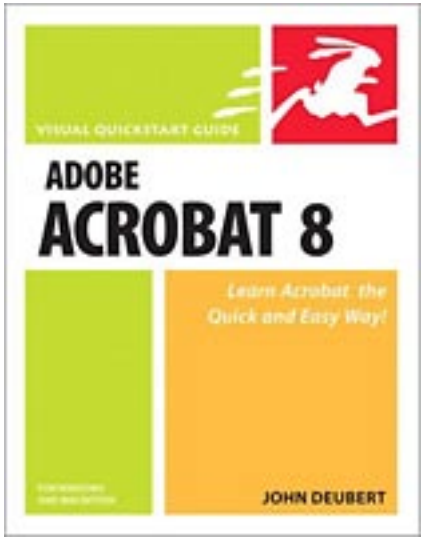
# Table of Contents

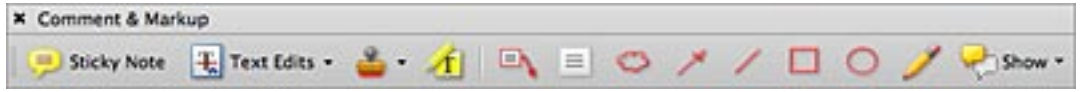




# Conducting an Email-Based Document Review


*You have bought this, haven't you?*

*Haven't you?*




One of my favorite features in the Acrobat universe is the commenting tools that have been improving and expanding since Acrobat 3. In Acrobat 8, the set of tools available for annotating a document is rich and extremely useful: highlight or cross out text; circle items of interest; draw arrows to problematic page contents; even stamp a picture or label onto the page.

In working on the books I've written with Peachpit Press (such as the Acrobat 8 Visual Quickstart Guide, of which you cannot have enough copies on your bookshelf), no paper was used until the book came off the press. Proofs were emailed to me as PDF files that I marked up using the Acrobat comment tools

Since Acrobat 6, Acrobat has included support for group reviews, where several people are reviewing a PDF document. In Acrobat 6 and 7, these tools entailed a lot of work to distribute and collect the comments from everyone and required some Acrobat expertise on the parts of the reviewers, as well.

Acrobat 8 has made it far easier to conduct and participate in such group reviews. In this article, we shall look at how to carry our such a group review of a PDF document.

Next Page ->

## Group Reviews

Acrobat supports two types of group reviews:

- *Shared Reviews,* in which the document to be reviewed resides on a server volume. Reviewers open the document on the server, add their comments, and then save the annotations back to the server. At any time, the originator of the document (let's call him or her the "author") can look at the comments that have been placed on the document.

  In the nature of things, a shared reviews requires all the reviewers have access to the server that holds the PDF document. This is probable if the review is being conducted within a single organization, but less so if the reviewers reside outside the author's company. This article will not discuss shared reviews.

- *Email Reviews,* in which the author emails the PDF file to a list of reviewers and receives the commented PDF files back as an email attachment. This works well with any reviewer who has email and is easy to initiate and to participate in.

  This is what we shall discuss in this article.

Next Page ->>

## Email-Based Reviews

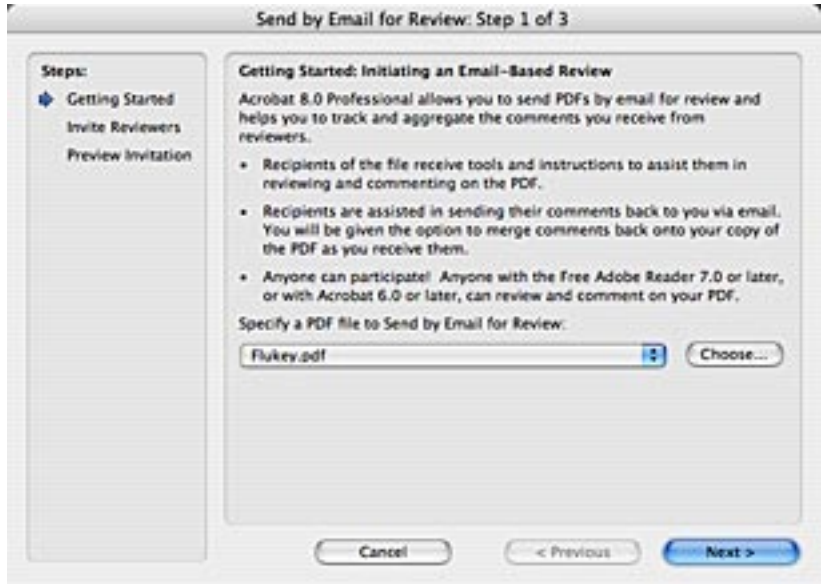
Email-based document reviews are very simple conceptually (and in fact). You email a copy of the PDF document to all of the reviewers. The reviewers comment on the document using the standard Acrobat annotation tools and then email the document back to you. You receive the commented documents and examine their comments.

Along the way, Acrobat does some work to streamline the process, keeping track of who your reviewers are and who among them still owe you reviews.

Let's see how it works.

## Initiating the Review

Initiate an email-based review of a document by selecting *Comments> Attach for Email Review.* Acrobat presents you with the first panel of the *Email Review* wizard, at right.




### *1. Choose a File*

This panel asks you for the file you want to send out for review. Click on the *Choose* button and select a file on your hard disk using the standard Pick-a-File dialog box.

