of a universal procedure pointer (UPP) to a Print dialog initialization callback function. **(Deprecated.** Use a printing dialog extension instead.)

[InvokePMItemUPP](#) (page 30) **Deprecated in Mac OS X v10.4**

Calls a dialog item event handling callback. **(Deprecated.** Use a printing dialog extension instead.)

[InvokePMPageSetupDialogInitUPP](#) (page 31) **Deprecated in Mac OS X v10.4**

Calls a Page Setup dialog initialization callback. **(Deprecated.** Use a printing dialog extension instead.)

[InvokePMPrintDialogInitUPP](#) (page 31) **Deprecated in Mac OS X v10.4**

Calls a Print dialog initialization callback. **(Deprecated.** Use a printing dialog extension instead.)

[NewPMItemUPP](#) (page 32) **Deprecated in Mac OS X v10.4**

Creates a new universal procedure pointer (UPP) to a dialog item event handling callback. **(Deprecated.** Use a printing dialog extension instead.)

[NewPMPageSetupDialogInitUPP](#) (page 32) **Deprecated in Mac OS X v10.4**

Creates a new universal procedure pointer (UPP) to a Page Setup dialog initialization callback. **(Deprecated.** Use a printing dialog extension instead.)

[NewPMPrintDialogInitUPP](#) (page 33) **Deprecated in Mac OS X v10.4**

Creates a new universal procedure pointer (UPP) to a Print dialog initialization callback. **(Deprecated.** Use a printing dialog extension instead.)

Customizing the Page Setup and Print Dialogs

The functions in this section make it possible to extend a printing dialog using the so-called `AppendDITL` method. Instead of using these functions, you should create a printing dialog extension. If you use these functions to extend a printing dialog, you cannot use sheets.

[PMGetDialogAccepted](#) (page 35) **Deprecated in Mac OS X v10.4**

Determines whether the user has confirmed a printing dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMGetDialogDone](#) (page 36) **Deprecated in Mac OS X v10.4**

Determines whether the user has finished with a dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMGetDialogPtr](#) (page 36) **Deprecated in Mac OS X v10.4**

Obtains the dialog reference for a dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMGetItemProc](#) (page 37) **Deprecated in Mac OS X v10.4**

Obtains the item proc callback function for a custom dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMGetModalFilterProc](#) (page 38) **Deprecated in Mac OS X v10.4**

Obtains the event handling callback function for a modal dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMSessionPageSetupDialogInit](#) (page 43) **Deprecated in Mac OS X v10.4**

Initializes a custom Page Setup dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMSessionPageSetupDialogMain](#) (page 44) **Deprecated in Mac OS X v10.4**

Displays your application's custom Page Setup dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMSessionPrintDialogInit](#) (page 44) **Deprecated in Mac OS X v10.4**

Initializes a custom Print dialog. **(Deprecated.** Use a printing dialog extension instead.)

[PMSessionPrintDialogMain](#) (page 45) **Deprecated in Mac OS X v10.4**

Displays your application's custom Print dialog. (**Deprecated.** Use a printing dialog extension instead.)

[PMSetDialogAccepted](#) (page 46) **Deprecated in Mac OS X v10.4**

Indicates that the user confirmed a custom dialog. (**Deprecated.** Use a printing dialog extension instead.)

[PMSetDialogDone](#) (page 47) **Deprecated in Mac OS X v10.4**

Indicates that the user finished with a custom dialog. (**Deprecated.** Use a printing dialog extension instead.)

[PMSetItemProc](#) (page 48) **Deprecated in Mac OS X v10.4**

Installs an item proc callback function for items in a custom dialog. (**Deprecated.** Use a printing dialog extension instead.)

[PMSetModalFilterProc](#) (page 48) **Deprecated in Mac OS X v10.4**

Installs an event handling callback function for a modal dialog. (**Deprecated.** Use a printing dialog extension instead.)

Legacy Carbon Printing Functions

The functions in this section are deprecated and have been replaced by session-based functions that take a `PMPrintSession` object. In some cases, the replacement function is also deprecated.

[PMBeginDocument](#) (page 33) **Deprecated in Mac OS X v10.4**

Establishes a graphics context for imaging a document. This context is an opaque `grafPort`. (**Deprecated.** Use [PMSessionBeginCGDocument](#) (page 10) instead.)

[PMBeginPage](#) (page 34) **Deprecated in Mac OS X v10.4**

Initializes a scaling rectangle for printing a page. (**Deprecated.** Use [PMSessionBeginPage](#) (page 11) instead.)

[PMEndDocument](#) (page 34) **Deprecated in Mac OS X v10.4**

Closes the context created for imaging a document. (**Deprecated.** Use [PMSessionEndDocument](#) (page 13) instead.)

[PMEndPage](#) (page 35) **Deprecated in Mac OS X v10.4**

Finishes printing the current page. (**Deprecated.** Use [PMSessionEndPage](#) (page 14) instead.)

[PMPageSetupDialog](#) (page 38) **Deprecated in Mac OS X v10.4**

Displays the Page Setup dialog and records the user's selections in a `PMPageFormat` object. (**Deprecated.** Use [PMSessionPageSetupDialog](#) (page 14) or [PMShowPageSetupDialogAsSheet](#) (page 17) instead.)

[PMPageSetupDialogInit](#) (page 39) **Deprecated in Mac OS X v10.4**

Initializes a custom Page Setup dialog. (**Deprecated.** Use a printing dialog extension instead.)

[PMPageSetupDialogMain](#) (page 40) **Deprecated in Mac OS X v10.4**

Displays your application's customized Page Setup dialog. (**Deprecated.** Use a printing dialog extension instead.)

[PMPrintDialog](#) (page 40) **Deprecated in Mac OS X v10.4**

Displays the Print dialog and records the user's selections in a `PMPrintSettings` object. (**Deprecated.** Use [PMSessionPrintDialog](#) (page 15), [PMShowPrintDialogWithOptions](#) (page 18), or [PMShowPrintDialogWithOptionsAsSheet](#) (page 19) instead.)

[PMPrintDialogInit](#) (page 41) **Deprecated in Mac OS X v10.4**

Initializes a custom Print dialog. (**Deprecated.** Use a printing dialog extension instead.)

[PMPrintDialogInitWithPageFormat](#) (page 41) **Deprecated in Mac OS X v10.4**

Initializes a custom Print dialog. (**Deprecated**. Use a printing dialog extension instead.)

[PMPrintDialogMain](#) (page 42) **Deprecated in Mac OS X v10.4**

Displays your application's custom Print dialog. (**Deprecated**. Use a printing dialog extension instead.)

Functions

DisposePMSheetDoneUPP

Disposes of a universal procedure pointer (UPP) to a sheet-done callback.

```
void DisposePMSheetDoneUPP (
    PMSheetDoneUPP userUPP
);
```

Parameters

userUPP

A UPP to your sheet-done callback.

Discussion

See the callback [PMSheetDoneProcPtr](#) (page 24) for more information.

Availability

Available in Mac OS X v10.0 and later.

Declared In

PMApplication.h

InvokePMSheetDoneUPP

Calls a sheet-done callback.

```
void InvokePMSheetDoneUPP (
    PMPrintSession printSession,
    WindowRef documentWindow,
    Boolean accepted,
    PMSheetDoneUPP userUPP
);
```

Discussion

You should not need to use the function `InvokePMSheetDoneUPP`, as the printing system calls your sheet-done callback function for you. See the callback [PMSheetDoneProcPtr](#) (page 24) for more information.

Availability

Available in Mac OS X v10.0 and later.

Declared In

PMApplication.h

NewPMSheetDoneUPP

Creates a new universal procedure pointer (UPP) to a sheet-done callback.

```
PMSheetDoneUPP NewPMSheetDoneUPP (
    PMSheetDoneProcPtr userRoutine
);
```

Parameters

userRoutine

A pointer to your sheet-done callback.

Return Value

On return, a UPP to the sheet-done callback.

Discussion

See the callback [PMSheetDoneProcPtr](#) (page 24) for more information.

