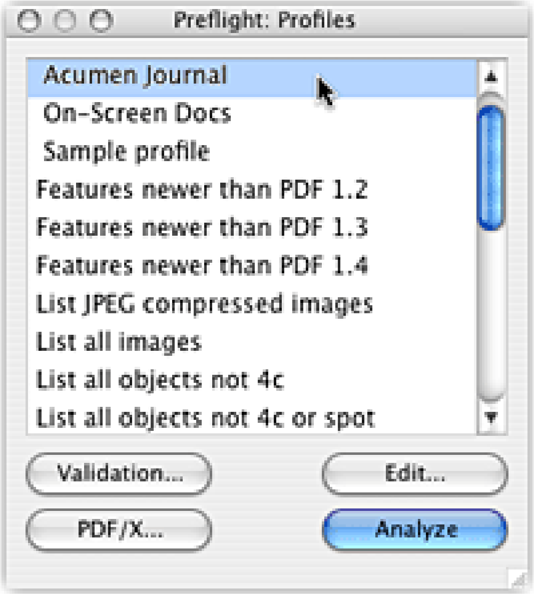
*John Deubert's* **Acumen Journal, March 2004**

# Table of Contents

# Acrobat 6 Preflighting, Part 2: Creating Preflight Profiles

In the previous *Acumen Journal,* we saw how to use Acrobat 6's Preflight feature to examine our PDF files, looking for missing fonts, insufficient resolution, and other things we consider a problem.


Preflight: Profiles

Acumen Journal
On-Screen Docs
Sample profile
Features newer than PDF 1.2
Features newer than PDF 1.3
Features newer than PDF 1.4
List JPEG compressed images
List all images
List all objects not 4c
List all objects not 4c or spot

## Preflight Profiles

From last month, we know that the first step in conducting a preflight scan of a PDF file is to select a Preflight Profile. This profile specifies exactly what we want Acrobat to look for: missing fonts, RGB colors, etc. Last month, we selected a profile from among those that ship with Acrobat.

The Preflight Profile we create in this article is available on the Acumen Training Resources page. Look for the file "PreflightAJ.zip."

The zip file also contains instructions for importing the profile into your copy of Acrobat.

However, the Adobe-supplied profiles will probably not be exactly what you need for your documents. In the long run—in the short run, for that matter—you will want to create your own preflight profiles so that the Acrobar preflight mechanism will look for exactly those things that you consider to be problems in your PDF files.

This month, we shall look at how to make a preflight profile. By way of example, we shall create a profile appropriate for use with the *Acumen Journal.*

Next Page ->

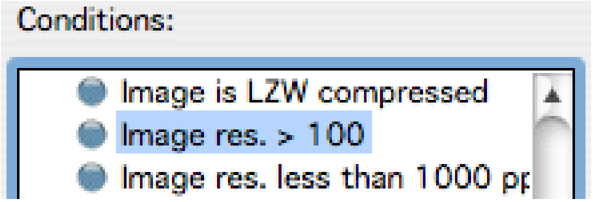
# Creating a Profile

## Overview

A preflight profile is built up out of two levels of more-primitive items: *conditions,* that are grouped together into *rules,* that are the components of a preflight profile.

### *Conditions*

A “condition” describes a single circumstance that constitutes a problem for your file; think of the term as short for “error condition.” This would be something like “Image resolution greater than 100 dpi” or “An object’s color is CMYK.”



Conditions:

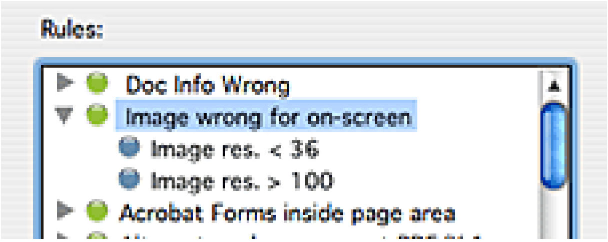
- Image is LZW compressed
- Image res. > 100
- Image res. less than 1000 p

The first step in creating a profile is to define all of the error conditions you want the preflight scan to look for.

### *Rules*

A “rule” is a collection of conditions that all apply to a broader problem. This is mostly an organizational convenience so that you can easily use a particular set of conditions in several scripts.

For example, for our *Acumen Journal* script, we shall make a rule named “Image is wrong for on-screen.” This rule will be made up of two conditions: “Image resolution > 100 dpi” and “Image resolution < 36 dpi.”



Rules:

- Doc Info Wrong
- Image wrong for on-screen
  - Image res. < 36
  - Image res. > 100
- Acrobat Forms inside page area

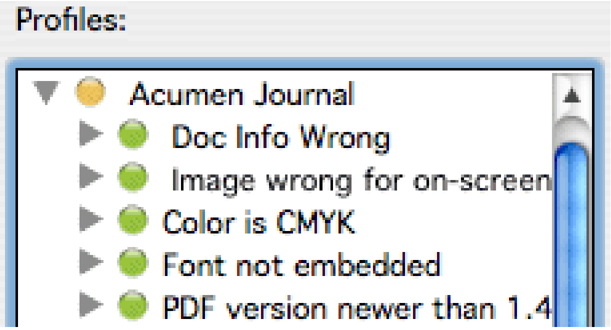
Next Page ->

## *Profiles*

Finally, our rules are grouped into a preflight profile that we actually use in our preflight session.

For the *Acumen Journal,* I need a profile that checks for the following:

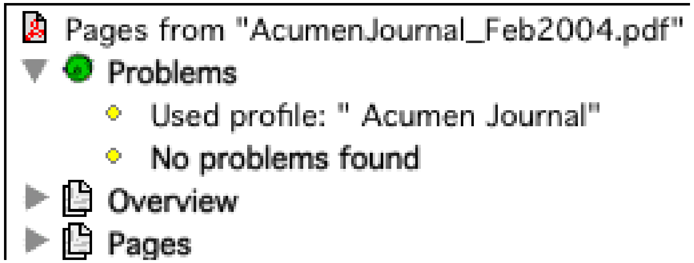
- Is any of the Document Info data missing?
- Are the images inappropriate for an on-screen document?
- Are there any CMYK colors in the document?
- Are any fonts not embedded?
- Is the PDF version required by the document greater than 1.4?



Profiles:

- Acumen Journal
  - Doc Info Wrong
  - Image wrong for on-screen
  - Color is CMYK
  - Font not embedded
  - PDF version newer than 1.4

If any of these questions come back positive, then the preflight report will tell me the file failed and will show me the details. Otherwise, it will tell me there were no problems, as at right.



Pages from "AcumenJournal_Feb2004.pdf"

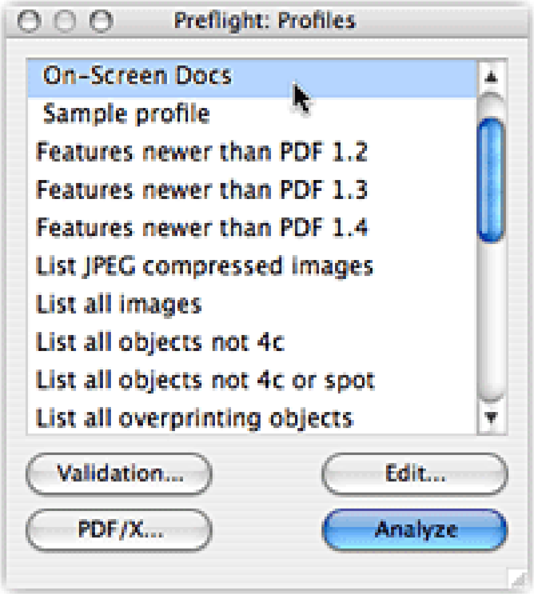
- Problems
  - Used profile: " Acumen Journal"
  - No problems found
- Overview
- Pages

In this article, we shall look in detail at making the *Acumen Journal* profile and adding to it the *Image is wrong for on-screen* rule.

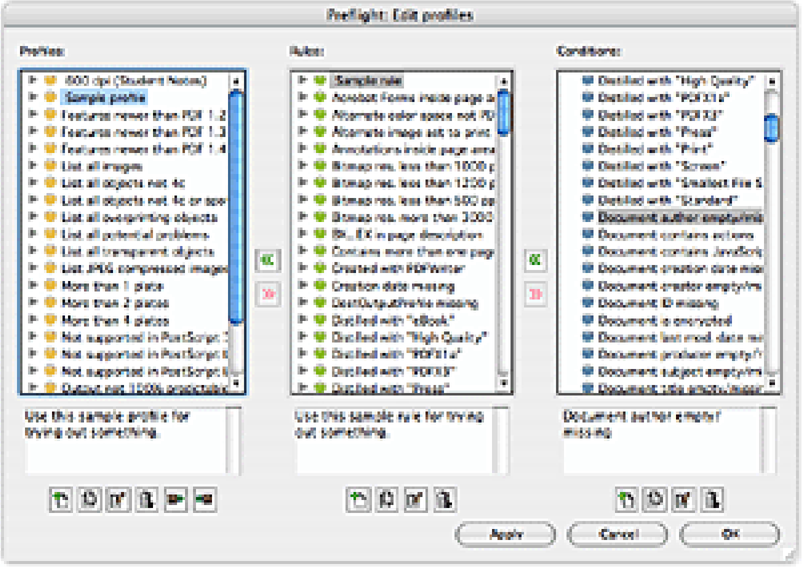
Next Page ->

## The *Edit Profiles* Dialog Box

To make a preflight profile, start by selecting *Document>Preflight* in Acrobat 6. You will be presented with the *Preflight Profiles* dialog box (right). This dialog box, as we saw last month, lets you run a preflight session with an existing profile.


Preflight: Profiles
On-Screen Docs
Sample profile
Features newer than PDF 1.2
Features newer than PDF 1.3
Features newer than PDF 1.4
List JPEG compressed images
List all images
List all objects not 4c
List all objects not 4c or spot
List all overprinting objects
Validation...
Edit...
PDF/X...
Analyze

If we click on the *Edit* button, Acrobat presents us with the *Edit Profiles* dialog box, which is where we create a new profile.


