Table of Contents

Acumen Training

Electronic Mark-up and Review

I spend a lot of time writing student materials and sending them around to other people for review. I used to do this by Fed-Ex'ing printed copies to my proofreaders for them to mark up and then Fed-Ex back to me. This meant a pretty fair amount of money and at least two days travel time for each review cycle.

Nowadays, I email the PDF version of my document to readers and let them mark up the document within Acrobat. They send the document back to me with text ~~crossed out,~~ illustrations circled and nasty notes about writing quality. Sorry, Not good enough. Try again

This month, I thought I'd describe how they use Acrobat annotation tools to mark up my deathless prose.

Acrobat Annotations

The annotation mechanism in Acrobat started out as just a simple "sticky note," but expanded in Acrobat 4 into twelve variations that allow you to make a wide variety of marks and comments on your PDF pages.

You can view the annotations in a PDF file with either the Acrobat Reader or the full Adobe Acrobat. However, you must have the full Acrobat application to create these annotations.

Here we will briefly examine the types and uses of the annotations available to us in Acrobat 4.

Next page ->

Acumen Journal: The Acrobat User

The Original "Sticky Note"

You can still make annotations in their original "sticky note" form. It's the top-most annotation type in the default toolbar. Just click on the tool and click-and-drag a rectangle. You will be faced with a blank sticky note into which you can type your note.

When closed, the sticky note looks like the one in the left margin of this page: just a marker indicating a note is here. Double-clicking this marker (sure, go ahead) opens the note.

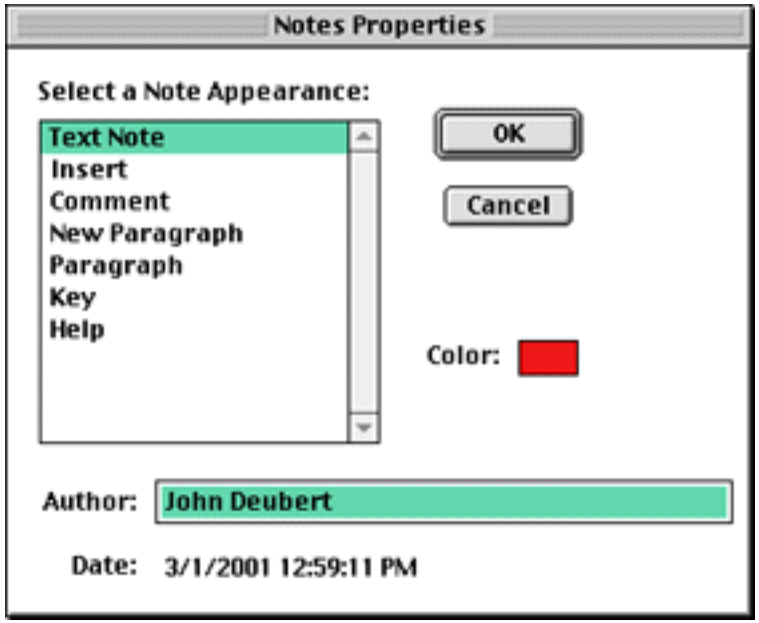
(By the way, you will not be able to change the annotations in this Acumen Journal PDF file; the file is password protected.)

Next page ->

Acumen Journal: Acrobat User

Sticky Note Preferences

If you click on a sticky note (or an annotation of any type) and select *Properties* from the Edit menu, you will gain access to a dialog box that lets you specify the properties of that annotation.



What you are offered as choices depends upon what kind of annotation you have selected. In the case of sticky notes, you may specify color, the name that appears in the title bar, and the icon that is used as a marker on the page.

(Note the markers in the margin at left. These are all available alternatives to the little picture of a sticky note.)

You should examine the properties available to you for all the other annotation types.