Availability

Available in Mac OS X v10.0 and later.

Declared In

PMApplication.h

PMSessionBeginCGDocument

Begins a print job in which all drawing is to a Quartz graphics context. A printing status dialog informs the user of the job's progress.

```
OSStatus PMSessionBeginCGDocument (
    PMPrintSession printSession,
    PMPrintSettings printSettings,
    PMPageFormat pageFormat
);
```

Parameters

printSession

The printing session for the new print job.

printSettings

The print settings for the new print job.

pageFormat

The page format for the new print job.

Return Value

A result code. See Core Printing Result Codes.

Discussion

This function starts a print job in which your application draws in a Quartz graphics context, and should be called within your application's print loop. This function is similar to the function `PMSessionBeginCGDocumentNoDialog` except that the printing status dialog is displayed.

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`. If you present a printing dialog before you call this function, when calling this function you should use the same `PMPrintSession` object you used to present the dialog.

Before you call `PMSessionBeginCGDocument`, you should call `PMSessionValidatePrintSettings` and `PMSessionValidatePageFormat` to make sure the specified print settings and page format objects are updated and valid. After you call `PMSessionBeginCGDocument`, if you call a function that changes the specified print settings or page format object, the change is ignored for the current print job.

This function must be called before its corresponding End function (`PMSessionEndDocument` (page 13)). If the function `PMSessionBeginCGDocument` returns `noErr`, you must later call the End function, even if errors occur within the scope of the Begin and End functions.

The printing system automatically handles printing multiple copies. Your application does not need to perform any tasks other than specifying the number of copies in the printing session.

Availability

Available in Mac OS X v10.4 and later.

Not available to 64-bit applications.

See Also

`PMSessionBeginCGDocumentNoDialog`

Declared In

`PMApplication.h`

PMSessionBeginPage

Starts a new page for printing in the specified printing session.

```
OSStatus PMSessionBeginPage (
    PMPrintSession printSession,
    PMPageFormat pageFormat,
    const PMRect *pageFrame
);
```

Parameters

printSession

The printing session for the print job.

pageFormat

The page format for the new page. If you pass `NULL`, the printing system uses the page format you passed to `PMSessionBeginCGDocument` (page 10).

pageFrame

You should pass `NULL`, as this parameter is currently unsupported.

Return Value

A result code. If the user cancels the print job, this function returns `kPMCancel`.

Discussion

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`. You must call the functions `PMSessionBeginPage` and `PMSessionEndPage` (page 14) within the scope of calls to the Begin print job function (`PMSessionBeginCGDocument` (page 10)) and the End print job function (`PMSessionEndDocument` (page 13)).

You should call the function `PMSessionError` immediately before you call `PMSessionBeginPage`. If `PMSessionError` returns an error, then you should not call the function `PMSessionBeginPage`. Because `PMSessionBeginPage` also initializes the printing graphics context, your application should not make

assumptions about the state of the context (for example, the current font) between successive pages. After each call to `PMSessionBeginPage`, your application should call `PMSessionGetCGGraphicsContext` to obtain the current printing context.

If the function `PMSessionBeginPage` returns `noErr`, you must later call the function `PMSessionEndPage`, even if errors occur within the scope of `PMSessionBeginPage` and `PMSessionEndPage`.

The printing system automatically handles printing multiple copies. Your application does not need to perform any tasks other than specifying the number of copies in the printing session.

Special Considerations

Prior to Mac OS X v10.5, the *pageFormat* parameter is ignored. In Mac OS X v10.5 and later, the printing system supports multiple orientations within a print job. When you call this function and supply a page format, the orientation specified in the page format is used for the current page. Other settings in the page format, such as paper size or scaling, are ignored.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Related Sample Code

CarbonSketch

Declared In

`PMApplication.h`

PMSessionDisablePrinterPresets

Disables the use of printer presets in the Print dialog.

```
OSStatus PMSessionDisablePrinterPresets (
    PMPrintSession session
);
```

Parameters

session

The printing session you use to present the Print dialog.

Return Value

A result code. See Core Printing Result Codes.

Discussion

When you call this function with the specified printing session, the Print dialog for that session does not show any printer presets. Presets are disabled by default.

Availability

Available in Mac OS X v10.2 and later.

Not available to 64-bit applications.

See Also

[PMSessionEnablePrinterPresets](#) (page 13)

Declared In

`PMApplication.h`

PMSessionEnablePrinterPresets

Enables the use of printer presets in the Print dialog.

```
OSStatus PMSessionEnablePrinterPresets (
    PMPrintSession session,
    CFStringRef graphicsType
);
```

Parameters

session

The printing session you use to present the Print dialog.

graphicsType

The printer presets in the dialog should be suitable for rendering this type of graphic. Currently, the only defined graphics type is "Photo".

Return Value

A result code. See Core Printing Result Codes.

Discussion

When you call this function with the specified printing session, the Print dialog shows the presets available for the specified graphics type.

Availability

Available in Mac OS X v10.2 and later.

Not available to 64-bit applications.

See Also

[PMSessionDisablePrinterPresets](#) (page 12)

Declared In

PMApplication.h

PMSessionEndDocument

Ends a print job started by calling the function [PMSessionBeginCGDocument](#) (page 10) or [PMSessionBeginDocument](#) (page 49).

```
OSStatus PMSessionEndDocument (
    PMPrintSession printSession
);
```

Parameters

printSession

The current printing session. On return, the printing session is no longer valid; however, you must still call the function `PMRelease` to release the object.

Return Value

A result code. See Core Printing Result Codes.

Discussion

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`. The function `PMSessionEndDocument` is used to end a print job, and it should be called within your application's print loop after the call to `PMSessionEndPage` and before releasing the printing session.

This function must be called after its corresponding `Begin` function (`PMSessionBeginCGDocument` (page 10) or `PMSessionBeginDocument` (page 49)). If the `Begin` function returns `noErr`, the function `PMSessionEndDocument` must be called, even if errors occur within the scope of the `Begin` and `End` functions. You should not call `PMSessionEndDocument` if the `Begin` function returns an error.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Related Sample Code

CarbonSketch

Declared In

PMApplication.h

PMSessionEndPage

Indicates the end of drawing the current page for the specified printing session.

```
OSStatus PMSessionEndPage (
    PMPrintSession printSession
);
```

Parameters

printSession

The printing session for the print job.

Return Value

A result code. See Core Printing Result Codes.

Discussion

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`. You must call the functions `PMSessionBeginPage` (page 11) and `PMSessionEndPage` within the scope of calls to the `Begin` print job function (`PMSessionBeginCGDocument` (page 10)) and the `End` print job function (`PMSessionEndDocument` (page 13)).

If the function `PMSessionBeginPage` returns `noErr`, you must later call the function `PMSessionEndPage`, even if errors occur within the scope of `PMSessionBeginPage` and `PMSessionEndPage`. You should not call `PMSessionEndPage` if `PMSessionBeginPage` returns an error.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Related Sample Code

CarbonSketch

Declared In

PMApplication.h

PMSessionPageSetupDialog

Displays the Page Setup dialog and records the user's selections in a page format object.

```
OSStatus PMSessionPageSetupDialog (  
    PMPrintSession printSession,  
    PMPageFormat pageFormat,  
    Boolean *accepted  
);
```

Parameters

printSession

The printing session for which you want to display the dialog.

pageFormat

The page format object in which the user's selections are recorded.

accepted

A pointer to your Boolean variable. On return, `true` if the user clicks the OK button, or `false` if the user clicks Cancel. If you are using a sheet dialog, you should ignore the value returned in this parameter.

Return Value

A result code. See Core Printing Result Codes.

Discussion

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`. Before you call the function `PMSessionPageSetupDialog` you should call the function `PMSessionValidatePageFormat` to make sure page format settings are updated and valid. You should call the function `PMSessionPageSetupDialog` outside the scope of your print loop.