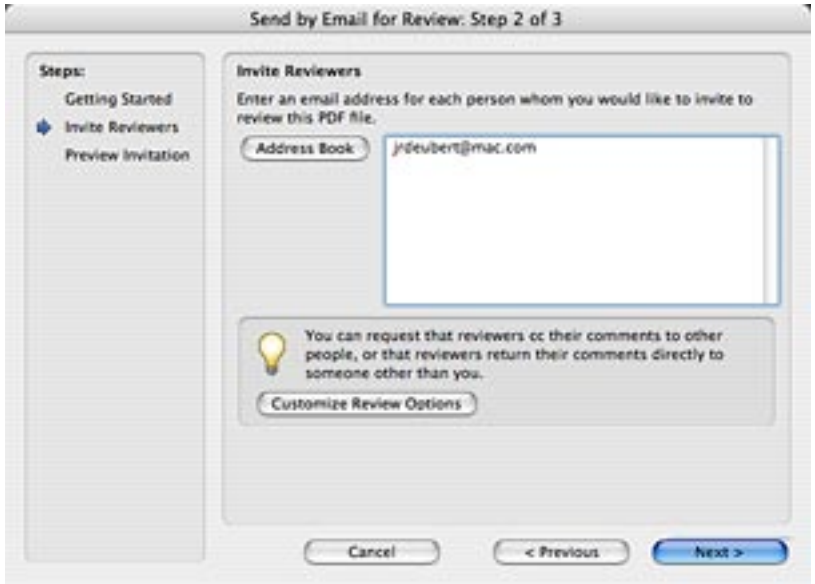
Next Page ->

As a convenience, you may select one of the currently-open files from the pop-up menu.

Click on the *Next* button when you are ready to proceed; Acrobat will present you with the second panel of the *Email Review* wizard.

## 2. Specify the Reviewers

Here you specify the email addresses to which you want to send review copies of your document.




Simply type the target email addresses into the text field. The addresses may be separated by spaces, commas, or newlines. You may also click on the *Address Book* button and select names from your system's address book.

The *Customize Review Options* button is important; it gives access to a very important setting. The button displays a dialog box (next page) that presents two options:

- *Alternative return email address*

  The upper part of the dialog box allows you to supply an email address to which the reviewed documents should be sent. This defaults to your own email address, but you may change it to anything you wish.

Next Page ->

Acumen Journal: Acrobat User 5

- *Allow review by Acrobat Reader*


  This is the important one. This checkbox tells Acrobat to make internal changes to the PDF file so that it may be reviewed by people who have only the free Adobe Reader application (rather than Acrobat Standard or Pro).

  You usually have no control over (or knowledge of) the software with which reviewers view your document; at least some of them may not own Adobe Acrobat and you will not want to insist that they spend hundreds of dollars purchasing the software just to have the privilege of reviewing your document. If you turn on this feature (which I strongly recommend), reviewers can annotate your document using the free Adobe Reader.

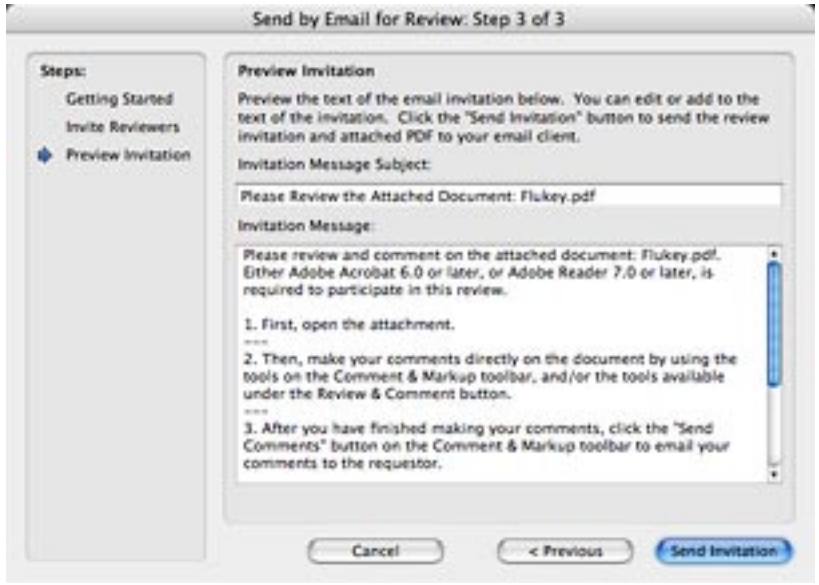
  There is a disadvantage to this, however; when you turn on the ability to comment on a PDF file in Adobe Reader, you also turn *off* the ability to do a variety of editing tasks in Adobe Acrobat. This means you won't be able to touch up text, rearrange pages, or make most other modifications to the document.

Click OK to return to the wizard then the *Next* button to go to the third and final wizard panel.

Next Page ->>

### 3. *Assemble the Message*

The third wizard panel lets you specify a subject and message body for the email that is sent to your reviewers. This panel is straightforward: type into the two fields the text you want for your email message's subject and body. I suggest you retain the default message body text, since it tells the reviewer how to proceed with the review. You can add your own text to the front of the default instructions.




When you are finished, click the *Send Invitation* button.

### 4. *Send the Message*

Acrobat will assemble an email message and do one of two things:

- On the Macintosh and in some Windows installations, Acrobat will open the new email message in your mail client. You can inspect the message and then send it on its way by clicking the client's *Send* button.


Next Page ->

- In many Windows installations, Acrobat will send the email message directly, without launching your mail client. I'm of two minds about this; it provides a much more seamless experience, but as someone who lives in a state of perpetual distraction, I like having a final look at the outgoing message.

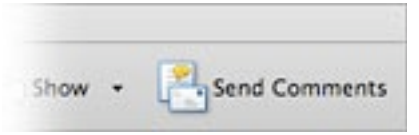
Either way, your document is now on its way to your reviewers.