Next page ->

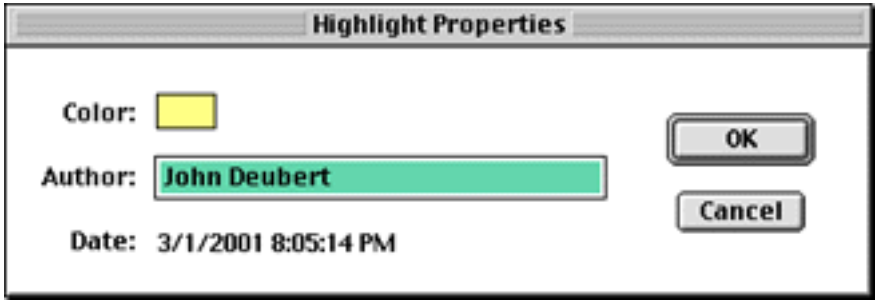
Text Annotations

Click and hold on the bottommost annotation tool and you will get a "fly-out" menu that allows you to select among the text annotation tools. These allow you to mark and annotate text.

You have three controls that allow you to highlight, ~~cross out,~~ or underline text on which you want to comment. If you double-click on any of the marked text, you can see the attached note. (Try it out.)

Text Annotation Properties

The properties you have available for text annotations allow you to specify only two things: the color of the highlighter and the name of the author.



I recommend setting the color of the highlight tool to a fairly pale color. It makes it much easier to read the highlighted text.

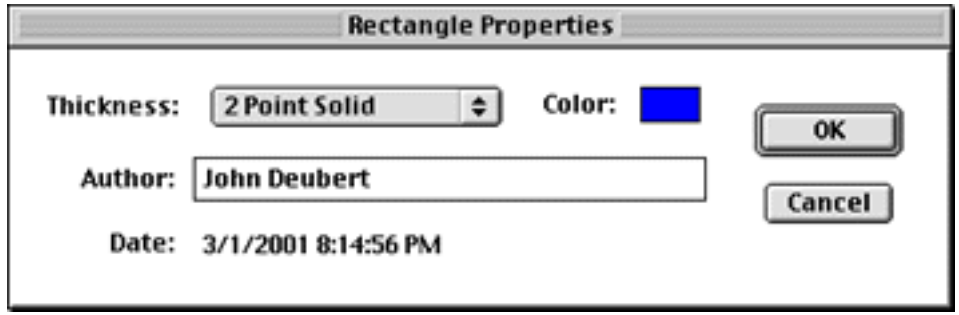
Next page ->

Drawing Tools

The middle set of annotation tools allow you to draw on your PDF page. The fly-out menu allows you to select among four tools: a pencil (for arbitrary drawings), a rectangle tool, an ellipse tool, and a line tool. (I've always been a little disappointed that there's no arrow tool.)

Properties

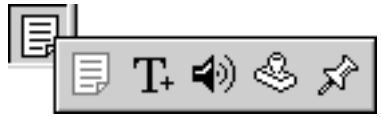
The Properties dialog box for these controls allow you to specify the color and line width of your annotation. You may also specify it be drawn with a dashed line. (It's one of your choices in the "Thickness" menu.)



Next page ->

Special Annotations

My favorite — though not my most-used — annotations are those in the topmost set, the "special" tools. These have no particular common theme to them, but there are some fun tools here.



The first of the tools is the time-honored sticky note, which we've discussed. The other four are interesting enough to spend some time discussing.

Text Notation

This tool is rather like the sticky note without a frame. The notation you make is just sitting on the page like any other text. The Properties dialog box allows you to specify a frame and colored background, if you wish.

Here's text without a frame and background.

And here's text with both.

Next page ->

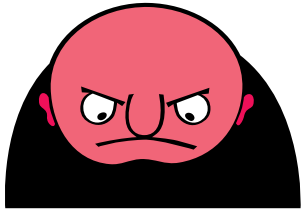
Sound Annotation

For sheer fun, it's hard to beat the sound annotation tool. (yes, I *do* have a simple social life; why do you ask?) Click on the page with this tool and the microphone on your computer goes live. Just click on the record button in the resulting dialog box and begin speaking into the microphone. Later, when the reader double-clicks on the sound icon (as at left), Acrobat plays back the sound.

Careful with this one! The sound is being embedded in your PDF file and can take up a *lot* of space very quickly. Make this a short recording.