Preflight: Edit profiles
Profiles:
600 dpi (Student Notes)
Sample profile
Features newer than PDF 1.2
Features newer than PDF 1.3
Features newer than PDF 1.4
List all images
List all objects not 4c
List all objects not 4c or spot
List all overprinting objects
List all potential problems
List all transparent objects
List JPEG compressed images
More than 1 plate
More than 2 plates
More than 4 plates
Not supported in PostScript
Not supported in PostScript
Not supported in PostScript
Output not 100% predictable
Rules:
Sample rule
Acrobat Forms inside page a
Alternate color space not PD
Alternate image set to print
Annotations inside page area
Bitmap res. less than 1000 p
Bitmap res. less than 1200 p
Bitmap res. less than 500 pp
Bitmap res. more than 3000
BI...EI in page description
Contains more than one page
Created with PDFWriter
Creation date missing
DestOutputProfile missing
Distilled with "eBook"
Distilled with "High Quality"
Distilled with "PDF/X1a"
Distilled with "PDF/X3"
Distilled with "Press"
Conditions:
Distilled with "High Quality"
Distilled with "PDF/X1a"
Distilled with "PDF/X3"
Distilled with "Press"
Distilled with "Print"
Distilled with "Screen"
Distilled with "Smallest File S
Distilled with "Standard"
Document author empty/miss
Document contains actions
Document contains JavaScrip
Document creation date miss
Document creator empty/mi
Document ID missing
Document is encrypted
Document last mod. date mi
Document producer empty/
Document subject empty/mi
Document title empty/missin
Use this sample profile for trying out something.
Use this sample rule for trying out something.
Document author empty/ missing
Apply
Cancel
OK

This dialog box has three scrolling lists, displaying the existing conditions, rules, and profiles.

We shall be working our way from right to left to create our *Acumen Journal* preflight profile.

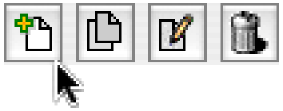
Next Page ->

## Specifying a Condition

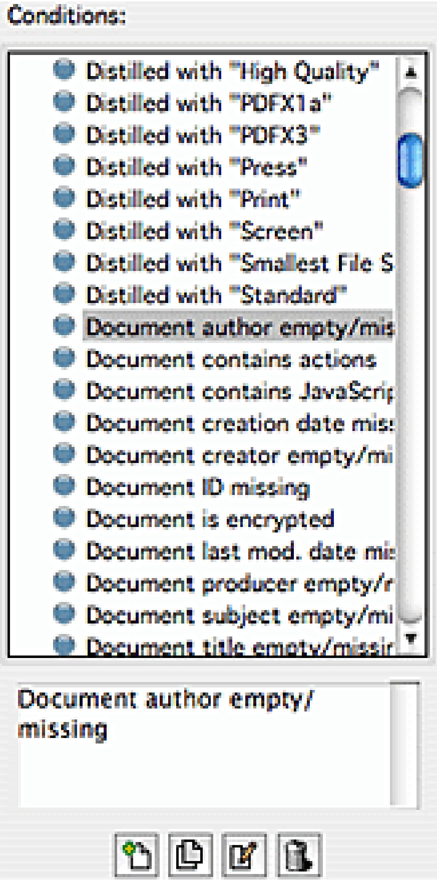
Let’s create one of the conditions needed by our *Images are wrong for on-screen* rule. Specifically, let’s create a condition that tells the preflight scanner to look for images with resolutions above 100 dpi. (Since the *Journal* is intended to be read on-screen, there’s no point to images with resolutions significantly greater than 72 dpi.)

The rightmost list in the *Edit Profiles* dialog box displays all of the conditions known to Acrobat. At the bottom of this list are four buttons that allow you to create, duplicate, edit, and delete conditions in the list.

We are interested in the left-most, “New Condition,” button; clicking on this yields the *Edit Condition* dialog box (next page).



Next Page ->



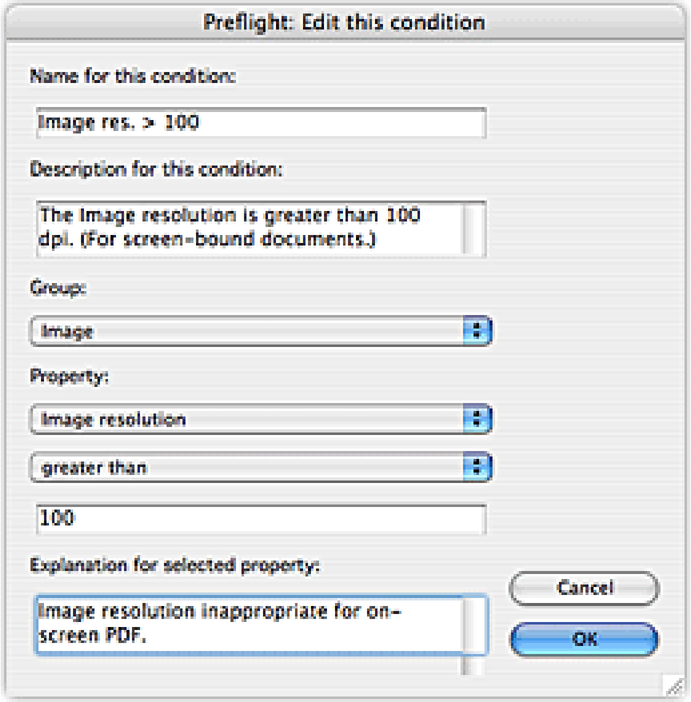
Conditions:

- Distilled with "High Quality"
- Distilled with "PDFX1a"
- Distilled with "PDFX3"
- Distilled with "Press"
- Distilled with "Print"
- Distilled with "Screen"
- Distilled with "Smallest File S
- Distilled with "Standard"
- Document author empty/mis
- Document contains actions
- Document contains JavaScrip
- Document creation date mis:
- Document creator empty/mi
- Document ID missing
- Document is encrypted
- Document last mod. date mi:
- Document producer empty/r
- Document subject empty/mi
- Document title empty/missir

Document author empty/
missing

## *Edit Condition* Dialog Box

The *Edit Condition* dialog box collects the following information about our new condition:



Preflight: Edit this condition

Name for this condition:

Image res. > 100

Description for this condition:

The Image resolution is greater than 100 dpi. (For screen-bound documents.)

Group:

Image

Property:

Image resolution

greater than

100

Explanation for selected property:

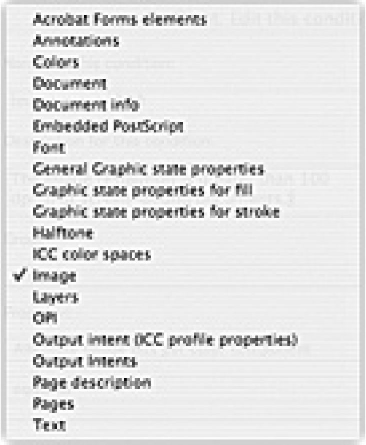
Image resolution inappropriate for on-screen PDF.

*Name for this Condition* This is the name that will appear in Acrobat’s listof conditions. We shall type “Image res. > 100.”

*Description* This is the text that will appear in the help text pane below the list of conditions on the Edit Profile dialog box.

*Group* This pop-up menu specifies the broad category to which the new condition belongs. In our case, we shall select *image.*

The category we select determines the appearance and nature of the remaining controls in this dialog box.

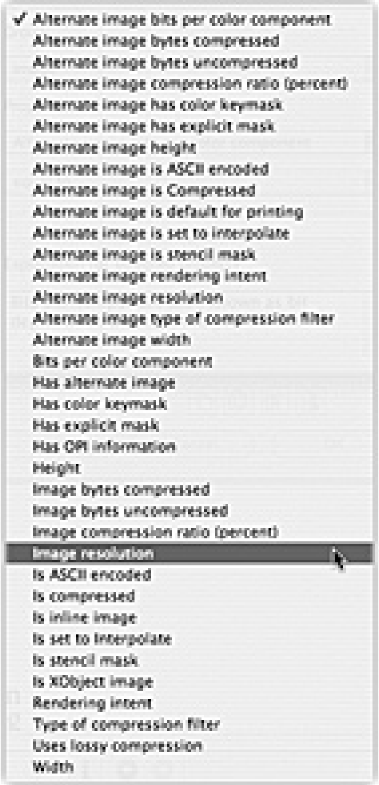


Next Page ->

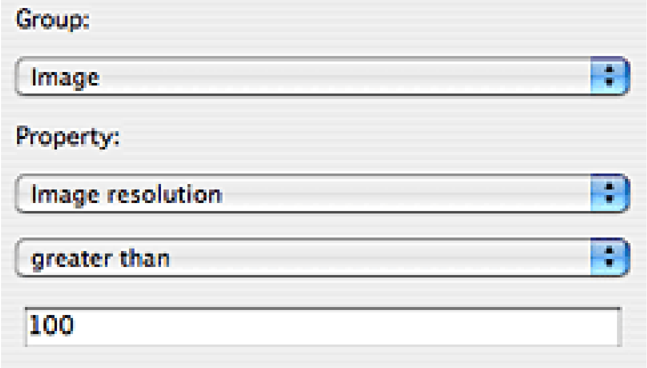
### *Property*

Select which property you want the preflight scanner to examine. The contents of this pop-up menu will differ according to the category you selected earlier.

In our case, we shall pick *Image resolution.*


✓ Alternate image bits per color component
Alternate image bytes compressed
Alternate image bytes uncompressed
Alternate image compression ratio (percent)
Alternate image has color keymask
Alternate image has explicit mask
Alternate image height
Alternate image is ASCII encoded
Alternate image is Compressed
Alternate image is default for printing
Alternate image is set to interpolate
Alternate image is stencil mask
Alternate image rendering intent
Alternate image resolution
Alternate image type of compression filter
Alternate image width
Bits per color component
Has alternate image
Has color keymask
Has explicit mask
Has OPI information
Height
Image bytes compressed
Image bytes uncompressed
Image compression ratio (percent)
Image resolution
Is ASCII encoded
Is compressed
Is inline image
Is set to interpolate
Is stencil mask
Is XObject image
Rendering intent
Type of compression filter
Uses lossy compression
Width

The remaining controls in the *Edit Condition* dialog box change according to the property we select. For *Image Resolution,* we get a pop-up menu (reading "greater than" at right) and a text field.