## Reviewing the Document

The reviewer who receives an emailed document treats it like any other mail attachment: retrieve the PDF file and open it in Acrobat or Adobe Reader.

When you do this, your copy of Acrobat or Adobe Reader detects that the document is part of a review and does three things to make reviewing the document easier:

- It makes visible the Comment & Markup toolbar, making all of your commenting tools immediately available to you.
- It adds a *Send Comments* button to the end of the Comment & Markup toolbar, making it easy to return the annotated document to the sender: just click this button and Acrobat will create an email with the document attached.




Next Page ->

- It places brief instructions for how to review the file at the top of the document window, as at right.

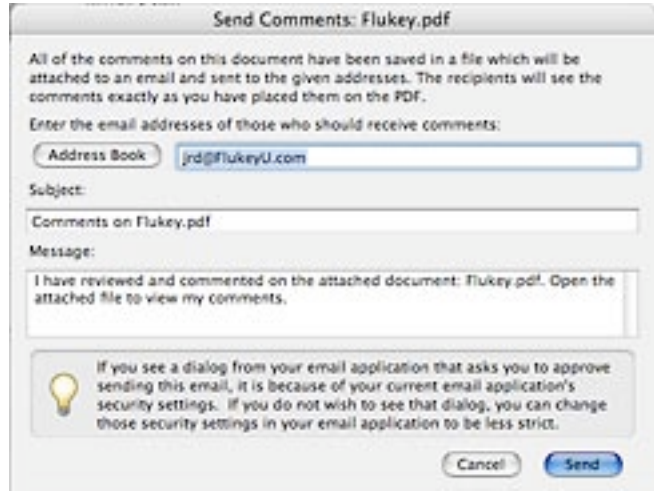

If the document was forwarded to you from another reviewer, Acrobat will also ask if you want to see the comments placed in the file by previous reviewers.


Actually reviewing the document is easy. Just add comments to the document using the Acrobat annotation tools, as usual.

When you have finished, click the *Send Comments* button in the Comment and Markup toolbar. Acrobat will attach the annotated document to a new message in your email client.

Next Page ->

Before sending your message, Acrobat will give you a chance to supply a subject and message body for the return email. It also let you change the email address to which the commented document should be sent. Usually, you will want to retain the default return address, which will the the address specified by the author.




When you click the *Send* button, Acrobat will either open a new, pre-addressed email message in your email client or (in Windows) send the commented document directly back to the author.

## Receiving Reviewed Documents

When you, the author, receive an annotated document back from a reviewer, you again treat this as you would any other email attachment. When you open the document in Acrobat, the software will inform you that this file is part of a review and asks if you want to see the comments attached to the document. The correct answer is "Yes."


Next Page ->

Acrobat will display the document, together with all the comments placed by the reviewer.

Easy.


## Review Tracker

Acrobat has made it simple to distribute a PDF file for review, review a document, and collect annotated documents from your reviewers. However, it provides one more feature that makes the whole review process easy to manage: the Review Tracker.

Acrobat does a lot of bookkeeping behind the scenes when you conduct a review. In particular, it maintains a list of your reviewers and keeps track of who has returned reviewed copies of your document. The Review Tracker lets you see this information.

When you select *Comments>Review Tracker,* Acrobat displays the Review Tracker dialog box, at right.


The hierarchical "tree" list in the left side of this dialog box lists all the documents you have sent for review to other people. The right-hand panel duplicates this list, with each entry being a link that opens the original PDF file.

Next Page ->


When you click on one of the review documents in the tree list, Acrobat displays in the right-hand pane the list of reviewers for that document and whether each reviewer has returned comments or not.

There are some additional, reasonably obvious links in the right-hand pane, including links to add new reviewers (takes you to the Initiate Email Review wizard) and to send an email to all the reviewers in the list.

In a lot of ways, the Review Tracker is the most useful part of the whole system. I can be remarkably slipshod in keeping track of who my reviewers are and whether I have gotten anything back from them; the Review Tracker does this bookkeeping for me.

I'm impressed with the improvements Acrobat 8 made to the document review process. I think Adobe has at last made it very easy to initiate, participate in, and keep track of group reviews.

Return to Main Menu

# PostScript Resources

Consider how fonts work in PostScript. Fonts may be stored in a variety of places on a PostScript device: in RAM, in ROM, on a hard disk, on a plug-in cartridge (remember those?), on a network volume, or in any other place the device manufacturer desires. As a PostScript programmer, you ask for fonts by name, without regard to where they are stored, using the *findfont* operator:

```
/Optima-Oblique findfont
```

The implementation of *findfont* on each PostScript device searches all the places fonts can be stored on that particular device.

PostScript Level 2 generalized the font mechanism in the form of *resources.* A PostScript resource is a piece of data (usually a dictionary, though it may be of any PostScript data type) that is stored and retrievable by two names:

- The name of the *Resource Category* to which the resource belongs. This indicates the purpose of the resource: is it a form, a font, a pattern, etc.
- The *Resource Name* of the particular resource you want within that category.

Thus, starting with Level 2, the Optima-Oblique font dictionary is stored as a resource whose category is *Font,* whose name is *Optima-Oblique,* and whose data is a font dictionary.

Next Page ->

## Resource Categories

PostScript defines set of default resource categories, each associated with a particular type of data, usually a dictionary. You can also define your own resource categories, though that is a topic for a different article.

PostScript resources fall into two broad classes: *regular resources* and *intrinsic resources.*

## Regular Resources

Regular resource categories, listed at right, are characterized by the fact that a PostScript program can create instances of these resources. Thus, Font is a regular resource category because a PostScript program can define a new font named, say, Kraken-Bold and save it as a Font resource.

