

AcroTeX.Net

**The annot\_pro Package**  
**Text, Stamp, and File Attachment**  
**Annotations**

**D. P. Story**

Copyright © 2009 [dpstory@acrotex.net](mailto:dpstory@acrotex.net)  
Prepared: March 10, 2009

[www.acrotex.net](http://www.acrotex.net)  
Version 1.0

## Table of Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Requirements</b>	<b>3</b>
2.1	TeX Package Requirements . . . . .	3
2.2	PDF Creator Requirements . . . . .	3
<b>3</b>	<b>Installation</b>	<b>4</b>
<b>4</b>	<b>The <code>\annotpro</code> Command</b>	<b>5</b>
4.1	Key-values common to all annotations . . . . .	6
4.2	Key-values for text annotations . . . . .	8
4.3	Key-values for stamp annotations . . . . .	9
4.4	Key-values file attachment annotations . . . . .	11
4.5	Setting Global Options with <code>\setAnnotOptions</code> . . . . .	12

## 1. Introduction

This package is used to create text, stamp, and file attachment annotations using **Adobe Distiller**, these annotations can be viewed in Adobe Reader. For users of pdf(la)tex, use the pdfcomment package by Josef Kleber.<sup>1</sup>

The package primarily in support of my [AcroTeX PDF Blog](#). I plan to use sticky notes, file attachments, and custom stamps to make side-comments, and to provide source files.

## 2. Requirements

The requirements for your  $\LaTeX$  system, and well as any other software, is highlighted in this section.

### 2.1. $\LaTeX$ Package Requirements

The following packages, in addition to the standard  $\LaTeX$  distribution, are required:

1. The xkeyval package is used to set up the key-value pairs of the `\annotpro` command. Get a recent version.
2. The xcolor package is strongly recommended.
3. The hyperref package, a recent version.
4. If you want to create custom stamps (on the fly) using the techniques developed for that purpose, the graphicxsp Package is required.<sup>2</sup>

The `annot_pro` package is part of the [AeB Pro](#) family, which means **Adobe Distiller** is required. The components of [AeB](#) and [AeB Pro](#) are not required.<sup>3,4</sup>

### 2.2. PDF Creator Requirements

The `annot_pro` package requires **Acrobat Distiller 5.0** (or later) as the PDF creator. The document author typically uses `dvips` (or `dvipsone`) to produce a PostScript file, which is then distilled to obtain a PDF.

---

<sup>1</sup> Available from CTAN in the directory `macros/latex/contrib/pdfcomment/`.

<sup>2</sup> Available at <http://www.math.uakron.edu/~dpstory/graphicxsp.html>

<sup>3</sup> AeB: <http://www.math.uakron.edu/~dpstory/webeq.html>

<sup>4</sup> AeB Pro: [http://www.math.uakron.edu/~dpstory/aeb\\_pro.html](http://www.math.uakron.edu/~dpstory/aeb_pro.html)

If you wish to use (dynamically) created stamps that have opacity less than 1, you need to distill using **Standard\_transparency.joboptions** with distiller, in this case, **Acrobat Distiller 6.0** (or later) is required; otherwise, this distiller job options file is not needed.

The **Standard\_transparency.joboptions** file is supplied with the `graphicxsp` package; the documentation of the `graphicxsp` package includes installation instructions.

### 3. Installation

The installation is simple enough. Unzip `annot_pro.zip` in a folder that is on your  $\text{\LaTeX}$  search path. Refresh your filename database, if appropriate.

**Important:** When creating a file attachment annotation, you must specify a path to the file to be attached, and distiller must embed this file. In recent versions of Acrobat, security restrictions have been put in place to prevent **Distiller** from reading files (the PostScript **file** operator does not work). Fortunately, Distiller has a switch that turns off this particular restriction. To successfully use this package, therefore, you need to run Distiller by using the `-F` command line switch.