Group:

Image

Property:

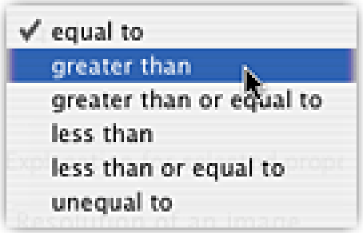
Image resolution

greater than

100

### *Error conditions*

We finish up by specifying what constitutes an error condition for the property we have selected. In our case, we select "greater than" from the pop-up menu and type "100" into the text field; an image resolution greater than 100 will count as an error.



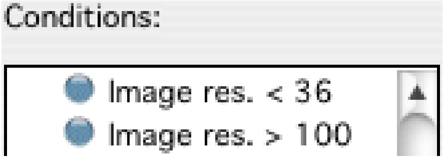
### *Click OK*

Our *Resolution* > *100* condition will be added to the list of conditions.

Next Page ->

## Need One More Condition

The *Journal* profile needs one more image-related condition, looking for images with resolutions less than 36 dpi. I'll leave creating this condition as an exercise for the reader.



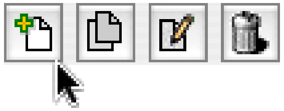
Conditions:

Image res. < 36
Image res. > 100

## Making a Rule

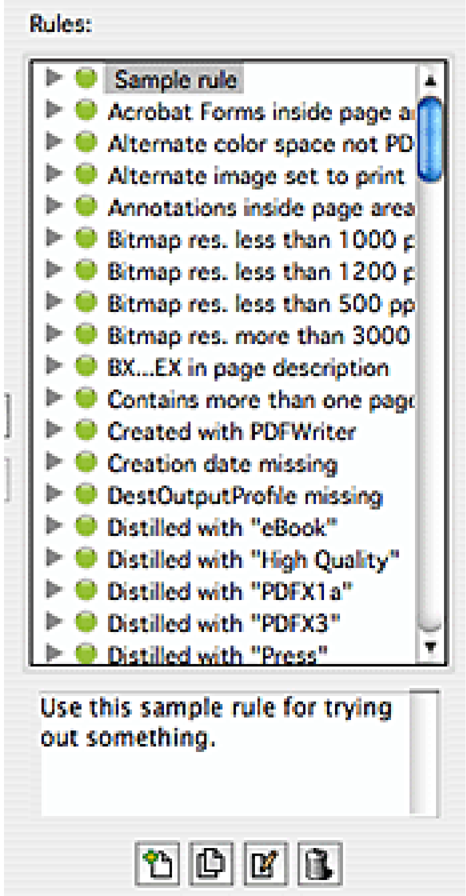
Having created our two image resolution conditions, let's package them into an "Images are wrong" rule for use in our *Journal* profile.

The center list in the *Edit Profiles* panel displays all the rules that Acrobat knows. As before, at the bottom of this list is a set of four buttons, of which the leftmost is the "New Rule" button.



When you click on this button, you get the *Edit Rule* dialog box (next page).

Next Page ->

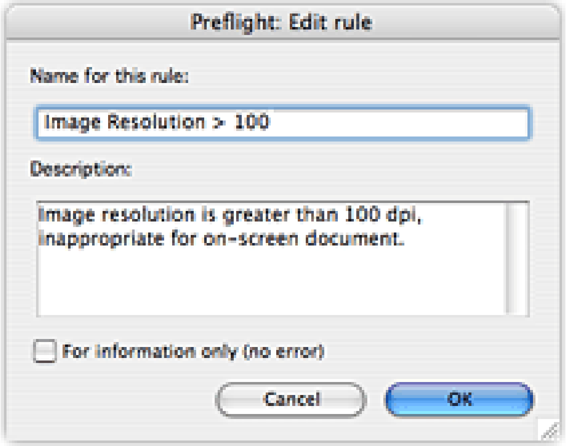


Rules:

Sample rule
Acrobat Forms inside page a
Alternate color space not PD
Alternate image set to print
Annotations inside page area
Bitmap res. less than 1000 p
Bitmap res. less than 1200 p
Bitmap res. less than 500 pp
Bitmap res. more than 3000
BX...EX in page description
Contains more than one page
Created with PDFWriter
Creation date missing
DestOutputProfile missing
Distilled with "eBook"
Distilled with "High Quality"
Distilled with "PDFX1a"
Distilled with "PDFX3"
Distilled with "Press"

Use this sample rule for trying out something.

In the *Edit Rule* dialog box, we supply three pieces of information for the new rule we are making:



Preflight: Edit rule

Name for this rule:

Image Resolution > 100

Description:

Image resolution is greater than 100 dpi, inappropriate for on-screen document.

□ For information only (no error)

- *Name for this rule*
  This is the name that will appear in the *Edit Profile* dialog box.
- *Description*
  This text will appear in the text field beneath the rules list in the *Edit Profile* dialog box.
- *For information only*
  If you select this checkbox, the Acrobat preflight will look for and report on this set of conditions, but will not mark the file as having failed if the conditions are true.

In our case, I have filled out the dialog box as above.

When you click on the *OK* button, this rule will be added to the list in the *Edit Preflight* dialog box. The new rule does not have any conditions associated with it yet.

That's our next step.

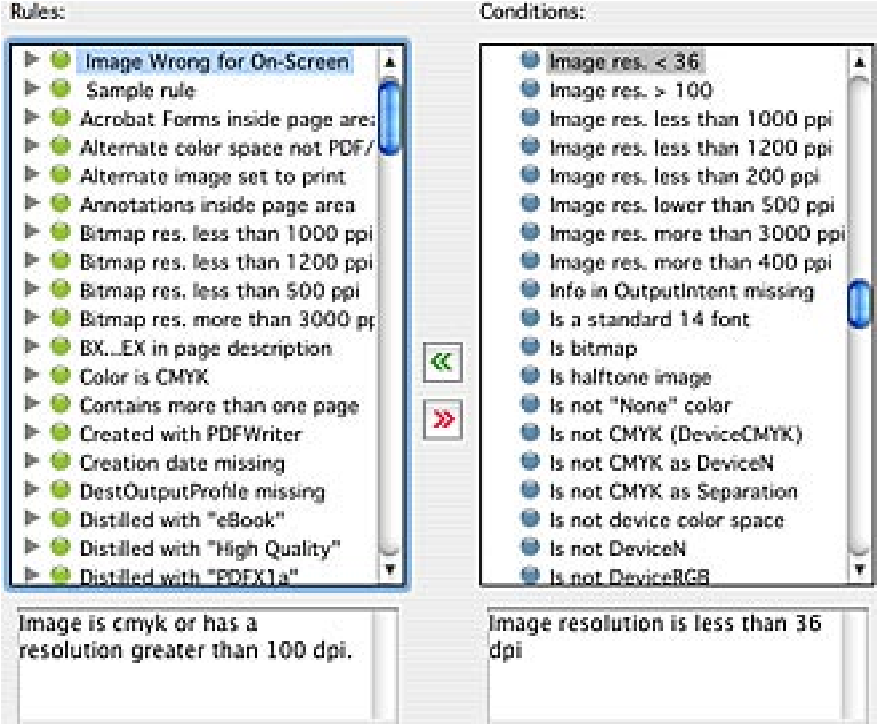
Next Page ->

## Adding Conditions to a Rule

We have now added our *Image is Wrong* rule to Acrobat's list of rules. We want to add to this rule the two *image resolution* conditions we created earlier.

At this point, the Rules and Conditions lists look as at right. Our new, empty *Image is Wrong* rule and the two *Image Resolution* conditions reside in their respective lists.

We add the two conditions to our rule using the green "Add" button in between the two lists.



Rules:

- Image Wrong for On-Screen
- Sample rule
- Acrobat Forms inside page are:
- Alternate color space not PDF/
- Alternate image set to print
- Annotations inside page area
- Bitmap res. less than 1000 ppi
- Bitmap res. less than 1200 ppi
- Bitmap res. less than 500 ppi
- Bitmap res. more than 3000 pp
- BX...EX in page description
- Color is CMYK
- Contains more than one page
- Created with PDFWriter
- Creation date missing
- DestOutputProfile missing
- Distilled with "eBook"
- Distilled with "High Quality"
- Distilled with "PDFX1a"

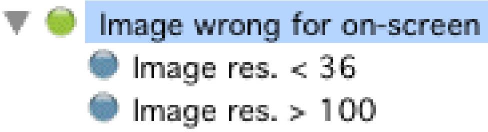
Conditions:

- Image res. < 36
- Image res. > 100
- Image res. less than 1000 ppi
- Image res. less than 1200 ppi
- Image res. less than 200 ppi
- Image res. lower than 500 ppi
- Image res. more than 3000 ppi
- Image res. more than 400 ppi
- Info in OutputIntent missing
- Is a standard 14 font
- Is bitmap
- Is halftone image
- Is not "None" color
- Is not CMYK (DeviceCMYK)
- Is not CMYK as DeviceN
- Is not CMYK as Separation
- Is not device color space
- Is not DeviceN
- Is not DeviceRGB

Image is cmyk or has a resolution greater than 100 dpi.

Image resolution is less than 36 dpi

If you select the *Image Wrong* rule and one of the *Image Resolution* conditions and then click on the green "Add" button, the selected condition is added to the selected rule. Repeating this for our other *image resolution* condition, we see the *Image Wrong* rule now lists its components, as at right.



- Image wrong for on-screen
  - Image res. < 36
  - Image res. > 100

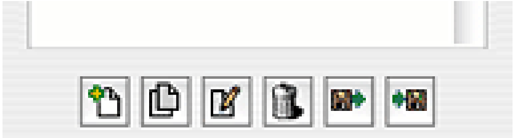
Next Page ->

## Creating the Preflight Profile

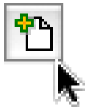
Finally, let's create the actual preflight profile we want to use with the *Acumen Journal* and add to it the *Image is wrong* rule.