**Regular Resource Categories**

| | | |
|---|---|---|
| Font | Encoding | Form |
| Pattern | ProcSet | ColorSpace |
| Halftone | Category | Generic |
| ColorRendering | FontSet | InkParams |
| TrapParams | IdiomSet | |

Next Page ->>

## Intrinsic Resources

By contrast, a PostScript program cannot make new instances of intrinsic resources, such as FontType or Filter. Examining the table, you can see that these resource categories represent built-in PostScript features that, for convenience, are managed using the resource mechanism.

**Implicit Resource Categories**

| | | |
|---|---|---|
| Filter | Emulator | ImageType |
| PatternType | HalftoneType | FontType |
| IODevice | FMapType | FormType |
| ColorSpaceFamily | ColorRenderingType | |
| FunctionType | ShadingType | |

By and large, you will have little reason to pay attention to implicit resources.

## Resource Operators

Since the resource mechanism is a generalization of the original PostScript font mechanism, you work with resources much the same way that you have forever worked with fonts. Where PostScript Level 1 had a *findfont* operator, PostScript Level 2 introduced *findresource*.

There are four PostScript resource-related operators of consequence.

### *findresource*

The *findresource* operator is analogous to the *findfont* operator; it retrieves the data associated with a resource.

```
/rsrcName  /rsrcCat  findresource  =>  data
```

Next Page ->>

The operator takes from the stack the name of the resource and the name of the category to which the resource belongs; *findresource* searches for the resource and returns its data (usually a dictionary) on the stack. Thus,

```
/Optima /Font findresource
```

is in every way identical to

```
/Optima findfont
```

In fact, in most PostScript implementations, the *findfont* operator calls *findresource*. This is why you will occasionally encounter an error whose offending command is *findresource* even though there are no calls to that operator anywhere in the PostScript file.

Generally, you will continue to use *findfont* when fetching a font; it's slightly more convenient and readable.

### *defineresource*

The *defineresource* operator is equivalent to *definefont;* you use it to make new resources within a category. It takes as arguments the name and data of the resource you want to make and the name of the category to which the new resource should belong.

```
/rsrcName   data   /rsrcCat   defineresource   =>   data
```

Thus,

```
/Helv << ..a font dict >>   /Font defineresource
```

is identical to

```
/Helv << ..a font dict >>   definefont
```

Next Page ->

Note that *defineresource* returns a copy of the resource data on the stack.

*resourcestatus* This operator lets you test for the availability of a resource. Its arguments are the name of the resource and its category:

```
/rsrcName  /rsrcCat  resourcestatus  =>  status size true
                                    =>  false
```

If the specified resource doesn't exist, *resourcestatus* returns a boolean *false;* if the resource is available, the operator returns two numbers and a *true*. The two numbers are:

*status* A code value that indicates how the resource got into VM. There are three values possible:

*0* The resource was placed in VM by *defineresource;* this indicates the resource was downloaded to the RIP.

*1* The resource was placed in VM by *findresource;* generally, this means the resource is built into the RIP (*e.g.,* a ROM font).

*2* The resource is not in VM, though it is available to the RIP. This indicates that the resource is stored on an external storage device, such as a hard disk.

I confess I don't know the circumstances under which this information is useful; I've never had occasion to pay attention to it.

Next Page ->>

*size* The amount of VM that would be occupied by this resource if it were loaded into VM. Do *not* depend on the accuracy of this number; it may be wildly inaccurate.

The only useful return value is the boolean, but that value is useful, indeed, since it tells you whether or not the resource is available to the RIP. As an obvious example, you can use this to determine whether a particular font is available:

```
/Optima /Font resourcestatus
{ pop pop (Yep, it's here!) = } { (Font not available) = }
ifelse
```

Less obviously, you can also use *resourcestatus* to determine whether the RIP has a TrueType renderer:

```
42 /FontType resourcestatus
{ pop pop (Yep, we do TrueType!) = } { (No TrueType here.) = }
ifelse
```

Remember that TrueType is implemented as a PostScript FontType 42 font.

Next Page ->>

*resourceforall*

Finally, *resourceforall* lets you iterate through all of the resources in a particular category.

```
(mask)  {proc}  (scratch) /rsrcCat  resourceforall
```

This operator steps through each resource in the specified category. For each resource, it places the resource name into the scratch string, pushes that string onto the stack, and then executes the procedure. The mask string is used to limit the search; within the string, the following characters have the following (common) meanings:

| | |
|---|---|
| * | Matches any run of characters. |
| ? | Matches any single character. |
| \ | Indicates the next character in the string is a literal; Thus "*" indicates that the resource names we want contain an asterisk character. |

Thus, to get a list of all the fonts available to the RIP, you can do the following:

```
(*)  { = }  128 string  /Font  resourceforall
```

If you wanted to see only members of the Helvetica family:

```
(Helv*)  { = }  128 string  /Font  resourceforall
```

By the way, this is a better method for getting a list of installed fonts than the traditional, Level 1 technique of dumping the contents of the *FontDirectory* dictionary:

```
FontDirectory { pop == } forall
```

Next Page ->

Acumen Journal: PostScript Tech 19

The Level 1 method lists only those fonts that reside in VM; fonts stored on the hard disk, for example, will not be reported. The *resourceforall* method lists all of the fonts available to the RIP, regardless of where they are stored.

## An Example

Let's create a Form resource.

The following discussion presumes you remember how PostScript forms work. If you have taken the Advanced PostScript or the Variable Data PostScript class, you should take a look at your student notes for a review. There is also a brief review of forms in the December 2002 Acumen Journal.