The Page Setup dialog shows the current settings contained in the page format object. If the user changes these settings and clicks the OK button, the page format object is updated by the printing system with the user's selections. If the user clicks the Cancel button, the page format object is returned unchanged.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Related Sample Code

CarbonSketch

Declared In

PMApplication.h

PMSessionPrintDialog

Displays the Print dialog and records the user's selections in a print settings object.

```
OSStatus PMSessionPrintDialog (  
    PMPrintSession printSession,  
    PMPrintSettings printSettings,  
    PMPageFormat constPageFormat,  
    Boolean *accepted  
);
```

Parameters

printSession

The printing session for which you want to display the dialog.

printSettings

The print settings object in which the user's selections are recorded.

constPageFormat

The page format object for the specified printing session.

accepted

A pointer to your Boolean variable. On return, `true` if the user accepts the dialog or `false` if the user cancels the dialog. If you are using a sheet dialog, you should ignore the value of this parameter.

Return Value

A result code. See Core Printing Result Codes.

Discussion

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`. Before you call the function `PMSessionPrintDialog` you should call the function `PMSessionValidatePrintSettings` to make sure print settings are updated and valid. You should call the function `PMSessionPrintDialog` outside the scope of your print loop.

The Print dialog shows the current settings in the print settings object. If the user changes these settings and accepts the dialog, the print settings object is updated by the printing system with the user's selections. If the user cancels the dialog, the print settings object is returned unchanged.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Related Sample Code

CarbonSketch

Declared In

`PMApplication.h`

PMSessionUseSheets

Specifies that a printing dialog should be displayed as a sheet and specifies a function to call when the user dismisses the printing dialog.

```
OSStatus PMSessionUseSheets (
    PMPrintSession printSession,
    WindowRef documentWindow,
    PMSheetDoneUPP sheetDoneProc
);
```

Parameters

printSession

The printing session for which you want to display the dialog.

documentWindow

The window to which the sheet dialog should be attached.

sheetDoneProc

A universal procedure pointer to your sheet-done function. The printing system calls this function when the user dismisses the dialog. Your sheet-done function is defined by the callback type [PMSheetDoneProcPtr](#) (page 24).

Return Value

A result code. See Core Printing Result Codes.

Discussion

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`.

When a printing dialog is displayed as a sheet, calls to the functions `PMSessionPageSetupDialog` (page 14) and `PMSessionPrintDialog` (page 15) return control immediately to your application, with the value of the `accepted` parameter undefined. The printing dialog sheet continues to be displayed until the user dismisses the dialog, at which time the printing system calls your sheet-done function. The page format and print settings objects associated with the dialog should not be used or changed until the sheet-done function is called.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

`PMApplication.h`

PMShowPageSetupDialogAsSheet

Displays a Page Setup dialog as a sheet attached to a window.

```
OSStatus PMShowPageSetupDialogAsSheet (
    PMPrintSession printSession,
    PMPageFormat pageFormat,
    WindowRef documentWindow,
    PMSheetDoneUPP sheetDoneProc
);
```

Parameters

printSession

The printing session for which you want to display the dialog.

pageFormat

The initial page format settings.

documentWindow

The window to which the Page Setup dialog sheet should be attached.

sheetDoneProc

A pointer to a sheet-done callback function that you provide. Your function is called when the user dismisses the dialog. See `PMSheetDoneProcPtr` (page 24) for information about how to implement this callback.

Return Value

A result code. See Core Printing Result Codes.

Discussion

This function is asynchronous; it displays the dialog in a sheet and returns immediately. When the user dismisses the dialog, the printing system calls the function specified in the *sheetDoneProc* parameter, passing it a Boolean value that indicates whether the user pressed the OK button.

Special Considerations

In Mac OS X v10.5 and later, you should use this function if you want to display the Page Setup dialog as a document-modal sheet. This function replaces the older method of calling `PMSessionUseSheets` (page 16) and `PMSessionPageSetupDialog` (page 14).

Availability

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared In

`PMApplication.h`

PMShowPrintDialogWithOptions

Displays a Print dialog with a specified set of controls.

```
OSStatus PMShowPrintDialogWithOptions (
    PMPrintSession printSession,
    PMPrintSettings printSettings,
    PMPageFormat pageFormat,
    PMPrintDialogOptionFlags printDialogOptions,
    Boolean *accepted
);
```

Parameters

printSession

The printing session for which you want to display the dialog.

printSettings

The initial print settings for the Print dialog.

pageFormat

The initial page format settings for the Print dialog.

printDialogOptions

Flags that specify what controls are displayed in the expanded version of the Print dialog. See [“Print Dialog Options”](#) (page 26) for a description of the flags you can use to specify the dialog options.

accepted

A pointer to a Boolean value. On return, `true` if the user accepts the dialog or `false` if the user cancels the dialog.

Return Value

A result code. See Core Printing Result Codes.

Discussion

In Mac OS X v10.5 and later, you can use this function to display the Print dialog with an optional set of controls in the expanded version of the dialog. When the Print dialog is first displayed, the user sees a simplified dialog with a disclosure button. If the button is pressed, the dialog expands and includes the controls specified in the *printDialogOptions* parameter. This function only allows you to specify controls for which flags are defined. The expanded dialog may also contain other controls that are not optional.

When you call this function, you should be prepared for the possibility that the page format settings may change. The expanded Print dialog could allow the user to modify page format settings explicitly. Even if the dialog is displayed with the default set of options, there may be other controls in the Print dialog that cause the page format settings to change.

This function is synchronous; it doesn't return until the user dismisses the dialog.

Availability

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared In

PMApplication.h

PMShowPrintDialogWithOptionsAsSheet

Displays a Print dialog with a specified set of controls as a sheet attached to a window.

```
OSStatus PMShowPrintDialogWithOptionsAsSheet (
    PMPrintSession printSession,
    PMPrintSettings printSettings,
    PMPageFormat pageFormat,
    PMPrintDialogOptionFlags printDialogOptions,
    WindowRef documentWindow,
    PMSheetDoneUPP sheetDoneProc
);
```

Parameters

printSession

The printing session for which you want to display the dialog.

printSettings

The initial print settings for the Print dialog.

pageFormat

The initial page format settings for the Print dialog.

printDialogOptions

Flags that specify what controls are displayed in the expanded version of the Print dialog. See [“Print Dialog Options”](#) (page 26) for a description of the flags you can use to specify dialog options.

documentWindow

The window to which the Print dialog sheet should be attached.

sheetDoneProc

A pointer to a sheet-done callback function that you provide. Your function is called when the user dismisses the dialog. See [PMSheetDoneProcPtr](#) (page 24) for information about how to implement this callback.

Return Value

A result code. See Core Printing Result Codes.

Discussion

In Mac OS X v10.5 and later, you can use this function to display the Print dialog with an optional set of controls in the expanded version of the dialog. When the Print dialog is first displayed, the user sees a simplified dialog with a disclosure button. If the button is pressed, the dialog expands and includes the controls specified in the *printDialogOptions* parameter. This function only allows you to specify controls for which flags are defined. The expanded dialog may also contain other controls that are not optional.

When you call this function, you should be prepared for the possibility that the page format settings may change. The expanded Print dialog could allow the user to modify page format settings explicitly. Even if the dialog is displayed with the default set of options, there may be other controls in the Print dialog that cause the page format settings to change.

This function is asynchronous; it displays the dialog in a sheet and returns immediately. When the user dismisses the dialog, the printing system calls the function specified in the *sheetDoneProc* parameter, passing it a Boolean value that indicates whether the user accepted or canceled the dialog.

Special Considerations

In Mac OS X v10.5 and later, you should use this function if you want to display the Print dialog as a document-modal sheet. This function replaces the older method of calling `PMSessionUseSheets` (page 16) and `PMSessionPrintDialog` (page 15).

Availability

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared In

`PMApplication.h`

Callbacks by Task

Sheet Dialog Callback

`PMSheetDoneProcPtr` (page 24)

Defines a pointer to a sheet-done function. The function is called when the user dismisses a printing dialog presented as a sheet.