I personally use the WinEdt application as my text editor,<sup>5</sup> and have defined a Distiller button on my toolbar. The Distiller button executes the following WinEdt macro.

```
Run(|"c:\Program Files\Adobe\Acrobat 9.0\Acrobat\acrodist.exe" -F
    "%P\%N.ps"|,'%P',0,0,'%N.ps - Distiller',1,1);
```

Note the use of the `-F` switch for `acrodist.exe`. Here the path is to the version 9 of distiller, Adobe has been fairly consistent about path to its application; for example, the path to distiller 7 (should be)

```
c:\Program Files\Adobe\Acrobat 7.0\Acrobat\acrodist.exe
```

If this package is used to create file attachment annotations without the `-F` switch, you typically get the following error message in the Distiller log file

```
%%[ Error: undefinedfilename; OffendingCommand: file ]%%
```

This tells you that either you have not started Distiller with the `-F` command line switch, or Distiller can't find one of the files that the **file** operator was trying to read.

<sup>5</sup>WinEdt home page: [www.winedt.com](http://www.winedt.com)

**Mac OS Users.** The above comments on the `-F` command line switch is for Windows users, Mac OS users must choose the `AllowPSFileOps` user preference, this is located in the `plist`, possibly located at

```
/Users/[User]/Library/Preferences/com.adobe.distiller9.plist
```

You can also use Spotlight, the search utility on Mac, to search for the string `com.adobe.distiller`; the result might be

```
com.adobe.distiller9.plist.6
```

Clicking on this find, Spotlight opens `com.adobe.distiller9.plist` in the `plist` editor, see [Figure 1](#). If necessary, click on the arrow next to the `Root` to expand the choices, then click the up and down arrows at the far right in the `AllowPSFileOps` row to select `Yes` as the value.

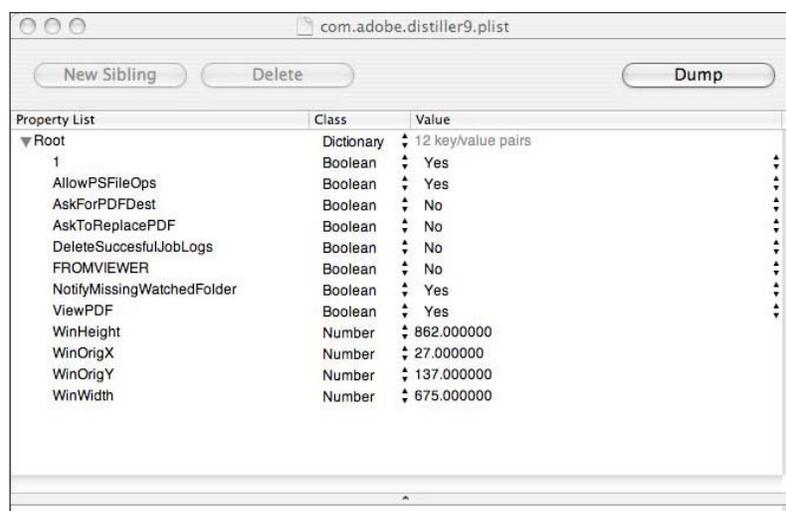


Figure 1: `com.adobe.distiller9.plist`

## 4. The `\annotpro` Command

The main command of this package is `\annotpro`; the command is controlled by its optional parameters. The same command can create a text annotation (sticky note), a stamp annotation, or a file attachment. The syntax of this command is

<sup>6</sup>In the case of Adobe Distiller, version 9.0

```
\annotpro [<key-values>] {<text>}
```

**Parameter Description:** The optional argument is one or more key-value pairs that describe the annotation; the required argument, `<text>`, is the “contents” of the annotation; for the text and stamp annotation, `<text>` becomes the contents of the popup annotation, for a file attachment annotation, which has not associated pop-up, it is the description of the file attachment that appears in the file attachment panel of Adobe Reader. In the latter case, the value of `<text>` should be rather short.

The key-value pairs are described over the next few sections.

#### 4.1. Key-values common to all annotations

The following are key-value pairs common to all annotations.