### *Forms Without Resources*

The usual way to use a form is as a key-value pair:

```
/SquareForm    <<
     /FormType 1
     /BBox [ 0 0 100 100 ]
     /Matrix [ 1 0 0 1 0 0 ]
     /PaintProc { 0 0 100 100 rectfill } bind
>> def
```

You use the form by handing the dictionary to the *execform* operator.

```
SquareForm execform
```

Next Page ->

Acumen Journal: PostScript Tech 20

### *Forms as Resources*

To use the form as a Form resource, we would use the form dictionary as the data in a call to *defineresource*.

```
/SquareForm    <<
     /FormType 1
     /BBox [ 0 0 100 100 ]
     /Matrix [ 1 0 0 1 0 0 ]
     /PaintProc { 0 0 100 100 rectfill } bind
>> /Form defineresource
```

Now we retrieve the form dictionary using the *findresource* operator.

```
/SquareForm  /Form  findresource  execform
```

### *So What's it Good For?*

You might say (I can hear you say it!) that the resource version is not obviously an improvement over the original. We are going to a bit of trouble saving and retrieving the form as a resource, to no obvious benefit.

This is true. Resources are not terribly interesting unless the RIP has a disk or other external storage available to it. In that case, resources become extremely useful, because you can save the resource definition (our call to *defineresource)* on the RIP's disk, so that all future PostScript programs can refer to the Form resource without having to define it.

Next Page ->>

At that point, the Form resource has much the same status as built-in fonts, which may be used without having to embed their definitions in the PostScript code. This is a very important technique for people doing such things as variable data printing with PostScript.

However, we're out of space this month.

We'll continue the discussion next time.

Return to Main Menu

# Schedule of Classes, February-April 2007

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

These classes are taught in Orange County, California and on corporate sites world-wide. See the Acumen Training web site for more information.

## Technical Classes

| | | | |
|---|---|---|---|
| **PDF File Content and Structure 1** | | Mar 5–8 | |
| **PDF File Content and Structure 2** | Feb 5–8 | | |
| **PostScript Foundations** | | Mar 26–30 | |
| **Variable Data PostScript** | Feb 12–16 | | |
| **Advanced PostScript** | | | Apr 9–12 |
| **Troubleshooting PostScript** | | Mar 19–21 | |

*Course Fee* The PostScript and PDF classes cost $2,000 per student, except for *Troubleshooting PostScript,* which is $1,500 per student.

Registration Info

# Acrobat Class Schedule

I am planning an *Acrobat for the Enterprise* class in 2007.

Watch for it!

Return to Main Menu

# Contacting John Deubert at Acumen Training

## For more information

For class descriptions, on-site arrangements or any other information about Acumen's classes:

**Web site:** http://www.acumentraining.com **email:** john@acumentraining.com

**telephone:** 949-248-1241

**mail:** 24996 Danamaple, Dana Point, CA 92629

## Registering for Classes

To register for an Acumen Training class, contact John any of the following ways:

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

**email:** registration@acumentraining.com

**telephone:** 949-248-1241

**mail:** 24996 Danamaple, Dana Point, CA 92629

## Back issues

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

Return to First Page

# What's New at Acumen Training?

## *Troubleshooting PostScript*

Acumen Training is introducing a new course, *Troubleshooting PostScript,* a three-day, hands-on course in diagnosing troublesome PostScript files. This is not a full programming course, but a streamlined, practical course in determining what is wrong with a PostScript file and what to do about it. The first class is scheduled for March 19–21. As always, you may also arrange an on-site class.

### *A Course for Field Support Engineers*

The class was written with printer field support engineers in mind, although it is useful for anyone who needs to handle broken PostScript files without becoming a PostScript programmer. Students are encouraged to bring their own samples of bad PostScript for examination in class.

### *Topics*

The class includes an introduction to PostScript programming, in-depth discussions of mechanisms that contribute to problems, PostScript operators that can be used to explore an aberrant PostScript file, techniques for diagnosing problems, and examination of common PostScript errors. We examine samples of driver output and practice troubleshooting problematic PostScript.

More information and a detailed topic list is on the Acumen Training website.

Return to First Page

# Journal Feedback

If you have any comments regarding the *Acumen Journal,* please let me know. In particular, I am looking for three types of information:

**Comments on usefulness.** Does the Journal provide you with worthwhile information? Was it well written and understandable? Do you like it, hate it? Did it hurt?