Legacy Callbacks

`PMPageSetupDialogInitProcPtr` (page 22)

Defines a pointer to a dialog initialization function for your custom Page Setup dialog. (**Deprecated.** Use a printing dialog extension instead.)

`PMPrintDialogInitProcPtr` (page 23)

Defines a pointer to a dialog initialization function for your custom Print dialog. (**Deprecated.** Use a printing dialog extension instead.)

`PMItemProcPtr` (page 21)

Defines a pointer to a dialog item proc function that handles items you add to your custom Page Setup or Print dialog. (**Deprecated.** Use a printing dialog extension instead.)

Callbacks

PMItemProcPtr

Defines a pointer to a dialog item proc function that handles items you add to your custom Page Setup or Print dialog. (**Deprecated.** Use a printing dialog extension instead.)

```
typedef void (*PMItemProcPtr) (
    DialogRef theDialog,
    SInt16 item
);
```

You would declare your event handling function like this if you were to name it `MyPMItemCallback`:

```
void MyPMItemCallback (
    DialogRef theDialog,
    SInt16 item
);
```

Parameters

theDialog

A reference to your customized Page Setup or Print dialog (`PMDialog`). You can use the function `PMGetDialogPtr` to get a dialog reference.

item

The number of the dialog item.

Discussion

You should provide an item proc callback function if you add items to a custom dialog you create using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension.

Your function is called by the printing system after you create a custom Page Setup or Print dialog and register your dialog item proc function by calling the function `PMSetItemProc`. In Mac OS X, your function is called for items you add to your custom Page Setup or Print dialog. Your item proc function should call the function `PMSessionSetError` if the user cancels, otherwise the printing system won't know about the user cancellation.

To provide a pointer to your item proc function, you create a universal procedure pointer (UPP) of type `PMItemUPP`, using the function `NewPMItemUPP`. You can do so with code similar to the following:

```
PMItemUPP MyPMItemCallbackUPP;
MyPMItemCallbackUPP = NewPMItemUPP (&MyPMItemCallback);
```

After you are finished with your item proc function, you can dispose of the UPP with the function `DisposePMItemUPP`. However, if you will use the same event handling function for subsequent displays of a printing dialog, you can reuse the same UPP, rather than dispose of it and later create a new UPP.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMPageSetupDialogInitProcPtr

Defines a pointer to a dialog initialization function for your custom Page Setup dialog. (**Deprecated.** Use a printing dialog extension instead.)

```
typedef void (*PMPageSetupDialogInitProcPtr) (
    PMPageFormat pageFormat,
    PMDialog *theDialog
);
```

You would declare your dialog initialization function like this if you were to name it `MyPageSetupDialogInitCallback`:

```
void MyPageSetupDialogInitCallback (
    PMPageFormat pageFormat,
    PMDialog *theDialog
);
```

Parameters

pageFormat

A `PMPageFormat` object.

theDialog

A pointer to the `PMDialog` object representing your custom Page Setup dialog. You obtain this object from the function `PMSessionPageSetupDialogInit`. Your application should store a pointer to the dialog object in a global variable or as extended data in the `PMPageFormat` object, where it is accessible by your initialization function.

Discussion

You should set up a dialog initialization callback only if you need to use a custom dialog you create using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. Your initialization function can append items to the Page Setup dialog. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

You pass a pointer to your dialog initialization function as a parameter to the function `PMSessionPageSetupDialogMain`. Your dialog initialization function is called before your custom Page Setup dialog is displayed. Your initialization function can append items to the Page Setup dialog, and should install an item proc using the function `PMSetItemProc`.

To provide a pointer to your dialog initialization function, you create a universal procedure pointer (UPP) of type `PMPageSetupDialogInitUPP`, using the function `NewPMPageSetupDialogInitUPP`. You can do so with code similar to the following:

```
PMPageSetupDialogInitUPP MyPageSetupDialogInitCallbackUPP;
MyPageSetupDialogInitCallbackUPP = NewPMPageSetupDialogInitUPP
(MyPageSetupDialogInitCallback);
```

After you are finished with your dialog initialization function, you can dispose of the UPP with the function `DisposePMPageSetupDialogInitUPP`. However, if you plan to use the same dialog initialization function for subsequent display of the Page Setup dialog, you can reuse the same UPP, rather than dispose of it and later create a new UPP.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

PMPrintDialogInitProcPtr

Defines a pointer to a dialog initialization function for your custom Print dialog. (**Deprecated.** Use a printing dialog extension instead.)

```
typedef void (*PMPrintDialogInitProcPtr) (
    PMPrintSettings printSettings,
    PMDialog *theDialog
);
```

You would declare your dialog initialization function like this if you were to name it `MyPrintDialogInitCallback`:

```
void MyPrintDialogInitCallback (
    PMPrintSettings printSettings,
    PMDialog *theDialog
);
```

Parameters*printSettings*

A `PMPrintSettings` object.

theDialog

A pointer to the `PMDialog` object representing your custom Print dialog. You obtain this object from the function `PMSessionPrintDialogInit`. Your application should store a pointer to the dialog object in a global variable or as extended data in the `PMPrintSettings` object, where it is accessible by your initialization function.

Discussion

You should set up a dialog initialization callback if you need to use a custom dialog you create using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. Your initialization function can append items to the Print dialog. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

You pass a pointer to your dialog initialization function as a parameter to the function `PMSessionPrintDialogMain`. Your dialog initialization function is called before your custom Print dialog is displayed. Your initialization function can append items to the Print dialog, and should install an item proc using the function `PMSetitemProc`.

To provide a pointer to your dialog initialization function, you create a universal procedure pointer (UPP) of type `PMPrintDialogInitUPP`, using the function `NewPMPrintDialogInitUPP`. You can do so with code similar to the following:

```
PMPrintDialogInitUPP MyPrintDialogInitCallbackUPP;
MyPrintDialogInitCallbackUPP = NewPMPrintDialogInitUPP
(&MyPrintDialogInitCallback);
```

After you are finished with your dialog initialization function, you can dispose of the UPP with the function `DisposePMPrintDialogInitUPP`. However, if you will use the same dialog initialization function for subsequent displays of the Print dialog, you can reuse the same UPP, rather than dispose of it and later create a new UPP.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

PMSheetDoneProcPtr

Defines a pointer to a sheet-done function. The function is called when the user dismisses a printing dialog presented as a sheet.

```
typedef void (*PMSheetDoneProcPtr) (
    PMPrintSession printSession,
    WindowRef documentWindow,
    Boolean accepted
);
```

You would declare your sheet-done function like this if you were to name it `MyPrintSheetDoneCallback`:

```
void MyPrintSheetDoneCallback (
    PMPrintSession printSession,
    WindowRef documentWindow,
    Boolean accepted
);
```

Parameters

printSession

The printing session object for the dialog.

documentWindow

The document window to which the sheet is attached.

accepted

A Boolean value indicating whether the user accepted (`true`) or canceled (`false`) the dialog.

Discussion

You pass a universal procedure pointer to your sheet-done callback function as a parameter to the functions [PMSessionUseSheets](#) (page 16), [PMShowPrintDialogWithOptionsAsSheet](#) (page 19), and [PMShowPageSetupDialogAsSheet](#) (page 17). Your sheet-done function is called when the user dismisses the dialog. If the user accepted the dialog, your function should take appropriate action depending on whether the Page Setup or Print dialog is being shown.

To provide a pointer to your sheet-done function, you create a universal procedure pointer (UPP) of type `PMSheetDoneUPP`, using the function `NewPMSheetDoneUPP`. You can do so with code similar to the following:

```
PMSheetDoneUPP gMyPrintSheetDoneUPP;
gMyPrintSheetDoneUPP = NewPMSheetDoneUPP (&MyPrintSheetDoneProc);
```

You should declare your universal procedure pointer as a global variable to allow for multiple windows, each with a dialog available at one time.