Next page ->

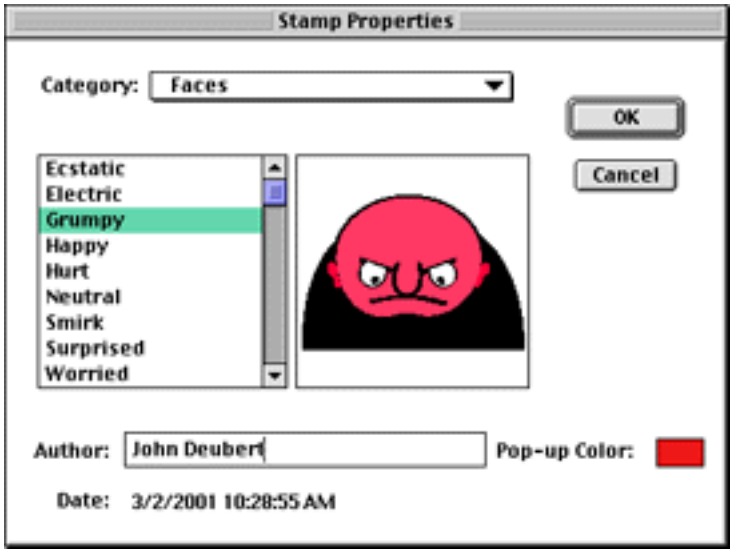
Rubber Stamp Tool



This one is fun, too. Select the tool, click and drag a rectangle, and Acrobat presents you with a dialog box that lets you pick from among a series of pictures you may place on the PDF page.

Double-clicking on the placed picture lets you enter a notation. The placed pictures are PDF, so you can freely resize them.

There's quite a range of pictures from which to choose. You can add pictures to the list, though the technique for doing so is a bit convoluted. (That may be a topic for a future article.)



Next page ->

File Attachment Tool

The final annotation type allows you to attach a file to your PDF document. This may be any kind of file: a spreadsheet, word processing document, Quark file, or anything else you wish. The file contents are actually embedded in the PDF file, so they travel together.

When the reader double-clicks the annotation icon, Acrobat opens whatever application is associated with the embedded file's type and lets them work with the document.

The Properties page allows you to specify an icon and tool tip text that should mark the attachment's position in the PDF file.

Next page ->

Returning the Document

Having marked up your document, you now need to send it back to the author. What, exactly, should you send them? You've got a couple of choices:

Send back the PDF file

You can simply resave the PDF file, together with your annotations, and email the file back to the author. The best thing about this is that the person receiving it doesn't have to do any work to read the annotations; he or she simply opens the PDF file and there they are.

On the other hand, it seems a waste to return the entire document, when they already have the original. The only thing you really want to send them is your annotations. Hence, the alternative:

Export the Annotations

The best thing to do, assuming the recipient is even familiar with Acrobat, is to export your annotations to a file (File->Export->Annotations) and send back only the annotations. This will be a much smaller file than the original-document-plus-annotations.

The person to whom you send the file needs only to import the annotations (File->Import->Annotations) into the PDF file and they are ready to be read.

Next page ->

Acumen Journal: Acrobat User

Reading Annotations

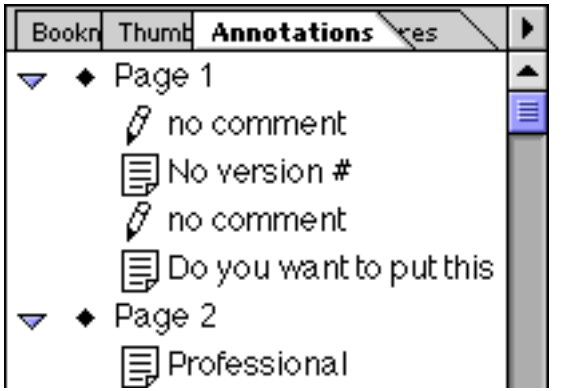
You have three ways of reading the annotations sent to you by someone else. (This assumes you have already imported them into your copy of the PDF file.)

Page through the file

You can just page through the document, looking for and reading the notes.

Use the Annotations Panel

If you open the Acrobat Navigation pane, one of your panels is "Annotations." This will list all of the notes in this PDF file. Double-clicking on any entry in this panel will take you to that note in the document.