### Creating the Profile

The process of creating a preflight profile is nearly identical to the process of creating a rule.



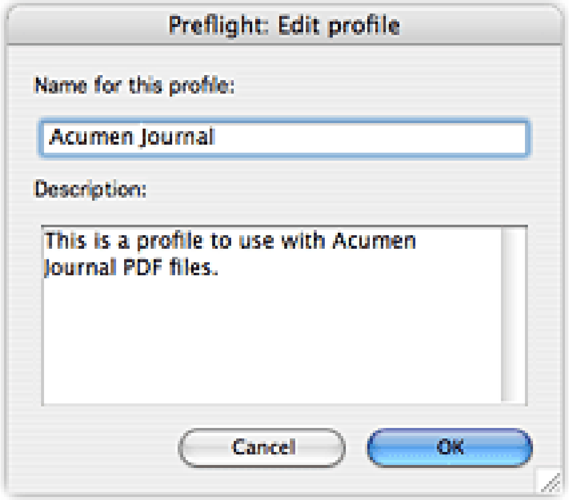
Beneath the list of preflight profiles (the leftmost list in the *Edit Profiles* dialog box), there is a set of four buttons; the leftmost of these is the *New Profile* button. Clicking on this button yields the *Edit Profile* dialog box (below right).



As before, supply a name and a description for the new profile. These will appear in the *Edit Profiles* dialog box.

Click OK and your new profile will be added to Acrobat's list.

Only one step left: we need to populate our newly-created profile with the rules it should use when scanning our PDF files.



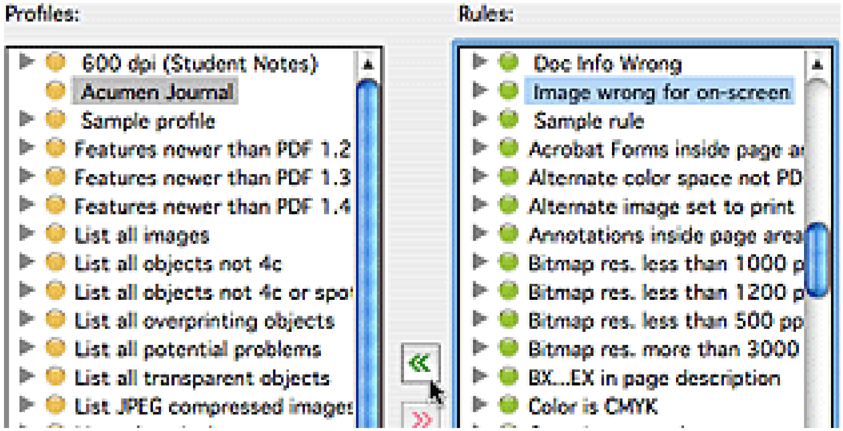
Acumen Journal

This is a profile to use with Acumen Journal PDF files.

Next Page ->

## Adding Rules to the Profile

You add rules to a preflight profile exactly as we added conditions to our *Image is wrong* rule: select the profile and the rule in their respective lists and then click the green "Add" button between the two lists.



Profiles:

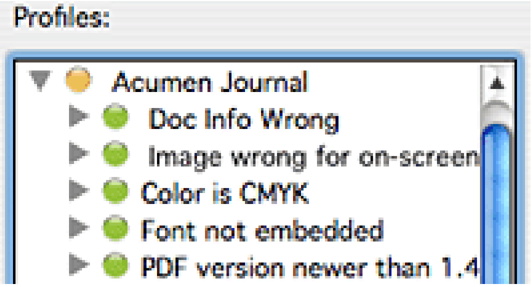
600 dpi (Student Notes)
Acumen Journal
Sample profile
Features newer than PDF 1.2
Features newer than PDF 1.3
Features newer than PDF 1.4
List all images
List all objects not 4c
List all objects not 4c or spot
List all overprinting objects
List all potential problems
List all transparent objects
List JPEG compressed images

Rules:

Doc Info Wrong
Image wrong for on-screen
Sample rule
Acrobat Forms inside page ar
Alternate color space not PD
Alternate image set to print
Annotations inside page area
Bitmap res. less than 1000 p
Bitmap res. less than 1200 p
Bitmap res. less than 500 pp
Bitmap res. more than 3000
BX...EX in page description
Color is CMYK

Each click of the green button will add the selected rule to the profile.

The final *Acumen Journal* profile is made up of several rules that look for a number of error conditions:



Profiles:

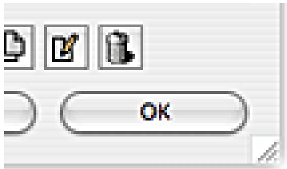
Acumen Journal
Doc Info Wrong
Image wrong for on-screen
Color is CMYK
Font not embedded
PDF version newer than 1.4

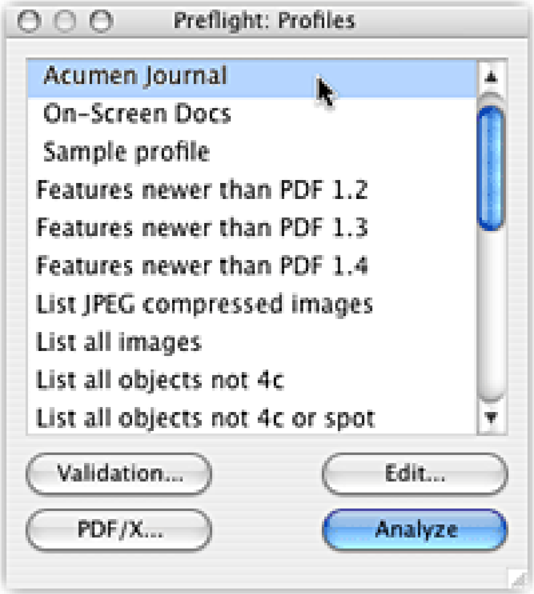
- The document information is wrong. (Specifically, the title, author, or copyright information is missing.)
- An image is wrong for on-screen viewing. (This one we created in this issue.)
- Any color is CMYK.
- Any font is not embedded
- The file requires Acrobat 6.

These are left for the reader to create and add to the profile.

Next Page ->

**Done!** That's it. Click on the *OK* button in the *Edit Profile* dialog box; Acrobat will add our *Acumen Journal* profile to the list of profiles available when we preflight a PDF file.




Preflight: Profiles

Acumen Journal
On-Screen Docs
Sample profile
Features newer than PDF 1.2
Features newer than PDF 1.3
Features newer than PDF 1.4
List JPEG compressed images
List all images
List all objects not 4c
List all objects not 4c or spot

Acrobat 6's Preflight feature is very powerful and easily usable. Setting up your own preflight profiles takes  a little work, but it's not particularly difficult. Once you have created your profiles, actually conducting preflight scans is very quick and easy.

Return to Main Menu

# CIDFonts: a Primer

In Europe and the Western Hemisphere, there is a tendency to think of character codes as being one byte only. Indeed, the words "character" and "byte" are often used synonymously in programming.

This, of course, is a reflection of the size of the character sets routinely used in that part of the world; a single byte is sufficient to encode a useful set of characters, and so this is all that was devoted to the task for many decades.

In much of the rest of the world (and East Asia in particular), the size of the character set needed to do useful printing exceeds the capacity of a single byte. In those scripts, computer systems must routinely use *multibyte fonts,* wherein each character within a string is represented by two (or more) bytes.

Modern PostScript provides a new font format called *CIDFonts,* which makes it remarkably easy to support Kanji and other multibyte scripts. This is also the mechanism by which you would support Unicode in modern PostScript.

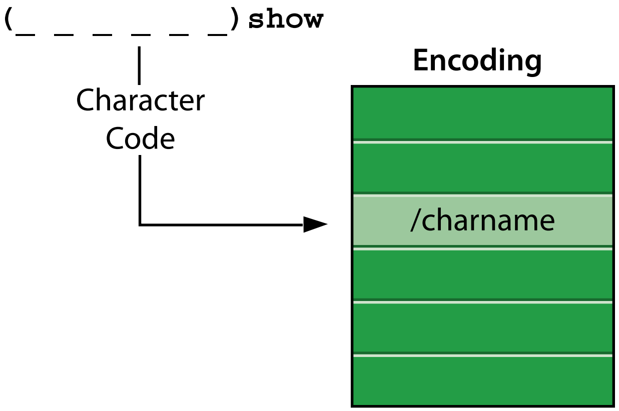
This month's PostScript article presents an overview of how CIDFonts work.

Next Page ->

## Some History: PostScript Fonts

### PostScript Level 1

The original PostScript Level 1 had no support for multibyte fonts. The Type 1 and Type 3 font specifications, dating to that era, both assume 1-byte character codes. The *Encoding* array in these fonts, which defines the correspondence between character codes and characters, is explicitly limited to 256 entries.

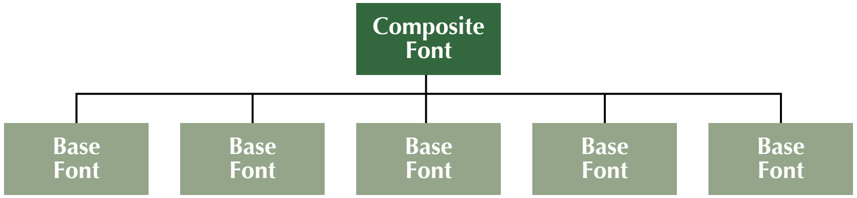


[Next Page ->]

## Composite Fonts

In 1987 (or so), Adobe introduced into Asian PostScript devices support for *Composite Fonts*. This was Adobe's first attempt at supporting multibyte fonts. They are a standard feature in PostScript level 2.

A composite font is a font that contains other fonts, rather than characters directly. The descendent fonts may be Type 1 or Type 3 "base fonts," containing characters, or may, themselves, be composite fonts, containing more fonts in turn.