When your print job is completed, you should use the function `DisposePMSheetDoneUPP` function to dispose of the universal procedure pointer associated with your sheet-done function. However, if you plan to use the same sheet-done function in subsequent print jobs, you can reuse the same UPP, rather than dispose of it and later create a new UPP.

The sheet-done function does not have a parameter for application-supplied data. Instead, you can attach any data your application needs to the window as a property using the Window Manager function `SetWindowProperty` and retrieve the data using the function `GetWindowProperty`.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

`PMApplication.h`

Data Types

PMItemUPP

A type that defines a universal procedure pointer to an item proc callback.

```
typedef PMItemProcPtr PMItemUPP;
```

Discussion

This data type is used by functions that are deprecated. For more information, see the description of the [PMItemProcPtr](#) (page 21) callback function.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMPageSetupDialogInitUPP

A type that defines a universal procedure pointer to a Page Setup dialog initialization callback.

```
typedef PMPageSetupDialogInitProcPtr PMPageSetupDialogInitUPP;
```

Discussion

This data type is used by functions that are deprecated. For more information, see the description of the [PMPageSetupDialogInitProcPtr](#) (page 22) callback function.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMPrintDialogInitUPP

A type that defines a universal procedure pointer to a Print dialog initialization callback.

```
typedef PMPrintDialogInitProcPtr PMPrintDialogInitUPP;
```

Discussion

This data type is used by functions that are deprecated. For more information, see the description of the [PMPrintDialogInitProcPtr](#) (page 23) callback function.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMSheetDoneUPP

A type that defines a universal procedure pointer to a sheet-done callback function.

```
typedef PMSheetDoneProcPtr PMSheetDoneUPP;
```

Discussion

For more information, see the description of the [PMSheetDoneProcPtr](#) (page 24) callback function.

Availability

Available in Mac OS X v10.0 and later.

Not available to 64-bit applications.

Declared In

`PMApplication.h`

Constants

Print Dialog Options

Flags that specify items to show in the expanded Print dialog in Mac OS X v10.5 and later.

```
typedef OptionBits PMPrintDialogOptionFlags;
enum {
    kPMHideInlineItems = (0L << 0),
    kPMShowDefaultInlineItems = (1L << 15),
    kPMShowInlineCopies = (1L << 0),
    kPMShowInlinePageRange = (1L << 1),
    kPMShowInlinePageRangeWithSelection = (1L << 6),
    kPMShowInlinePaperSize = (1L << 2),
    kPMShowInlineOrientation = (1L << 3),
    kPMShowInlineScale = (1L << 7),
    kPMShowPageAttributesPDE = (1L << 8),
};
```

Constants

`kPMHideInlineItems`

Show nothing in the inline portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowDefaultInlineItems`

Show the default set of items (copies & page range) in the inline portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowInlineCopies`

Show the copies edit-text field, the collated check box, and the two-sided check box (if the printer supports it) in the top portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowInlinePageRange`

Show the page range radio buttons and the from & to page edit-text fields in the top portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowInlinePageRangeWithSelection`

Show the page range radio buttons with the addition of a selection button and the From & To Page edit-text fields in the top portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowInlinePaperSize`

Show the formatting paper size popup menu in the top portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowInlineOrientation`

Show the page orientation buttons in the top portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowInlineScale`

Show the scale edit-text field in the top portion of the Print dialog.

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

`kPMShowPageAttributesPDE`

Add a separate pane to the Print dialog that contains the Page Setup dialog information (paper size, orientation, and scale).

Available in Mac OS X v10.5 and later.

Not available to 64-bit applications.

Declared in `PMApplication.h`.

Declared In

`PMApplication.h`

Deprecated Carbon Printing Functions

A function identified as deprecated has been superseded and may become unsupported in the future.

Deprecated in Mac OS X v10.4

DisposePMItemUPP

Disposes of a universal procedure pointer (UPP) to a dialog item event handling callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
void DisposePMItemUPP (
    PMItemUPP userUPP
);
```

Parameters

userUPP

The universal procedure pointer.

Discussion

See the [PMItemProcPtr](#) (page 21) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

DisposePMPageSetupDialogInitUPP

Disposes of a universal procedure pointer (UPP) to a Page Setup dialog initialization callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
void DisposePMPageSetupDialogInitUPP (
    PMPageSetupDialogInitUPP userUPP
);
```

Parameters

userUPP

The universal procedure pointer.

Discussion

See the [PMPageSetupDialogInitProcPtr](#) (page 22) callback function.

Deprecated Carbon Printing Functions

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

DisposePMPrintDialogInitUPP

Disposes of a universal procedure pointer (UPP) to a Print dialog initialization callback function. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
void DisposePMPrintDialogInitUPP (
    PMPrintDialogInitUPP userUPP
);
```

Parameters

userUPP

The universal procedure pointer.

Discussion

See the [PMPrintDialogInitProcPtr](#) (page 23) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

InvokePMItemUPP

Calls a dialog item event handling callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
void InvokePMItemUPP (
    DialogRef theDialog,
    short item,
    PMItemUPP userUPP
);
```

Discussion

You should not need to use the function `InvokePMInvokeUPP`, as the printing system calls your item event handling callback function for you. See the [PMItemProcPtr](#) (page 21) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Deprecated Carbon Printing Functions

Declared In

PMApplicationDeprecated.h

InvokePMPageSetupDialogInitUPP

Calls a Page Setup dialog initialization callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
void InvokePMPageSetupDialogInitUPP (
    PMPageFormat pageFormat,
    PMDialog *theDialog,
    PMPageSetupDialogInitUPP userUPP
);
```

Discussion

You should not need to use the function `InvokePMPageSetupDialogInitUPP`, as the printing system calls your Page Setup dialog initialization callback function for you. See the [PMPageSetupDialogInitProcPtr](#) (page 22) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

InvokePMPrintDialogInitUPP

Calls a Print dialog initialization callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
void InvokePMPrintDialogInitUPP (
    PMPrintSettings printSettings,
    PMDialog *theDialog,
    PMPrintDialogInitUPP userUPP
);
```

Discussion

You should not need to use the function `InvokePMPrintDialogInitUPP`, as the printing system calls your Print dialog initialization callback function for you. See the [PMPrintDialogInitProcPtr](#) (page 23) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

NewPMItemUPP

Creates a new universal procedure pointer (UPP) to a dialog item event handling callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
PMItemUPP NewPMItemUPP (
    PMItemProcPtr userRoutine
);
```

Parameters

userRoutine

A pointer to your dialog item event handling callback.

Return Value

On return, a UPP to the dialog item event handling callback.

Discussion

See the [PMItemProcPtr](#) (page 21) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

NewPMPageSetupDialogInitUPP

Creates a new universal procedure pointer (UPP) to a Page Setup dialog initialization callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
PMPageSetupDialogInitUPP NewPMPageSetupDialogInitUPP (
    PMPageSetupDialogInitProcPtr userRoutine
);
```

Parameters

userRoutine

A pointer to your Page Setup dialog initialization callback.

Return Value

On return, a UPP to the Page Setup dialog initialization callback.

Discussion

See the [PMPageSetupDialogInitProcPtr](#) (page 22) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

NewPMPrintDialogInitUPP

Creates a new universal procedure pointer (UPP) to a Print dialog initialization callback. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
PMPrintDialogInitUPP NewPMPrintDialogInitUPP (
    PMPrintDialogInitProcPtr userRoutine
);
```

Parameters

userRoutine

A pointer to your Print dialog initialization callback.

Return Value

On return, a UPP to the Print dialog initialization callback.

Discussion

See the [PMPrintDialogInitProcPtr](#) (page 23) callback function.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

PMBeginDocument

Establishes a graphics context for imaging a document. This context is an opaque `grafPort`. (Deprecated in Mac OS X v10.4. Use [PMSessionBeginCGDocument](#) (page 10) instead.)