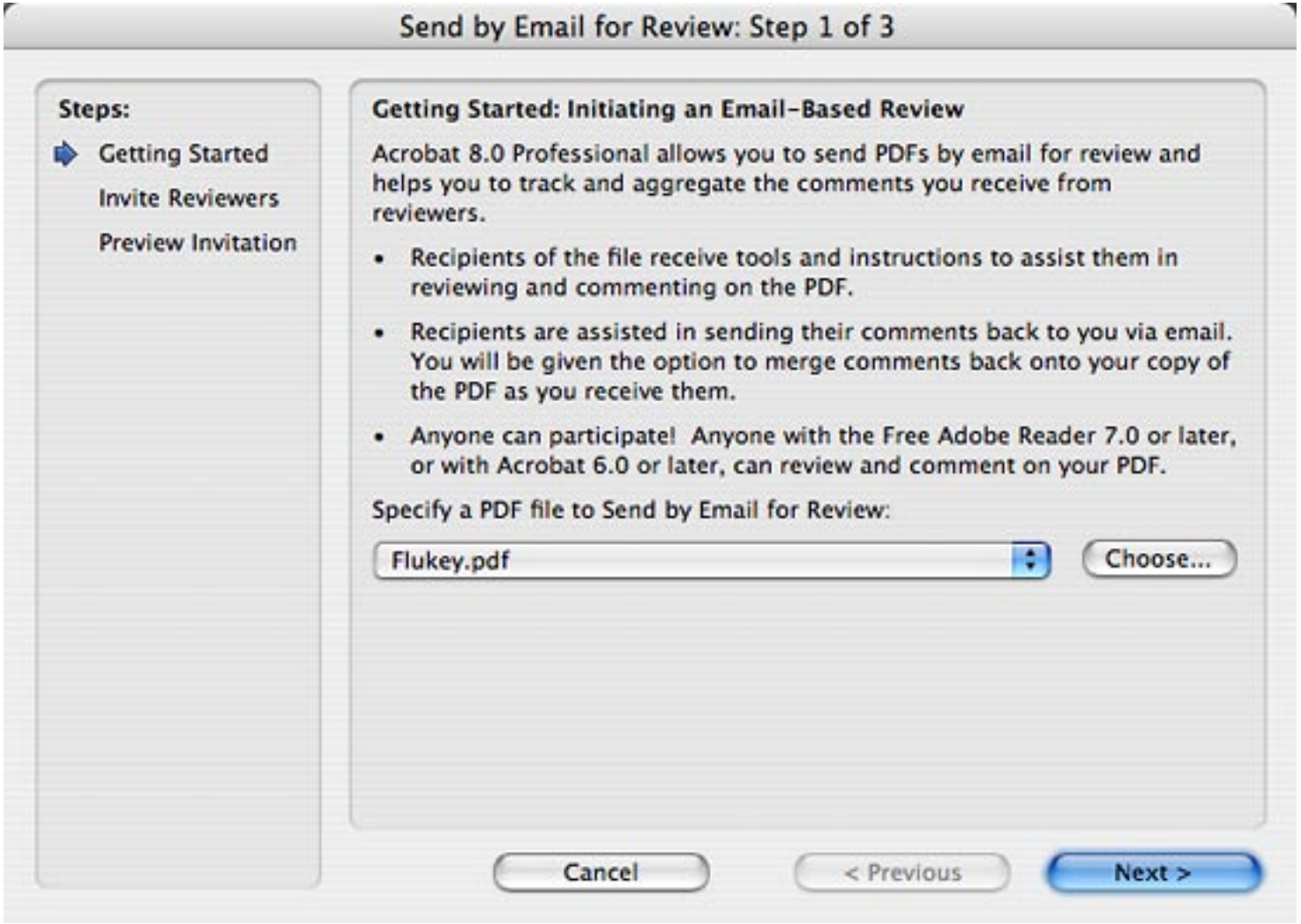
**Suggestions for articles.** Each Journal issue contains one article each on PostScript and Acrobat. What topics would you like me to write about?

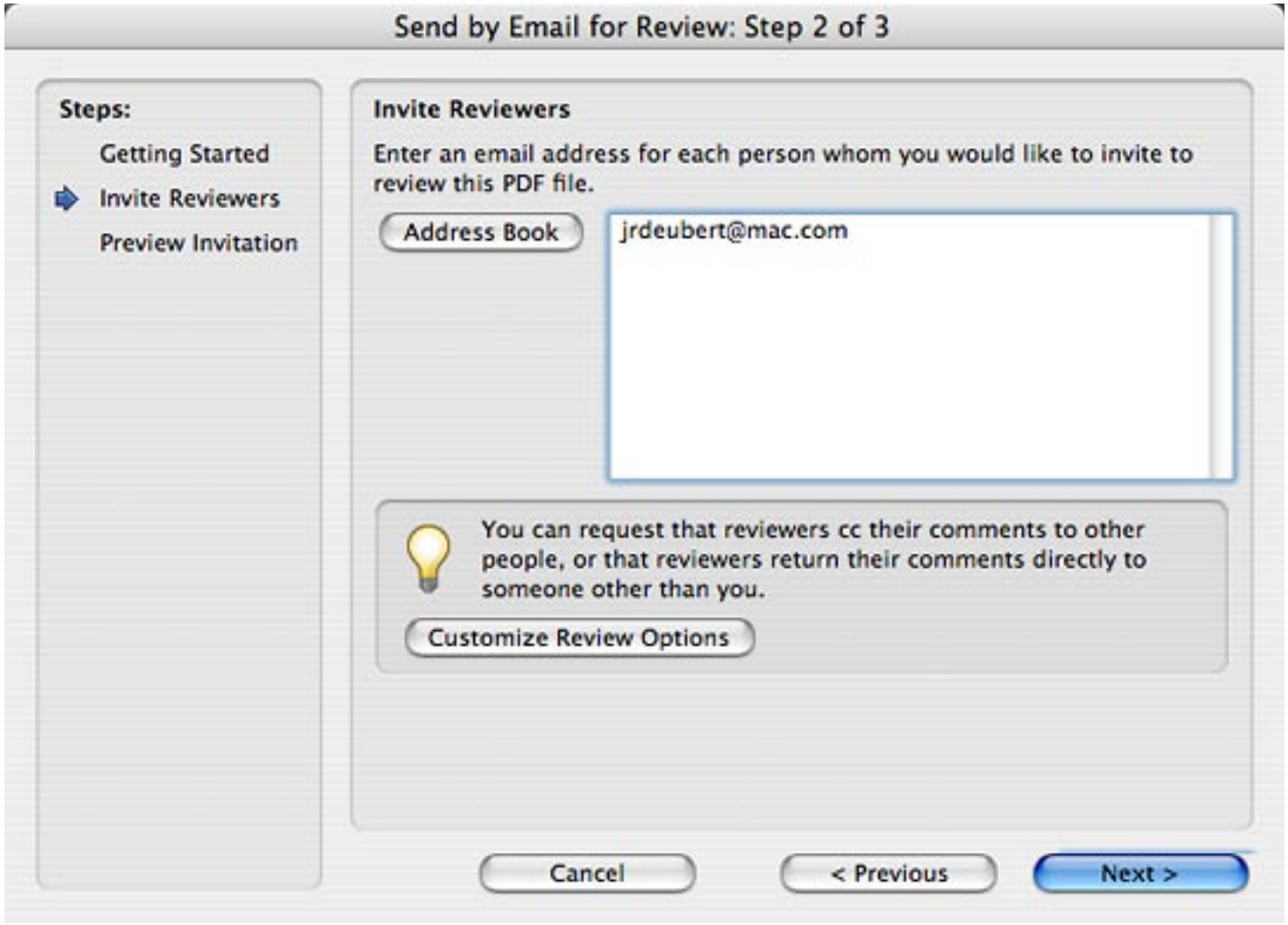
**Questions and Answers.** Do you have any questions about Acrobat, PDF, or PostScript? Feel free to email me about. I'll answer your question if I can. (If enough people ask the same question, I can turn it into a Journal article.)