A composite font can have an indefinite number of characters contained among its descendents. However, implementing any of the standard Asian character encoding (such as JIS), is a surprisingly difficult. (I once had to implement one of the simpler Asian character encodings as a composite font; I still twitch when I think about that job, although the doctor assures me I continue to improve.)

## *They Have Their Uses*

Because of the tedium involved in implementing Asian character encodings, composite fonts were never a popular way of doing multibyte fonts. They *do* turn out to be very useful for printing Western Roman text, however. The May 2002 issue of the *Acumen Journal* describes the benefit and use of composite fonts in printing single-byte text.

Next Page ->

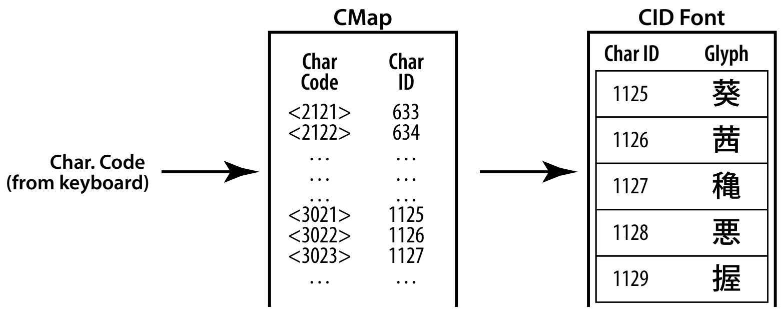
## CID Fonts

Adobe introduced CID fonts in version 2015 of their PostScript interpreter, about midway through the Level 2 era. This made it *much* easier to support multibyte fonts and is now the standard way of doing multibyte text in PostScript.

A CID font can have an any number of characters in it, each assigned an arbitrary numeric code, a “character ID” (whence “CID”). The character ID assigned to a character is arbitrary, though it must be documented; in particular, the ID has no particular relationship to the character encoding used on a computer system.

### *CID Fonts and CMaps*

A CID font is simply a repository for the character shapes (“glyphs”) that make up the font. To use the font with a computer system, you must specify the mapping from character codes generated by the system’s keyboard to the character ID for that character in the CID Font. This mapping is supplied by a *CMap Resouce,* external to the font, itself.



CMap

| Char Code | Char ID |
|---|---|
| <2121> | 633 |
| <2122> | 634 |
| … | … |
| … | … |
| … | … |
| <3021> | 1125 |
| <3022> | 1126 |
| <3023> | 1127 |
| … | … |

CID Font

| Char ID | Glyph |
|---|---|
| 1125 | 葵 |
| 1126 | 茜 |
| 1127 | 穐 |
| 1128 | 悪 |
| 1129 | 握 |

### *CID-Keyed Fonts*

Thus, you never use a CID font by itself, but always in concert with a CMap resource. The combination of the two is referred to as a *CID-Keyed Font.*

Next Page ->

## CID-Keyed Font Components

Three components make up a CID-Keyed font: a *character collection document*, a *CMap resource*, and the CIDFont *resource*, itself. Let's look at each of these in more detail.

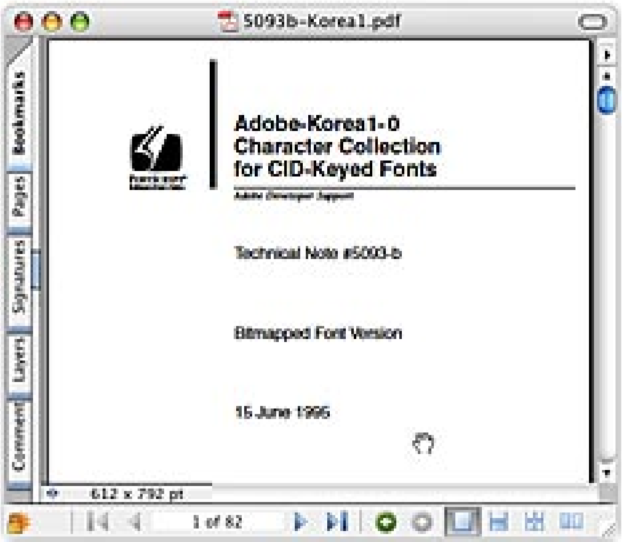
### Character Collection Document

A CID-Keyed font starts with a *character collection document*. This is a published document that defines a named collection of characters. Specifically, it defines what characters we are going to include in our CID font.

It is not practical to include all of Kanji's tens of thousands of characters in your Kanji CID font.

Before you can create a Kanji font, you need to declare what characters you will put in that font, chosen from among the thousands possible. This is the purpose of a character collection document: it defines a named collection of characters and the character ID that you will assign to each character.

A character collection document does not make for exciting reading; it typically consists of page after page of pictured character glyphs and numbers.



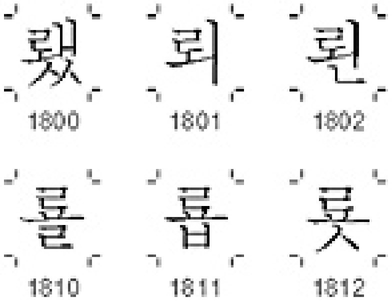
Adobe-Korea1-0
Character Collection
for CID-Keyed Fonts

Adobe Developer Support

Technical Note #5093-b

Bitmapped Font Version

15 June 1996

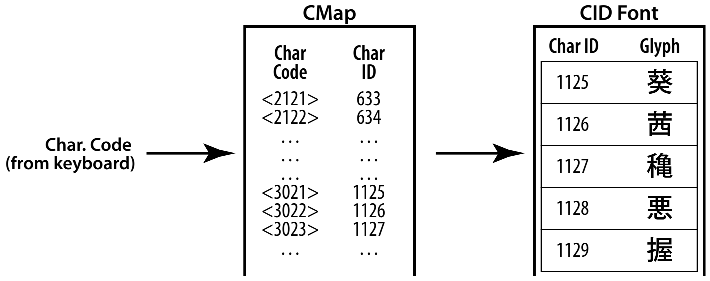


Next Page ->

## CMap Resource

The character ID associated with each character in your character collection document are arbitrary; you can pick whatever values you wish for these numbers. Specifically, they need have no relation whatsoever to the character *encoding*, the character codes used by a particular computer system.

To print strings generated on a particular system, you need to provide a mapping from the system's character encoding (ASCII, Shift-JIS, etc.) to the character IDs defined by the character collection document. This table of correspondence resides in a *CMap resource*.



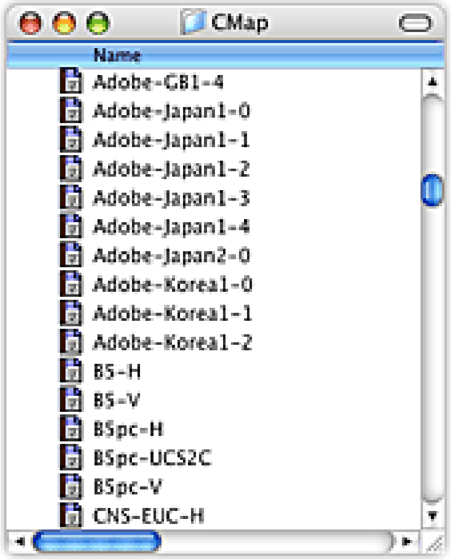
CMap

| Char Code | Char ID |
|---|---|
| <2121> | 633 |
| <2122> | 634 |
| … | … |
| … | … |
| … | … |
| <3021> | 1125 |
| <3022> | 1126 |
| <3023> | 1127 |
| … | … |

CID Font

| Char ID | Glyph |
|---|---|
| 1125 | 葵 |
| 1126 | 茜 |
| 1127 | 穐 |
| 1128 | 悪 |
| 1129 | 握 |

Adobe provides CMap resources for all the commonly-used Asian character encodings. *Distiller* and *Jaws PDF Creator* both ship with a very large collection of CMap resources as part of their Asian language support.



Name
- Adobe-GB1-4
- Adobe-Japan1-0
- Adobe-Japan1-1
- Adobe-Japan1-2
- Adobe-Japan1-3
- Adobe-Japan1-4
- Adobe-Japan2-0
- Adobe-Korea1-0
- Adobe-Korea1-1
- Adobe-Korea1-2
- B5-H
- B5-V
- B5pc-H
- B5pc-UCS2C
- B5pc-V
- CNS-EUC-H

## CIDFont Resource

Finally, the third component of a CID-Keyed font is the CIDFont resource itself. This PostScript resource (of type *CIDFont)* is a repository of character shapes defined using Type 1, TrueType, or PostScript drawing commands.

Next Page ->

## Creating and Using CID-Keyed Fonts

You join a CIDFont resource and a CMap resource into a CID-Keyed font with a PostScript operator added to version 2015 of the language: *composefont*.

```
   /CKFontName CMap [ CIDFont0 CIDFont1 ... ] composefont
                                                 => <<fdict>>
```

The *composefont* operator creates a CID-Keyed font with a specified name. It returns the font dictionary for that newly-created CID-Keyed font.

The *composefont* arguments are:

*/CKFontName* This is the name of the resulting CID-Keyed font. This is the name you can later hand to the *selectfont* operator to use this font.

*CMap* This is the CMap resource to be used in the CID-Keyed font. This may be the name of a resource of type *CMap* or the actual CMap dictionary, itself.

*[ CIDFont… ]* This is an array of one or more CIDFonts that will be the sources of the characters in the CID-Keyed font. Each entry in the array may be either the name of a CIDFont resource or the actual resource dictionary.

Note that you supply an array of CIDFonts; the characters in your final CID-Keyed font may be drawn from a combination of CIDFonts. The CIDFonts must have non-overlapping character IDs (and, therefore, implement different character collections).