```
OSStatus PMBeginDocument (
    PMPrintSettings printSettings,
    PMPageFormat pageFormat,
    PMPrintContext *printContext
);
```

Parameters

printSettings

A `PMPrintSettings` object.

pageFormat

A `PMPageFormat` object.

printContext

On return, an initialized `PMPrintContext` object.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid within a `PMBegin/PMEnd` block. You must balance a call to `PMBeginDocument` with a call to `PMEndDocument`.

Availability

Available in Mac OS X v10.0 and later.

Deprecated Carbon Printing Functions

Deprecated in Mac OS X v10.4.

Declared In

PMApplicationDeprecated.h

PMBeginPage

Initializes a scaling rectangle for printing a page. (Deprecated in Mac OS X v10.4. Use [PMSessionBeginPage](#) (page 11) instead.)

```
OSStatus PMBeginPage (
    PMPrintContext printContext,
    const PMRect *pageFrame
);
```

Parameters

printContext

A `PMPrintContext` object.

pageFrame

A pointer to a bounding rectangle for drawing the page. This value is usually obtained from the function `PMGetAdjustedRect`, but if no scaling is needed, pass `NULL`. In Mac OS X, this parameter is ignored.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin` and `PMBeginDocument`.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

PMApplicationDeprecated.h

PMEndDocument

Closes the context created for imaging a document. (Deprecated in Mac OS X v10.4. Use [PMSessionEndDocument](#) (page 13) instead.)

```
OSStatus PMEndDocument (
    PMPrintContext printContext
);
```

Parameters

printContext

On return, an invalidated `PMPrintContext` object.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin` and `PMBeginDocument`.

Deprecated Carbon Printing Functions

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

PMApplicationDeprecated.h

PMEndPage

Finishes printing the current page. (Deprecated in Mac OS X v10.4. Use [PMSessionEndPage](#) (page 14) instead.)

```
OSStatus PMEndPage (
    PMPrintContext printContext
);
```

Parameters

printContext

A `PMPrintContext` object.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin`, `PMBeginDocument`, and `PMBeginPage`.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

PMApplicationDeprecated.h

PMGetDialogAccepted

Determines whether the user has confirmed a printing dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMGetDialogAccepted (
    PMDialog pmDialog,
    Boolean *process
);
```

Parameters

pmDialog

A `PMDialog` object representing your customized Page Setup or Print dialog.

process

Returns `true` if the user confirms the dialog. In the case of the Page Setup dialog, this means the user clicked the OK button; in the case of the Print dialog, this means the user clicked the Print or Preview button. Returns `false` if the user clicks the Cancel button.

Return Value

A result code. See Core Printing Result Codes.

Deprecated Carbon Printing Functions

Discussion

The function `PMGetDialogAccepted` is valid between the creation and release of a printing session and while displaying your custom Page Setup or Print dialog.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMGetDialogDone

Determines whether the user has finished with a dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMGetDialogDone (
    PMDialog pmDialog,
    Boolean *done
);
```

Parameters

pmDialog

A `PMDialog` object representing your customized Page Setup or Print dialog.

done

Returns `true` if the user clicked the OK, Print, or Cancel button; `false` if the user did not click any of these buttons.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMGetDialogDone` is valid between the creation and release of a printing session and while displaying your Page Setup or Print dialog.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMGetDialogPtr

Obtains the dialog reference for a dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

Deprecated Carbon Printing Functions

```
OSStatus PMGetDialogPtr (
    PMDialog pmDialog,
    DialogRef *theDialog
);
```

Parameters

pmDialog

The `PMDialog` object from which you wish to extract the `DialogRecord` structure.

theDialog

On return, a pointer to a Dialog Manager `DialogRef` object.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMGetDialogPtr` is valid between the creation and release of a printing session and after calling `PMSessionPageSetupDialogInit` to initialize a Page Setup dialog, or calling `PMSessionPrintDialogInit` to initialize a Print dialog.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMGetItemProc

Obtains the item proc callback function for a custom dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMGetItemProc (
    PMDialog pmDialog,
    PMItemUPP *itemProc
);
```

Parameters

pmDialog

A `PMDialog` object representing the Page Setup or Print dialog for which you want to obtain the item proc function.

itemProc

On return, a pointer to the current dialog item proc function.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMGetItemProc` is valid only between the creation and release of a printing session. You can call `PMGetItemProc` to get the existing item proc function before you replace it with a call to the function `PMSetItemProc`. You should call this function only if you need to handle events in a custom dialog you create using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

Deprecated Carbon Printing Functions

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

PMGetModalFilterProc

Obtains the event handling callback function for a modal dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMGetModalFilterProc (
    PMDialog pmDialog,
    ModalFilterUPP *filterProc
);
```

Parameters

pmDialog

A `PMDialog` object representing the Page Setup or Print dialog for which you want to obtain the event handling function.

filterProc

On return, a pointer to the current dialog event handling function.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMGetModalFilterProc` is valid only between the creation and release of a printing session. You can call `PMGetModalFilterProc` to get the existing event handling function before you replace it with a call to the function `PMSetModalFilterProc`. The event handling function is not called in Mac OS X.

You should call the function `PMGetModalFilterProc` only if you create a custom dialog using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

PMPageSetupDialog

Displays the Page Setup dialog and records the user's selections in a `PMPageFormat` object. (Deprecated in Mac OS X v10.4. Use `PMSessionPageSetupDialog` (page 14) or `PMShowPageSetupDialogAsSheet` (page 17) instead.)

Deprecated Carbon Printing Functions

```
OSStatus PMPageSetupDialog (
    PMPageFormat pageFormat,
    Boolean *accepted
);
```

Parameters

pageFormat

A PMPageFormat object.

accepted

Returns `true` if the user clicks the OK button, or `false` if the user clicks Cancel.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin` and creating a page format object. Never call `PMPageSetupDialog` between the pages of a document.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

`PMApplicationDeprecated.h`

PMPageSetupDialogInit

Initializes a custom Page Setup dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMPageSetupDialogInit (
    PMPageFormat pageFormat,
    PMDialog *newDialog
);
```

Parameters

pageFormat

A PMPageFormat object.

newDialog

On return, a pointer to an initialized `PMDialog` object, ready for customization by your application. Because the `PMPageSetupDialogMain` function does not include a parameter for passing this `PMDialog` object to your dialog initialization callback function, your application should store this pointer in a global variable or as extended data in the `PMPageFormat` object. See the discussion of the `PMPageSetupDialogInitProcPtr` callback function for more information.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin` and creating a page format object.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Deprecated Carbon Printing Functions

Declared In

PMAplicationDeprecated.h

PMPageSetupDialogMain

Displays your application's customized Page Setup dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMPageSetupDialogMain (
    PMPageFormat pageFormat,
    Boolean *accepted,
    PMPageSetupDialogInitUPP myInitProc
);
```

Parameters*pageFormat*

A PMPageFormat object.

accepted

Returns true if the user clicks the OK button, or false if the user clicks Cancel.

myInitProc

A universal procedure pointer to your dialog initialization function. Your initialization function is defined by the callback PMPageSetupDialogInitProcPtr.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling PMPBegin and creating a page format object

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

PMAplicationDeprecated.h

PMPrintDialog

Displays the Print dialog and records the user's selections in a PMPrintSettings object. (Deprecated in Mac OS X v10.4. Use PMSessionPrintDialog (page 15), PMShowPrintDialogWithOptions (page 18), or PMShowPrintDialogWithOptionsAsSheet (page 19) instead.)