The annotations in the document may not appear in the panel at first. In this case, you need to select *Rescan Document* in the fly-out menu attached to the Navigation pane.

Next page ->

Get a Summary

If you want a printed summary of all of the annotations in the document, you can select *Summary* from the Acrobat Tools menu. This creates a new PDF file with a list of all of the annotations in the document. This is excellent as a printed record of people's notes.

Conclusion

The annotation tools are well worth exploring. They are well thought out and well implemented and very easy to use. They vastly streamline the document review cycle.

Use them for a while and I think you'll find them quite addictive!

Next page ->

Return to menu

Concatenating PostScript Files

One of the recurring problems in PostScript is how to properly concatenate two or more PostScript files into a single file. In particular, how do you eliminate unwanted interactions between them? It's common to have the individual files work fine, but have the concatenated version generate a PostScript error.

This month, we'll take a look at two ways of doing this. In particular, we shall look at the PostScript you need to place before, after, and between the individual files to ensure they work well together.

Background

When concatenating PostScript files, there are four things you need to do in between them:

- Reclaim the memory used by the first file.
- Remove all key-value pairs defined by the first file.
- Remove all objects from the operand stack.
- Remove from the dictionary stack any dictionaries placed there by the first file.

You can carry out these activities "by hand," inserting PostScript code that carries out each of these steps. On the other hand, if you are sending the concatenated files to a LanguageLevel 2 or 3 printer, you can have all of these happen automatically upon invoking a single PostScript command.

Let's see how to do it.

next page ->

The Hard Way

We'll do the long version first. Here we're going to explicitly carry out our between-file tasks. This takes a few lines of PostScript code, but will work on any PostScript printer.

To do this, we're going to insert some PostScript code at the beginning and end of each individual PostScript file.

At the Beginning

At the start of the concatenated stream, we shall create a dictionary that we'll use in our between-file PostScript.

```
/$concatDict 2 dict def
```

The name of this dictionary is arbitrary, of course.

Note that this PostScript is inserted only once, at the beginning of your concatenated stream.

Next Page ->

Before Each File Before each file, we need to do the following:

- Do a *save,* allowing us to later recover the memory consumed by the PostScript file. As a side effect, this will also undo any key-value pair definitions.
- Note how many items are on the dictionary stack. We'll need this value afterwards to figure out how many additional dictionaries have been put on the stack by the concatenated PostScript code.

The PostScript code is as follows:

```
$concatDict /interFileState save put
$concatDict /dictCount countdictstack put
```

If you've not encountered it before, the *countdictstack* operator returns on the operand stack the number of dictionaries currently on the dictionary stack.

Next Page ->

After each file

At the end of each of the individual files making up your stream, we need to recover memory, undo key-value pairs, and restore the operand and dictionary stacks.

We do this as follows:

```
clear
countdictstack $concatDict /dictCount get sub { end } repeat
$concatDict /interFileState get restore
```

In the first line, *clear* removes everything from the operand stack.

The second line determines how many items are on the dictionary stack and subtracts that number from that the number of items on the stack before the PostScript file executed. We calculate the difference and then execute *end* that many times.

The final line, of course, just gets our save object out of *$concatDict* and does a *restore.* This recovers the used VM and destroys the key-value pairs created in the PostScript file.

Next Page ->

The Easy Way

The previous PostScript code isn't particularly difficult, but LanguageLevel 2 provides us a much easier way of isolating the individual PostScript files in our concatenated stream.

The PostScript *startjob* operator starts a new job in our PostScript stream. It causes the interpreter to do all of the usual end-of-job activities (including everything we need to do between files) and then continues executing the incoming PostScript code as a new job.

Doing things this way (which is what I recommend), all you need to do is place the following between each pair of files in the stream:

```
false 0 startjob pop
```

The boolean *false* indicates you do not want the next file executed behind the Server Loop; a *true* here would execute the file persistently.

The zero is a password; zero is the default password on all printers. If the password has been changed, you will get an *invalidaccess* error.

That is literally all you need to do if you are concatenating files for a LanguageLevel 2 or 3 printer. We get all the bookkeeping and clean up for free as a result of starting a new job.