Next Page ->

Thus, a typical call to *composefont* will look something like this:

```
/Ryumin-RKSJ /83pv-rksj-h [ /Ryumin-Light ] composefont
```

The above call to *composefont* leaves on the operand stack the font dictionary for a new CID-Keyed font named *Ryumin-RKSJ*. If you choose to not use this font immediately in your PostScript code, you can always retrieve it with *selectfont* or *findfont:*

```
/Ryumin-RKSJ 36 selectfont
```

The *show* operator will then treat the contents of its string argument as, in this case, Roman-Kana-Shift-JIS character codes, printing characters taken from the CIDFont *Ryumin-Light*.

```
72 600 moveto
(........) show
```

Next Page ->

## CID Support Library

Support for CID fonts was introduced in Adobe PostScript version 2015. You can use CID fonts and CMap resources with earlier PostScript printers by using the *CID Support Library (CSL).* This is a set of ProcSet resources that teaches an earlier PostScript Level 2 printer (specifically, PostScript 2011 through 2014) how to use CID fonts. (Under the hood, it uses the CMap and CIDFont to create a composite font.)

When downloaded to a printer, the CSL redefines the *showpage* operator to accept a name consisting of a concatenated CIDFont and CMap resource name, the two names separated by one or two dashes:

```
/Ryumin-RKSJ--83pv-rksj-h findfont 26 scalefont setfont
```

The CSL is available, free, from Adobe Systems through their "Adobe Solutions Network." Just go to www.adobe.com and search for "CSL."

## CIDFonts Good

CIDFonts have made it relatively simple to support Kanji and other multibyte scripts in PostScript. Particularly since Adobe has already made most of the CMap resources you will likely need, there is relatively little work entailed in using these fonts.

Return to Main Menu

# Schedule of Classes, Apr - Sep 2004

Following are the dates  of Acumen Training's upcoming PostScript and PDF Technical classes. Clicking on a class name below 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 worldwide. See the Acumen Training web site for more information.

## Technical Classes

Sorry there aren't a lot of classes scheduled through July. I've an extremely busy on-site schedule, so I can't conduct many open classes.

| | | | |
|---|---|---|---|
| **PDF File Content and Structure** | Jun 14–17 | | Aug 30–Sep 2 |
| **PostScript Foundations** | Apr 26–30 | | Apr 2–6 |
| **Variable Data PostScript** | Aug 9–13 | | |
| **Advanced PostScript** | | | Sep 13–17 |
| **PostScript for Support Engineers** | | | |
| **Jaws Development** | | *On-site only* | |

*Course Fee* The PostScript and PDF classes cost $2,000 per student.

Registration Info

Acrobat Classes

# Acrobat Class Schedule

These classes are taught occasionally in Costa Mesa, California, and on corporate sites. Clicking on a course name below will take you to the class description on the Acumen Training web site.

**Acrobat Essentials**

*No Acrobat classes scheduled for this quarter.* See the Acumen Training website regarding setting up an on-site class.

**Interactive Acrobat**

**Creating Acrobat Forms**

**Acrobat Class Fees**

*Acrobat Essentials* and *Creating Acrobat Forms* (½-day each) cost $180.00 or $340.00 for both classes. There is a 10% discount if three or more people from the same organization sign up for the same class.

Registration ->

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

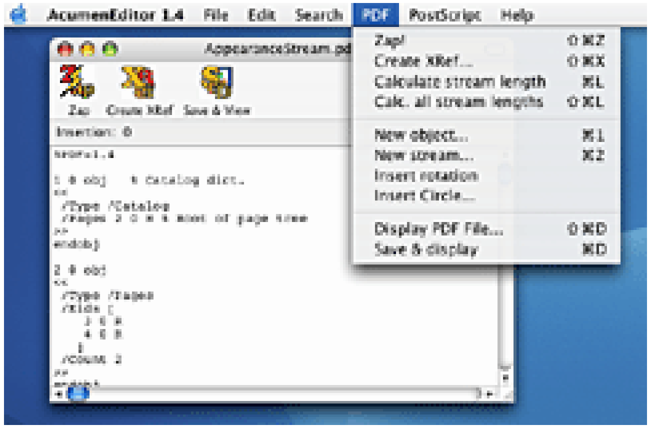
Back issues of the *Acumen Journal* are available at the Acumen Training website: http://www.acumenjournal.com/AcumenJournal.html

Return to First Page

# What’s New at Acumen Training?

## *Acumen Editor* is on the website

The *Acumen Editor* will now be available on the Acumen Training [Resources](Resources) page.



```
%PDF-1.4

1 0 obj    % catalog dict.
<<
 /Type /Catalog
 /Pages 2 0 R % root of page tree
>>
endobj

2 0 obj
<<
 /Type /Pages
 /Kids [
    3 0 R
    4 0 R
   ]
 /Count 2
>>
endobj
```

*Acumen Editor* is the software used in the *PDF File Content and Structure* class. It’s a text editor with a few features convenient for creating hand-written PDF files. Among other things, it:

- Inserts *xref* table and stream lengths.
- Inserts proper PDF code for rotations and circles.
- Displays the current PDF file in your Acrobat viewer.

Eventually, *Acumen Editor* will replace *Download Mechanic,* used in the PostScript classes.

This software is freely available to anyone who wants to use it, but there is currently no documentation. People who have taken the *PDF File Content and Structure* class already have used it; everyone else will have to figure it out for now. It’s not that hard.

There is both a Mac and Windows version available on the 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? Does it remind you of your old Uncle Pete who absolutely won't stop talking even though his dentures click?

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

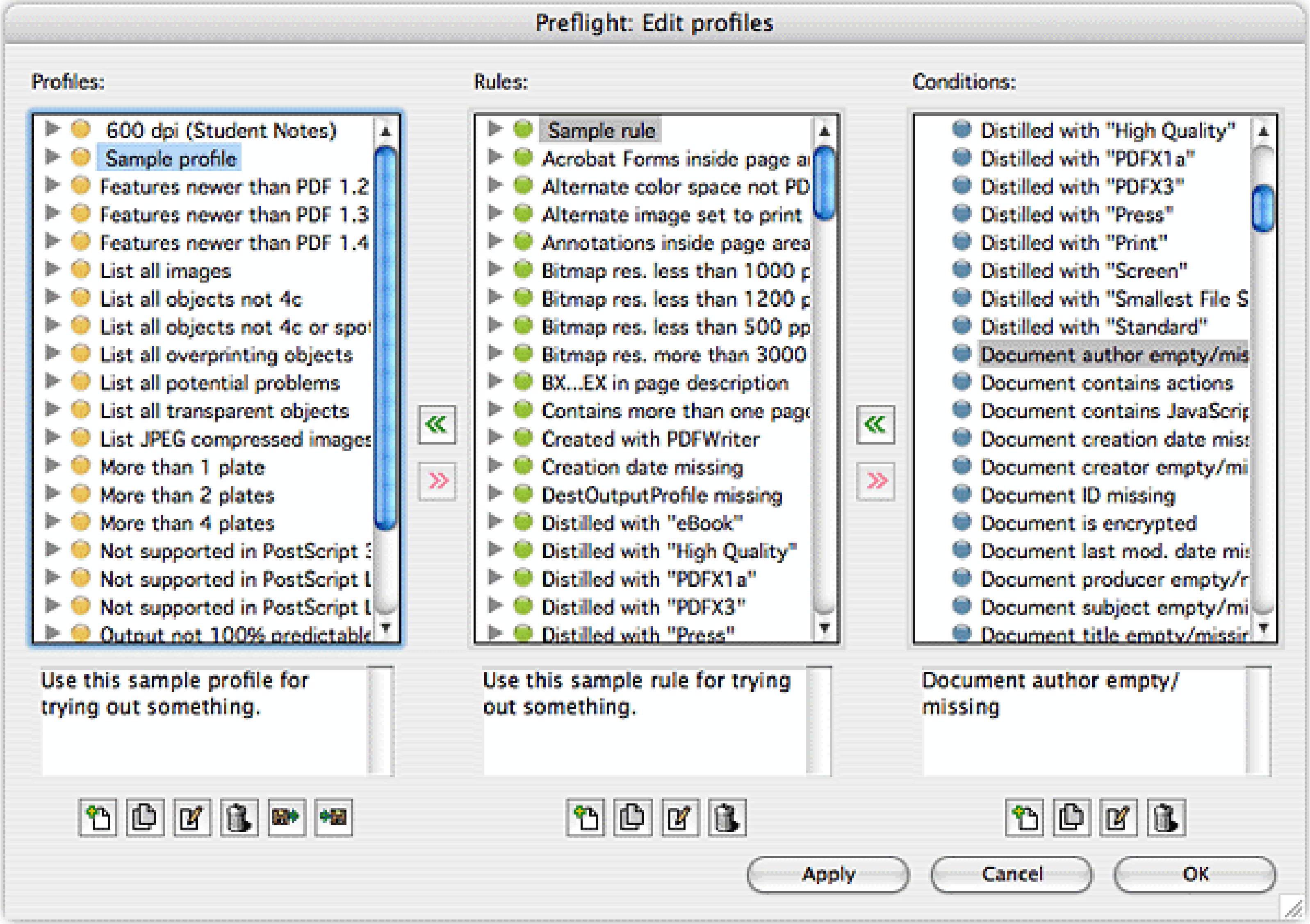
**Questions and Answers.** Do you have any questions about Acrobat, PDF, or PostScript? Feel free to email me about. I'll answer your question if I can. (If enough people ask the same question, I can turn it into a Journal article.)

Please send any comments, questions, or problems to:

journal@acumentraining.com

Return to Menu

# Preflight Profiles



Preflight: Edit profiles

Profiles:

- 600 dpi (Student Notes)
- Sample profile
- Features newer than PDF 1.2
- Features newer than PDF 1.3
- Features newer than PDF 1.4
- List all images
- List all objects not 4c
- List all objects not 4c or spot
- List all overprinting objects
- List all potential problems
- List all transparent objects
- List JPEG compressed images
- More than 1 plate
- More than 2 plates
- More than 4 plates
- Not supported in PostScript 3
- Not supported in PostScript L
- Not supported in PostScript L
- Output not 100% predictable

Use this sample profile for trying out something.