```
OSStatus PMPrintDialog (
    PMPrintSettings printSettings,
    PMPageFormat constPageFormat,
    Boolean *accepted
);
```

Parameters*printSettings*

A PMPrintSettings object.

APPENDIX A

Deprecated Carbon Printing Functions

constPageFormat

A `PMPageFormat` object.

accepted

Returns `true` if the user clicks the OK button, or `false` if the user clicks Cancel.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin` and creating a page format and print settings object.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

`PMApplicationDeprecated.h`

PMPrintDialogInit

Initializes a custom Print dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMPrintDialogInit (  
    PMPrintSettings printSettings,  
    PMDialog *newDialog  
);
```

Parameters

printSettings

A `PMPrintSettings` object.

newDialog

On return, a pointer to an initialized `PMDialog` object, ready for customization by your application. Because the `PMPrintDialogMain` function does not include a parameter for passing this `PMDialog` object to your dialog initialization callback function, your application should store this pointer in a global variable or as extended data in the `PMPrintSettings` object. See the discussion of the `PMPrintDialogInitProcPtr` callback function for more information.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin` and creating a page format and print settings object.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

`PMApplicationDeprecated.h`

PMPrintDialogInitWithPageFormat

Initializes a custom Print dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

Deprecated Carbon Printing Functions

```
OSStatus PMPrintDialogInitWithPageFormat (
    PMPrintSettings printSettings,
    PMPageFormat constPageFormat,
    PMDialog *newDialog
);
```

Parameters*printSettings*

A PMPrintSettings object.

constPageFormat

A PMPageFormat object.

newDialog

On return, a pointer to an initialized PMDialog object, ready for customization by your application. Because the PMPrintDialogMain function does not include a parameter for passing this PMDialog object to your dialog initialization callback function, your application should store this pointer in a global variable or as extended data in the PMPrintSettings object. See the discussion of the PMPrintDialogInitProcPtr callback function for more information.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling PMBegin and creating a page format and print settings object.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

PMApplicationDeprecated.h

PMPrintDialogMain

Displays your application's custom Print dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMPrintDialogMain (
    PMPrintSettings printSettings,
    PMPageFormat constPageFormat,
    Boolean *accepted,
    PMPrintDialogInitUPP myInitProc
);
```

Parameters*printSettings*

A PMPrintSettings object.

constPageFormat

A PMPageFormat object.

accepted

Returns true if the user clicks the OK button, or false if the user clicks Cancel.

Deprecated Carbon Printing Functions

myInitProc

A universal procedure pointer to your dialog initialization function. Your initialization function is defined by the callback `PMPrintDialogInitProcPtr`.

Return Value

A result code. See Core Printing Result Codes.

Discussion

Valid after calling `PMBegin` and creating a page format and print settings object.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Declared In

`PMApplicationDeprecated.h`

PMSessionPageSetupDialogInit

Initializes a custom Page Setup dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMSessionPageSetupDialogInit (
    PMPrintSession printSession,
    PMPageFormat pageFormat,
    PMDialog *newDialog
);
```

Parameters*printSession*

The current printing session.

pageFormat

A page format object.

newDialog

On return, a pointer to an initialized `PMDialog` object, ready for customization by your application. Because the `PMSessionPageSetupDialogMain` (page 44) function does not include a parameter for passing this `PMDialog` object to your dialog initialization callback function, your application should store this pointer in a global variable or as extended data in the `PMPageFormat` object.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSessionPageSetupDialogInit` is valid only between the creation and release of a printing session and after creating a page format object. You should call this function only if you create a custom dialog using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

Deprecated Carbon Printing Functions

PMSessionPageSetupDialogMain

Displays your application's custom Page Setup dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMSessionPageSetupDialogMain (
    PMPrintSession printSession,
    PMPageFormat pageFormat,
    Boolean *accepted,
    PMPageSetupDialogInitUPP myInitProc
);
```

Parameters

printSession

The current printing session.

pageFormat

A page format object.

accepted

A pointer to your Boolean variable. On return, `true` if the user clicks the OK button, or `false` if the user clicks Cancel.

myInitProc

A universal procedure pointer to your dialog initialization function. Your initialization function is defined by the callback `PMPageSetupDialogInitProcPtr`.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSessionPageSetupDialogMain` is valid only between the creation and release of a printing session and after calling the function `PMSessionPageSetupDialogInit`. You should call this function only if you create a custom dialog using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

Your dialog initialization function is called before your custom Page Setup dialog is displayed. Your initialization function can append items to the Page Setup dialog, and should install an item proc using the `PMSetItemProc` (page 48) function. You must pass the same page format object to each of the functions `PMSessionPageSetupDialogMain` and `PMSessionPageSetupDialogInit` (page 43).

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMSessionPrintDialogInit

Initializes a custom Print dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

Deprecated Carbon Printing Functions

```
OSStatus PMSessionPrintDialogInit (
    PMPrintSession printSession,
    PMPrintSettings printSettings,
    PMPageFormat constPageFormat,
    PMDialog *newDialog
);
```

Parameters*printSession*

The current printing session.

printSettings

A print settings object.

constPageFormat

A page format object.

newDialog

On return, a pointer to an initialized `PMDialog` object, ready for customization by your application. Because the `PMSessionPrintDialogMain` (page 45) function does not include a parameter for passing this `PMDialog` object to your dialog initialization callback function, your application should store this pointer in a global variable or as extended data in the `PMPrintSettings` object. See the discussion of the `PMPrintDialogInitProcPtr` (page 23) callback function for more information.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSessionPrintDialogInit` is valid only between the creation and release of a printing session and after creating a print settings object. You should call this function only if you create a custom dialog using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In`PMApplicationDeprecated.h`**PMSessionPrintDialogMain**

Displays your application's custom Print dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

Deprecated Carbon Printing Functions

```
OSStatus PMSessionPrintDialogMain (
    PMPrintSession printSession,
    PMPrintSettings printSettings,
    PMPageFormat constPageFormat,
    Boolean *accepted,
    PMPrintDialogInitUPP myInitProc
);
```

Parameters*printSession*

The current printing session.

printSettings

A print settings object.

constPageFormat

A page format object.

*accepted*A pointer to your Boolean variable. On return, `true` if the user accepts the dialog or `false` if the user cancels the dialog.*myInitProc*A universal procedure pointer to your dialog initialization function. Your initialization function is defined by the callback `PMPrintDialogInitProcPtr`**Return Value**

A result code. See Core Printing Result Codes.

Discussion

The function `PMSessionPrintDialogMain` is valid only between the creation and release of a printing session and after calling the function `PMSessionPrintDialogInit`. You should call this function only if you create a custom dialog using the `AppendDITL` method. In Mac OS X, you should create a custom dialog by writing a printing dialog extension. If you use this function to extend a printing dialog in Mac OS X, you cannot use sheets.

Your dialog initialization function is called before your custom Print dialog is displayed. Your initialization function can append items to the Print dialog, and should install an item proc using the `PMSetItemProc` (page 48) function. You must pass the same page format and print settings objects to each of the functions `PMSessionPrintDialogMain` and `PMSessionPrintDialogInit` (page 44).

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In`PMApplicationDeprecated.h`**PMSetDialogAccepted**

Indicates that the user confirmed a custom dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

Deprecated Carbon Printing Functions

```
OSStatus PMSetDialogAccepted (
    PMDialog pmDialog,
    Boolean process
);
```

Parameters*pmDialog*

A `PMDialog` object representing your customized Page Setup or Print dialog.

process

Pass `true` if the user confirms the dialog. In the case of the Page Setup dialog, this means the user clicked the OK button; in the case of the Print dialog, this means the user clicked the Print button. Pass `false` if the user clicks the Cancel button.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSetDialogAccepted` is valid between the creation and release of a printing session and while displaying your custom Page Setup or Print dialog.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

PMSetDialogDone

Indicates that the user finished with a custom dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMSetDialogDone (
    PMDialog pmDialog,
    Boolean done
);
```

Parameters*pmDialog*

A `PMDialog` object representing your custom Page Setup or Print dialog.

done

Pass `true` when your callback function has finished with this dialog.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSetDialogDone` is valid between the creation and release of a printing session and while displaying your custom Page Setup or Print dialog.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Deprecated Carbon Printing Functions

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

PMSetItemProc

Installs an item proc callback function for items in a custom dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMSetItemProc (
    PMDialog pmDialog,
    PMItemUPP itemProc
);
```

Parameters

pmDialog

A `PMDialog` object representing your custom Page Setup or Print dialog.

itemProc

A universal procedure pointer to your dialog item event handling function.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSetItemProc` is valid between the creation and release of a printing session and after creating your custom Page Setup or Print dialog.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

PMApplicationDeprecated.h

PMSetModalFilterProc

Installs an event handling callback function for a modal dialog. (Deprecated in Mac OS X v10.4. Use a printing dialog extension instead.)