Pretty slick.

Next Page ->

Acumen Journal: PostScript Tech

A Caveat

On thing to watch for: *startjob* will fail if there is an unbalanced *save* in the file. If the PostScript file does a *save,* at the beginning of a page, for example, and never executes the corresponding *restore,* you'll get a PostScript error. Fortunately, this is relatively rare.

Otherwise, the *startjob* technique is foolproof and definitely the way to concatenate files.

Return to Menu

Schedule of Classes, Mar 2001 - May 2001

Following are the dates and locations of Acumen Training's PostScript and Acrobat classes. Clicking on a class name below will take you to the Acumen training website to the description of that class.

The PostScript classes are taught in Orange County, California, near the Orange County airport, and in London at Adobe Systems' office near Heathrow.

PostScript Classes

PostScript Foundations	Orange Co., CA Apr 2 - 6	London, UK May 7 - 11
Advanced PostScript	Orange Co., CA Apr 23 - 26	
PostScript for Support Engineers	Orange Co., CA Mar 5 - 9	Orange Co., CA May 14 - 30

For more classes, see the Acumen Training website:
www.acumentraining.com/schedule.html

PostScript Course Fees PostScript classes cost $1,750 per student

Registration ->
Acrobat Classes ->

Acrobat Class Schedule

Acumen training teaches three users' classes in Adobe Acrobat (the links below will take you to the Acumen website's complete description):

Acrobat Essentials — This class teaches the student how to make perfect PDF files. It includes complete coverage of the meaning and proper settings of all of the Distiller Job Options.

Interactive Acrobat — Here we show you how to add bookmarks, links, buttons, sounds, movies, form fields, and other interactive features to an Acrobat file.

Troubleshooting with Enfocus' PitStop — This class shows the student how to use all of the capabilities of this popular editing and preflight software.

On-site Only — The Acrobat classes are taught only on corporate sites. If you have an interest in any of these classes for your group, please see the Acumen website regarding setting up an on-site class.

Back to PostScript Classes

Return to First Page

Contacting John Deubert at Acumen Training

For more information

For class descriptions or for any other information about Acumen's classes:

Web site: http://www.acumentraining.com

email: john@acumentraining.com

telephone: 949-248-1241

mail: 25142 Danalaurel, Dana Point, CA 92629

Registering for Classes

To register for an Acquired Knowledge class, contact us any of the following ways:

Register On-line: http://www.acumentraining.com/registration.html

email: registration@acumentraining.com

telephone: 949-248-1241

mail: 25142 Danalaurel, Dana Point, CA 92629

Back issues

Back issues of the Acumen Journal are available at the Acumen Training website: www.acumenjournal.com/AcumenJournal.html

Return to First Page

What's New at Acumen Training?

PSSE Class Renewed

The PostScript for Support Engineers class has been redone. The student notes have been freshened and brought up to date. PostScript sample code has been rethought and redone. If you sign up for this class, you will definitely like the new materials!

New Class: Technical PDF

A new class, provisionally named *Technical PDF,* will be available by late summer. This will be a three-day, hands-on class on PDF file structure, objects, etc. It will be aimed at anyone intending to generate, manipulate, or otherwise work with PDF files at a low level. I will be posting a proposed list of topics on the Acumen website (www.acumentraining.com/TechPDF.html) very soon. I would like your feedback and suggestions as to what else should be in such a class. (There's a link on the TechPDF page for doing this.)

Return to First Page

Journal Feedback

If you have any comments regarding the *AKI Journal,* please let us know. In particular, we are looking for three types of information:

Comments on usefulness. Does the Journal provide you with worthwhile information? Was it well written and understandable? Did you like it, hate it, or did it make you want to drown your sorrows? How could we make it better? Do you like the PDF format?

Suggestions for articles. Each Journal issue contains one article each on PostScript and Acrobat. What topics would you like us to address?

Questions and Answers. We are planning a Q&A section for future issues. Do you have any questions about Acrobat, PDF or PostScript?

Please send any comments, questions, or problems to

journal@acumentraining.com

Return to Menu