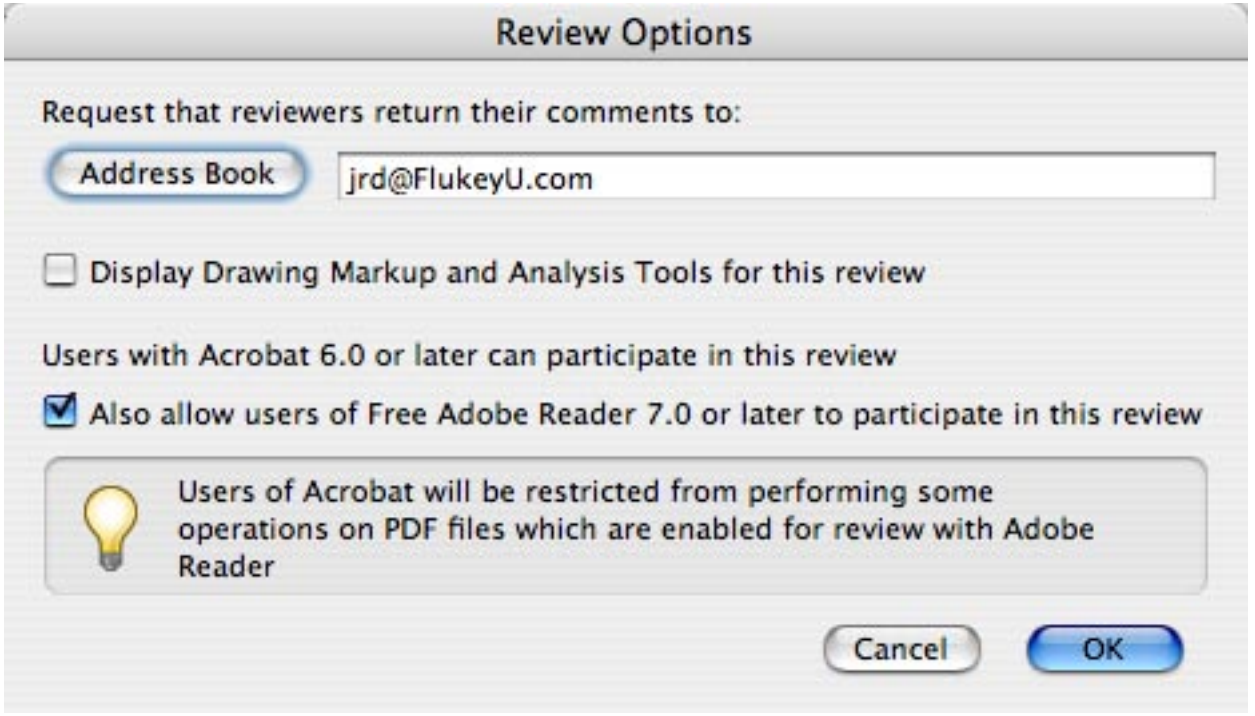
Please send any comments, questions, or problems to:

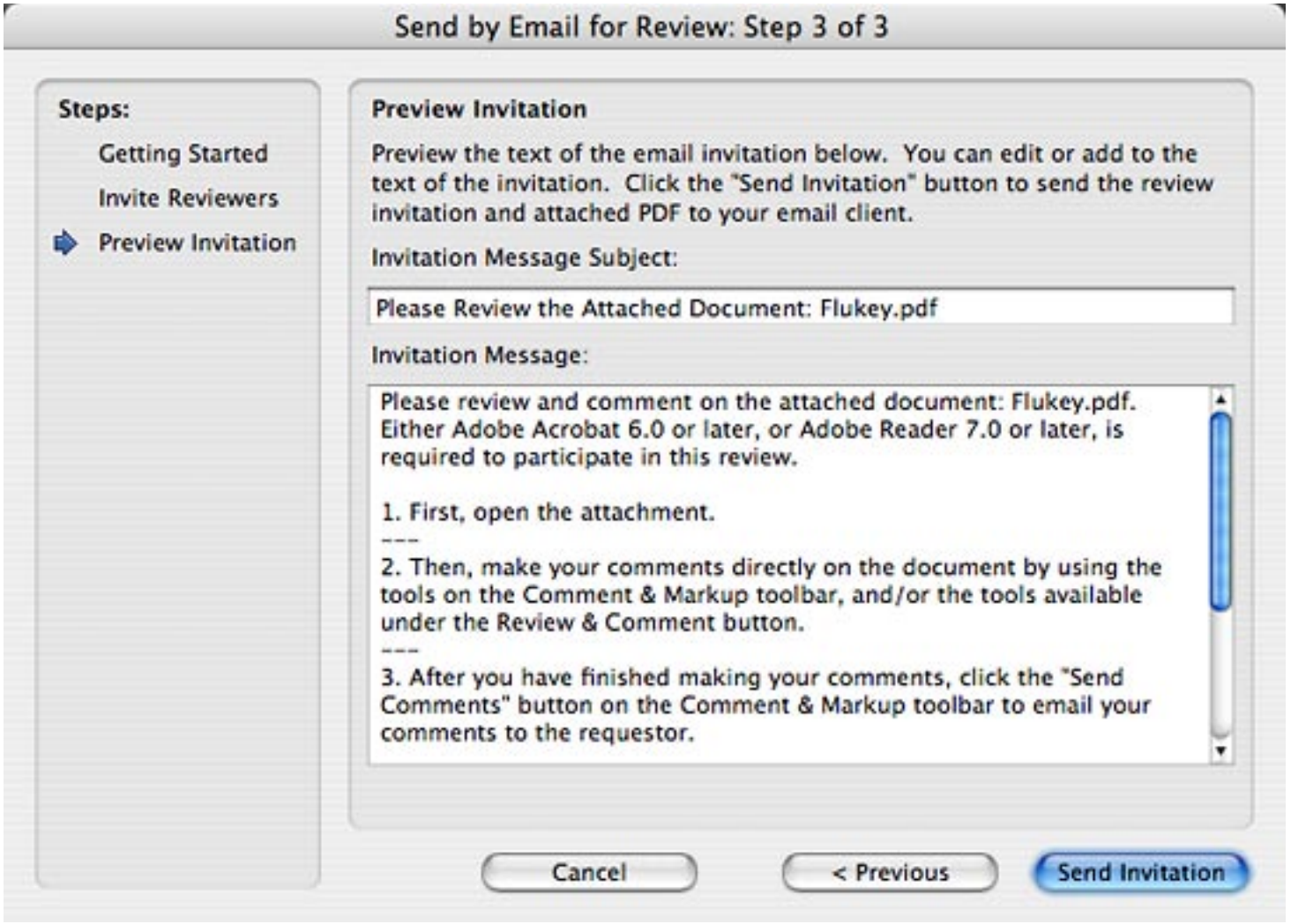
journal@acumentraining.com

Return to Menu


Send by Email for Review: Step 1 of 3
Steps:
Getting Started
Invite Reviewers
Preview Invitation
Getting Started: Initiating an Email-Based Review
Acrobat 8.0 Professional allows you to send PDFs by email for review and helps you to track and aggregate the comments you receive from reviewers.
Recipients of the file receive tools and instructions to assist them in reviewing and commenting on the PDF.
Recipients are assisted in sending their comments back to you via email. You will be given the option to merge comments back onto your copy of the PDF as you receive them.
Anyone can participate! Anyone with the Free Adobe Reader 7.0 or later, or with Acrobat 6.0 or later, can review and comment on your PDF.
Specify a PDF file to Send by Email for Review:
Flukey.pdf
Choose...
Cancel
< Previous
Next >


Send by Email for Review: Step 2 of 3
Steps:
Getting Started
Invite Reviewers
Preview Invitation
Invite Reviewers
Enter an email address for each person whom you would like to invite to review this PDF file.
Address Book
jrdeubert@mac.com
You can request that reviewers cc their comments to other people, or that reviewers return their comments directly to someone other than you.
Customize Review Options
Cancel
< Previous
Next >


Review Options
Request that reviewers return their comments to:
Address Book
jrd@FlukeyU.com
Display Drawing Markup and Analysis Tools for this review
Users with Acrobat 6.0 or later can participate in this review
Also allow users of Free Adobe Reader 7.0 or later to participate in this review
Users of Acrobat will be restricted from performing some operations on PDF files which are enabled for review with Adobe Reader
Cancel
OK


Send by Email for Review: Step 3 of 3
Steps:
Getting Started
Invite Reviewers
Preview Invitation
Preview Invitation
Preview the text of the email invitation below. You can edit or add to the text of the invitation. Click the "Send Invitation" button to send the review invitation and attached PDF to your email client.
Invitation Message Subject:
Please Review the Attached Document: Flukey.pdf
Invitation Message:
Please review and comment on the attached document: Flukey.pdf. Either Adobe Acrobat 6.0 or later, or Adobe Reader 7.0 or later, is required to participate in this review.
1. First, open the attachment.
---
2. Then, make your comments directly on the document by using the tools on the Comment & Markup toolbar, and/or the tools available under the Review & Comment button.
---
3. After you have finished making your comments, click the "Send Comments" button on the Comment & Markup toolbar to email your comments to the requestor.
Cancel
< Previous
Send Invitation

# Email with Attached PDF Document

Please Review the Attached Document: Flukey.pdf

Send | Chat | Attach | Address | Fonts | Colors | Save As Draft

To: A_B_See@CommaAds.com Doh@Raymi.com

Subject: Please Review the Attached Document: Flukey.pdf

Account: John Deubert <john@acumentraining.com> Signature: None


Flukey.pdf
152%
Find
Comments can be added to this PDF using the tools on the Comment & Markup toolbar.
When finished adding comments to Flukey.pdf, click "Send Comments" to submit your comments to the requestor.
How To...
Mother