

# PaginationLab

by Samuel Streeper, NeXT Technical publications

## Overview

When you print a large View, the Application kit will dice your View up into page size rectangles automatically and place these pages on sheets of paper for you. However, it's quite likely that the page breaks that the kit selects are not where you might like them. This application demonstrates how you can influence the kit's pagination routines, and how you can take control of pagination altogether.

This app contains a big scrolling View, an instance of the appropriately named BigView class. A BigView has very obvious page breaks, and each page has one inch cells that simulate graphic objects that should not be split across pages, much as you wouldn't want to clip a line of text or a spreadsheet cell.

All the information about how printing should be formatted is kept in a PrintInfo object. The settings of the PrintInfo are set by the PageLayout panel, which has been extended to display additional fields in the PrintInfo and to allow you to set these fields and view the printed results.

## Program Organization

### How to build the nib files

This app contains two nib files. The first is for the main application with its menus and window with the scrolling BigView. The other only contains a panel with a View which is used as the accessory View for the customized PageLayout panel. The accessory View was given its own nib file so you can reuse the extended PageLayout panel simply by copying its object, interface, and nib files. Otherwise you would always have to rip it out of the main nib which goes against the grain of reusability.

There are only a few interesting connections in this application. In the main nib file, the Page Layout command is connected to the Worker object, which then runs the customized Page Layout panel rather than running the normal panel.

When the MyPageLayout panel is run for the first time, it loads the MyPageLayout nib file, which instantiates the view hierarchy it needs for the panel's accessory view, and sets several outlets in the nib files owner, which is the customized MyPageLayout instance. These outlets refer to the accessory view and the various controls within it.

When the app starts up, the main window is not visible. This is because for some reason I can't seem to pre-scroll the BigView using Interface Builder. I therefore scroll to the top of the BigView when the application is initialized in **appDidInit:**, and then I make the window visible.

## Major Classes in the Application

Worker	This class functions as NXApp's delegate, initializing stuff before PaginationLab runs. It also handles requests to run the pageLayout panel.
BigView	A very simple huge view with obvious places to break pages. Demonstrates hinting to the kit about suggested page breaks, and also shows how to take complete control over page breaks.
MyPageLayout	A customized page layout panel that allows you to inspect and set more of the interesting fields in a PrintInfo object.
MyPrintInfo	A PrintInfo object contains all the settings necessary to format the output of a document. This subclass adds one field that indicates how pagination should occur; whether the kit should use its normal algorithm, whether Views should help the kit, or whether Views should take control. The settings of the printInfo are inspected and set by the PageLayout panel, and during printing these settings are used by the kit and by a BigView to determine pagination settings.

## Topics Of Interest

Using this app you can readily see the differences between pages and sheets. A page is a conceptual unit of output, and one page may be printed on several physical sheets of paper, or several pages may be grouped on a single sheet.

## Other Files

PaginationLab.nib	The main nib file, the user-interface of the application.
MyPageLayout.nib	Contains the accessory view for the Page Layout panel. The page layout panel is thus fairly independent of the application.
PagIcon.tiff	PaginationLab's ugly icon
drawRect.psw	The PostScript language procedures used to create each "page" in the BigView.
PB.project, Makefile, PaginationLab.iconheader , PaginationLab_main.m	

Valid for 3.0