Rules:

- Sample rule
- Acrobat Forms inside page ar
- Alternate color space not PD
- Alternate image set to print
- Annotations inside page area
- Bitmap res. less than 1000 p
- Bitmap res. less than 1200 p
- Bitmap res. less than 500 pp
- Bitmap res. more than 3000
- BX...EX in page description
- Contains more than one page
- Created with PDFWriter
- Creation date missing
- DestOutputProfile missing
- Distilled with "eBook"
- Distilled with "High Quality"
- Distilled with "PDFX1a"
- Distilled with "PDFX3"
- Distilled with "Press"

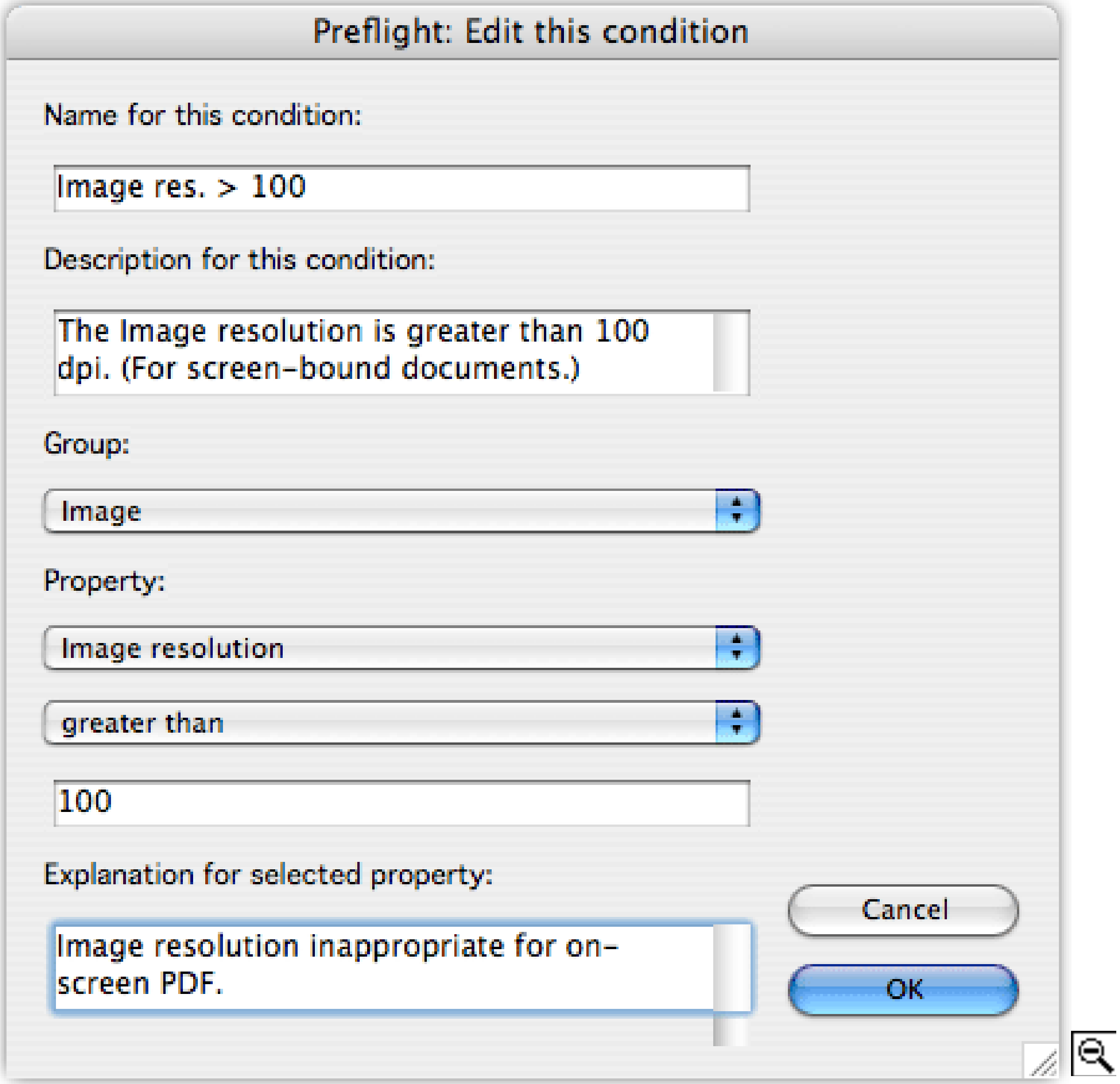
Use this sample rule for trying out something.

Conditions:

- Distilled with "High Quality"
- Distilled with "PDFX1a"
- Distilled with "PDFX3"
- Distilled with "Press"
- Distilled with "Print"
- Distilled with "Screen"
- Distilled with "Smallest File S
- Distilled with "Standard"
- Document author empty/mis
- Document contains actions
- Document contains JavaScrip
- Document creation date miss
- Document creator empty/mi
- Document ID missing
- Document is encrypted
- Document last mod. date mis
- Document producer empty/r
- Document subject empty/mi
- Document title empty/missir

Document author empty/ missing

# Snap to Item



Preflight: Edit this condition

Name for this condition:

Image res. > 100

Description for this condition:

The Image resolution is greater than 100 dpi. (For screen-bound documents.)

Group:

Image

Property:

Image resolution

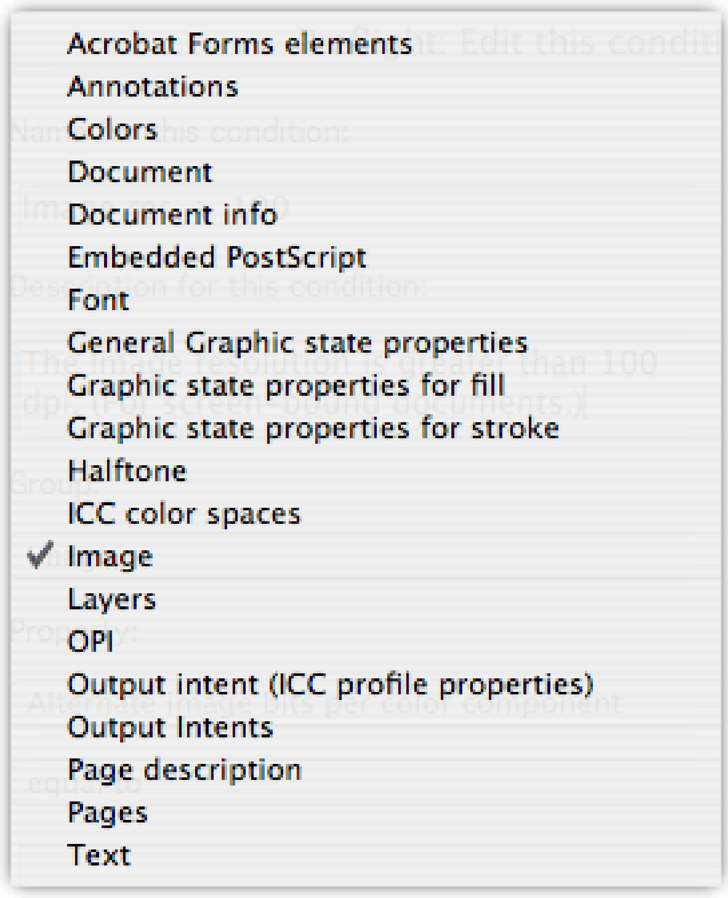
greater than

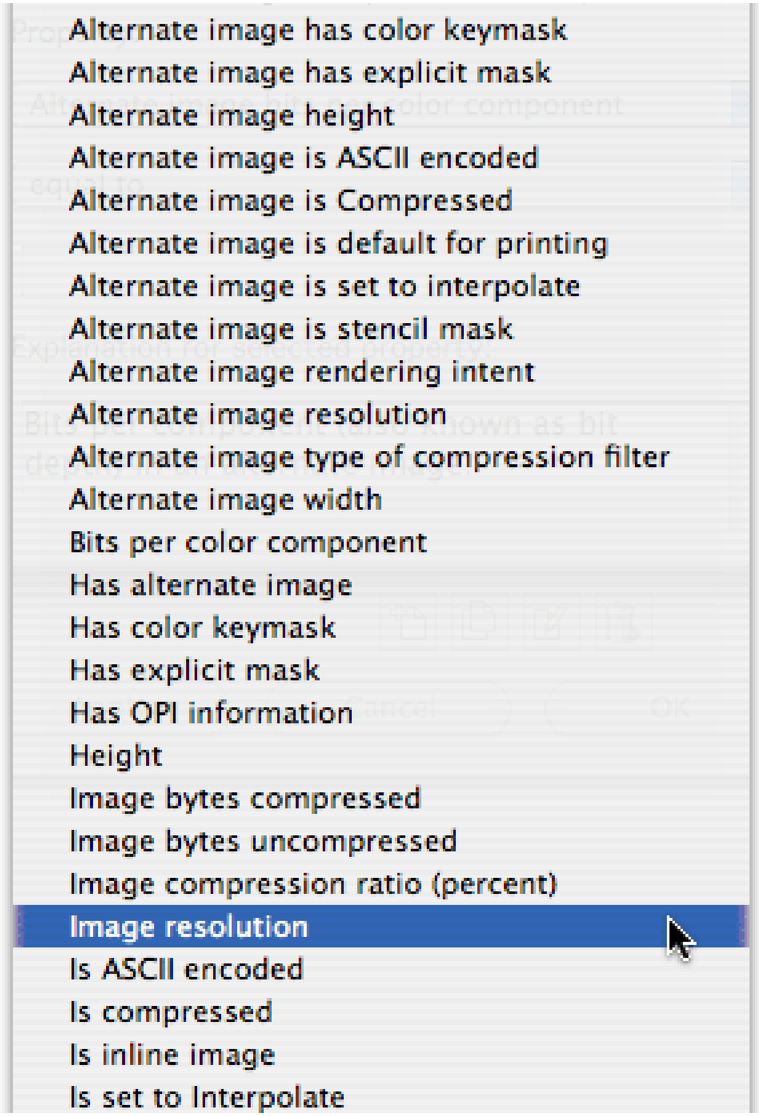
100

Explanation for selected property:

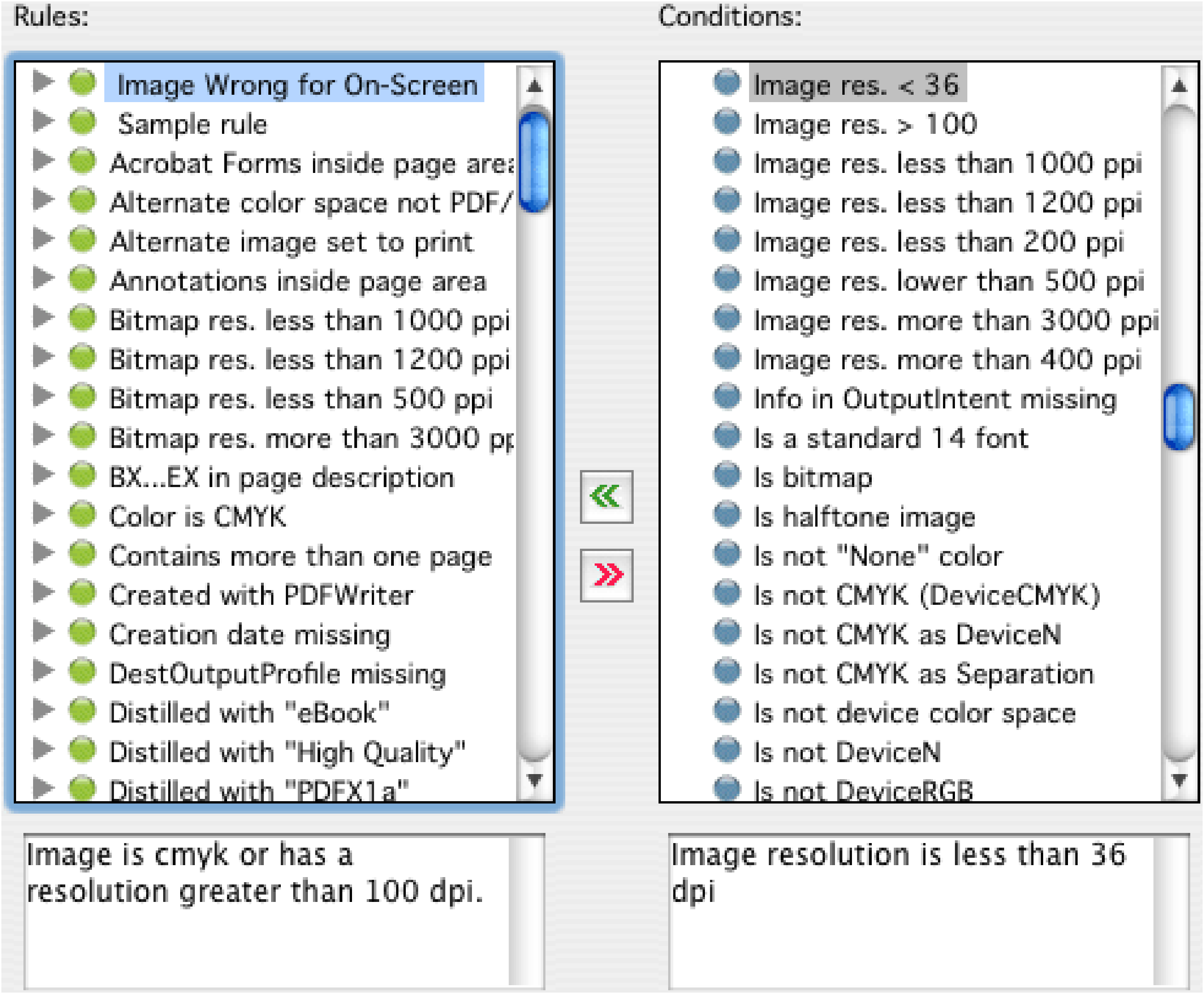
Image resolution inappropriate for on-screen PDF.

# Categories and Image Properties




Alternate image has color keymask
Alternate image has explicit mask
Alternate image height
Alternate image is ASCII encoded
Alternate image is Compressed
Alternate image is default for printing
Alternate image is set to interpolate
Alternate image is stencil mask
Alternate image rendering intent
Alternate image resolution
Alternate image type of compression filter
Alternate image width
Bits per color component
Has alternate image
Has color keymask
Has explicit mask
Has OPI information
Height
Image bytes compressed
Image bytes uncompressed
Image compression ratio (percent)
Image resolution
Is ASCII encoded
Is compressed
Is inline image
Is set to Interpolate

# Rules and Conditions Lists



Rules:

- Image Wrong for On-Screen
- Sample rule
- Acrobat Forms inside page area
- Alternate color space not PDF/
- Alternate image set to print
- Annotations inside page area
- Bitmap res. less than 1000 ppi
- Bitmap res. less than 1200 ppi
- Bitmap res. less than 500 ppi
- Bitmap res. more than 3000 pp
- BX...EX in page description
- Color is CMYK
- Contains more than one page
- Created with PDFWriter
- Creation date missing
- DestOutputProfile missing
- Distilled with "eBook"
- Distilled with "High Quality"
- Distilled with "PDFX1a"

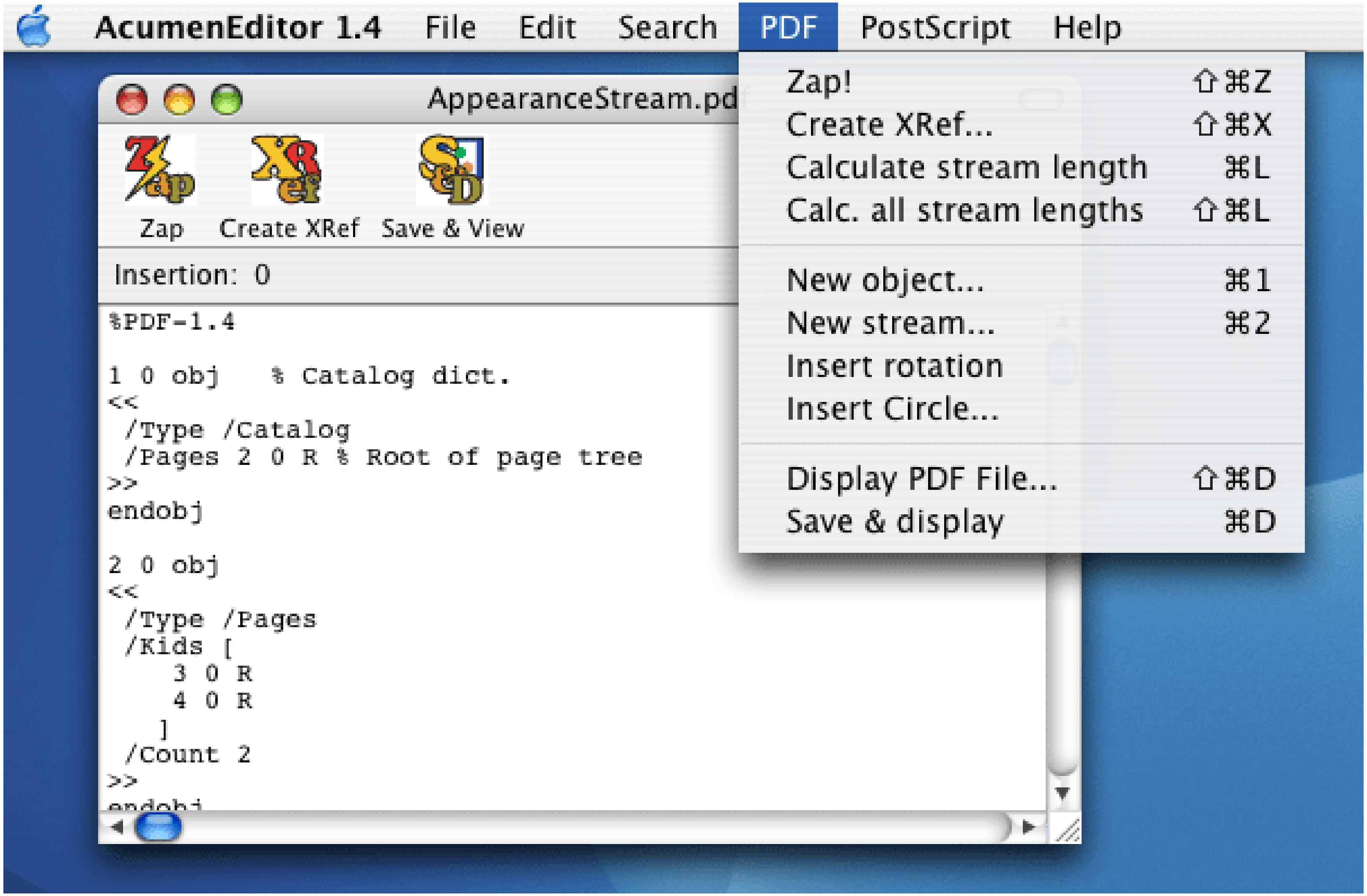
Conditions:

- Image res. < 36
- Image res. > 100
- Image res. less than 1000 ppi
- Image res. less than 1200 ppi
- Image res. less than 200 ppi
- Image res. lower than 500 ppi
- Image res. more than 3000 ppi
- Image res. more than 400 ppi
- Info in OutputIntent missing
- Is a standard 14 font
- Is bitmap
- Is halftone image
- Is not "None" color
- Is not CMYK (DeviceCMYK)
- Is not CMYK as DeviceN
- Is not CMYK as Separation
- Is not device color space
- Is not DeviceN
- Is not DeviceRGB

Image is cmyk or has a resolution greater than 100 dpi.

Image resolution is less than 36 dpi

# Acumen Editor



Insertion: 0

```
%PDF-1.4

1 0 obj    % Catalog dict.
<<
 /Type /Catalog
 /Pages 2 0 R % Root of page tree
>>
endobj

2 0 obj
<<
 /Type /Pages
 /Kids [
     3 0 R
     4 0 R
    ]
 /Count 2
>>
endobj
```