```
OSStatus PMSetModalFilterProc (
    PMDialog pmDialog,
    ModalFilterUPP filterProc
);
```

Parameters

pmDialog

A `PMDialog` object representing the Page Setup or Print dialog for which you want to install an event handling function.

Deprecated Carbon Printing Functions

filterProc

A universal procedure pointer to the event handling function you want called while the Page Setup or Print dialog is displayed. In Mac OS X this function is not called.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSetModalFilterProc` is valid between the creation and release of a printing session. You must install your filter procedure before calling `PMSessionPageSetupDialogMain` or `PMSessionPrintDialogMain`.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.4.

Not available to 64-bit applications.

Declared In

`PMApplicationDeprecated.h`

Deprecated in Mac OS X v10.5

PMSessionBeginDocument

Begins a print job in which, by default, all drawing is to a QuickDraw graphics port. A printing status dialog informs the user of the job's progress. (Deprecated in Mac OS X v10.5. Use [PMSessionBeginCGDocument](#) (page 10) instead.)

```
OSStatus PMSessionBeginDocument (
    PMPrintSession printSession,
    PMPrintSettings printSettings,
    PMPageFormat pageFormat
);
```

Parameters

printSession

The printing session for the new print job.

printSettings

The print settings for the new print job.

pageFormat

The page format for the new print job.

Return Value

A result code. See Core Printing Result Codes.

Discussion

The function `PMSessionBeginDocument` starts a print job and should be called within your application's print loop. This function is similar to the function `PMSessionBeginDocumentNoDialog` except that the printing status dialog is displayed.

Deprecated Carbon Printing Functions

You must call this function between the creation and release of a printing session. See the function `PMCreateSession`. If you present a printing dialog before you call `PMSessionBeginDocument`, you should use the same `PMPrintSession` object you used to present the dialog.

Before you call `PMSessionBeginDocument`, you should call `PMSessionValidatePrintSettings` and `PMSessionValidatePageFormat` to make sure the specified print settings and page format objects are updated and valid. After you call `PMSessionBeginDocument`, if you call a function that changes the specified print settings or page format object, the change is ignored for the current print job.

This function must be called before its corresponding End function (`PMSessionEndDocument` (page 13)). If the function `PMSessionBeginDocument` function returns `noErr`, you must later call the End function, even if errors occur within the scope of the Begin and End functions.

The printing system automatically handles printing multiple copies. Your application does not need to perform any tasks other than specifying the number of copies in the printing session.

Special Considerations

In Mac OS X v10.4 and later, Apple recommends using the function `PMSessionBeginCGDocument` (page 10) instead of this function. QuickDraw is deprecated and your application should be using Quartz 2D for its rendering.

Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.5.

Not available to 64-bit applications.

Related Sample Code

`CarbonSketch`

Declared In

`PMApplicationDeprecated.h`

Document Revision History

This table describes the changes to *Carbon Printing Reference*.

Date	Notes
2007-07-24	New document that describes the C API used by Carbon applications to provide a user interface for printing.

REVISION HISTORY

Document Revision History

Index

D

DisposePMItemUPP **function** (Deprecated in Mac OS X v10.4) [29](#)
DisposePMPageSetupDialogInitUPP **function** (Deprecated in Mac OS X v10.4) [29](#)
DisposePMPrintDialogInitUPP **function** (Deprecated in Mac OS X v10.4) [30](#)
DisposePMSheetDoneUPP **function** [9](#)

I

InvokePMItemUPP **function** (Deprecated in Mac OS X v10.4) [30](#)
InvokePMPageSetupDialogInitUPP **function** (Deprecated in Mac OS X v10.4) [31](#)
InvokePMPrintDialogInitUPP **function** (Deprecated in Mac OS X v10.4) [31](#)
InvokePMSheetDoneUPP **function** [9](#)

K

kPMHideInlineItems **constant** [27](#)
kPMShowDefaultInlineItems **constant** [27](#)
kPMShowInlineCopies **constant** [27](#)
kPMShowInlineOrientation **constant** [28](#)
kPMShowInlinePageRange **constant** [27](#)
kPMShowInlinePageRangeWithSelection **constant** [27](#)
kPMShowInlinePaperSize **constant** [27](#)
kPMShowInlineScale **constant** [28](#)
kPMShowPageAttributesPDE **constant** [28](#)

N

NewPMItemUPP **function** (Deprecated in Mac OS X v10.4) [32](#)
NewPMPageSetupDialogInitUPP **function** (Deprecated in Mac OS X v10.4) [32](#)
NewPMPrintDialogInitUPP **function** (Deprecated in Mac OS X v10.4) [33](#)
NewPMSheetDoneUPP **function** [10](#)

P

PMBeginDocument **function** (Deprecated in Mac OS X v10.4) [33](#)
PMBeginPage **function** (Deprecated in Mac OS X v10.4) [34](#)
PMEndDocument **function** (Deprecated in Mac OS X v10.4) [34](#)
PMEndPage **function** (Deprecated in Mac OS X v10.4) [35](#)
PMGetDialogAccepted **function** (Deprecated in Mac OS X v10.4) [35](#)
PMGetDialogDone **function** (Deprecated in Mac OS X v10.4) [36](#)
PMGetDialogPtr **function** (Deprecated in Mac OS X v10.4) [36](#)
PMGetItemProc **function** (Deprecated in Mac OS X v10.4) [37](#)
PMGetModalFilterProc **function** (Deprecated in Mac OS X v10.4) [38](#)
PMItemProcPtr **callback** [21](#)
PMItemUPP **data type** [25](#)
PMPageSetupDialog **function** (Deprecated in Mac OS X v10.4) [38](#)
PMPageSetupDialogInit **function** (Deprecated in Mac OS X v10.4) [39](#)
PMPageSetupDialogInitProcPtr **callback** [22](#)
PMPageSetupDialogInitUPP **data type** [25](#)
PMPageSetupDialogMain **function** (Deprecated in Mac OS X v10.4) [40](#)
PMPrintDialog **function** (Deprecated in Mac OS X v10.4) [40](#)

PMPrintDialogInit function (Deprecated in Mac OS X v10.4) 41
 PMPrintDialogInitProcPtr callback 23
 PMPrintDialogInitUPP data type 26
 PMPrintDialogInitWithPageFormat function (Deprecated in Mac OS X v10.4) 41
 PMPrintDialogMain function (Deprecated in Mac OS X v10.4) 42
 PMSessionBeginCGDocument function 10
 PMSessionBeginDocument function (Deprecated in Mac OS X v10.5) 49
 PMSessionBeginPage function 11
 PMSessionDisablePrinterPresets function 12
 PMSessionEnablePrinterPresets function 13
 PMSessionEndDocument function 13
 PMSessionEndPage function 14
 PMSessionPageSetupDialog function 14
 PMSessionPageSetupDialogInit function (Deprecated in Mac OS X v10.4) 43
 PMSessionPageSetupDialogMain function (Deprecated in Mac OS X v10.4) 44
 PMSessionPrintDialog function 15
 PMSessionPrintDialogInit function (Deprecated in Mac OS X v10.4) 44
 PMSessionPrintDialogMain function (Deprecated in Mac OS X v10.4) 45
 PMSessionUseSheets function 16
 PMSetDialogAccepted function (Deprecated in Mac OS X v10.4) 46
 PMSetDialogDone function (Deprecated in Mac OS X v10.4) 47
 PMSetItemProc function (Deprecated in Mac OS X v10.4) 48
 PMSetModalFilterProc function (Deprecated in Mac OS X v10.4) 48
 PMSheetDoneProcPtr callback 24
 PMSheetDoneUPP data type 26
 PMShowPageSetupDialogAsSheet function 17
 PMShowPrintDialogWithOptions function 18
 PMShowPrintDialogWithOptionsAsSheet function 19
 Print Dialog Options 26