- `type`: The `type` key determines the type of annotation to be produced; permissible values are `text`, `stamp`, and `fileattachment`. This key is optional, if not present, `type=text` is assumed.
- `name`: The name of the icon to use for the declared type. Permissible values are dependent on the type, and are discussed in subsequent sections.
- `title`: Text to be displayed in the title bar of the annotation’s pop-up window when open and active. By convention, this entry identifies the user who made the annotation, though any (short) text may be used. You can use `\setAnnotOptions` to globally set the title of each annotation, perhaps using your name.

```
\setAnnotOptions{title={D. P. Story}}
```

- `subject`: Text representing a short description of the subject of the annotation. You can use `\setAnnotOptions` to globally set the subject of each annotation,

```
\setAnnotOptions{title={D. P. Story},
subject={AcroTeX Communiqu'e}}
```

- `color`: The color of the title bar of the pop-up window of the annotation.
- `readonly`: Set the annotation to readonly. The user can click on the annot to see the popup, but the user, if using Acrobat, cannot move the annotation around on the page. The popup window can still be move by the user. (This property makes no difference to the user of Adobe Reader.)

- `opacity`: The opacity value ( $0 \leq \text{opacity} \leq 1$ ) to be used in painting the (icon of the) annotation, but does apply to the pop-up window. The default is 1.0.

Adobe Distiller handles the opacity for us in all cases except when we create a (dynamic) stamp. If an opacity value less than 1 is desired, special techniques are needed, and the file needs to be distilled using the **Standard\_transparency** job options.

The following key-values are  $\text{\TeX}$  based concerning placement of the annotation in the margin.

- `margin`: Use this key (it has no value), to declare that you want the annotation to appear in the margin. The `\marginpar` command from core  $\text{\TeX}$  is used, the placement of the annotation follows the rules set down by  $\text{\TeX}$ . You can reverse the placement of the annotation by using the  $\text{\TeX}$  command `\reversemarginpar` (annots placed in the right margin, and now placed in the left); you can return to the default by using `\normalmarginpar`.

Given that you have use the `margin` key, there is an associated key that can be used, as well as a command.

- `margintext`: The value of this key is text that will be typeset just below the annotation icon.
- `\marginpartextformat`: A  $\text{\TeX}$  command for formatting the text in the margin, the default definition is

```
\margintextformat{\bfseries\tiny\color{blue}}
```

For an annotation placed in the margin with margin text, you might want to use the `readonly` key, this prevents the user—even one using Acrobat—from moving the annotation away from its caption.

The following key-value is for your convenience.

- `presets`: A key to allow the introduction of pre-defined options, for example, you might like all your comment annotations to be red, so you can define

```
\def\myComments{type=text,name=Comment,color=red}
```

then say

```
\annotpro[presets=\myComment]{Way to go!}
```

Additional options may be included,

`\annotpro [presets=\myComment,margin]{Way to go!}`  
for example.

## 4.2. Key-values for text annotations

The position of the annotation is determined by its bounding rectangle; for the text annotation, an icon is placed so that its upper left corner is at the upper left corner of the bounding rectangle. The icons themselves have certain dimensions that have been recorded within the `annot_pro` package, so you need not worry about leaving space for them.

An important fact about the icons used by text annotation is that they *do not zoom in or out* as the page magnification is changed; furthermore, the graphics commands `\scalebox` and `\resizebox` do not rescale the icons as expected.

The following are options specific to the text annotation. Recall, the text annotation is of `type=text`.

- `name`: The name of the icon to use when displaying this annotation in closed form (no pop-up window visible). Possible values—as specified in the *PDF Reference*—are `Comment`, `Key`, `Note`, `Help`, `NewParagraph`, `Paragraph`, and `Insert`. Additional icons that are available in recent versions of Adobe Reader are `Check`, `Circle`, `Cross`, `Star`, `RightArrow`, `RightPointer`, `UpArrow`, and `UpLeftArrow`.

If you are using comments in your document, and your audience have older versions of Adobe Reader, it is best to use only the seven listed in the *PDF Reference*.

- `open`: A Boolean value that determines if the pop-up window is open or not. The package default is `false`. You can use `\setAnnotOptions` to set this option globally.
- `nohspace`, `novspace`, `nospace`: The presence of these commands zeroes out the dimension(s) of the bounding rectangle. Specifying `nohspace` as an option causes the icon to take up no horizontal space as the page is latexed. Here  is an example of a sticky note with the `nohspace` option.

Without any of these three keys, the text annotation  fits exactly into the allotted space at 100% magnification, try it.

When the icon takes up no TeX space, it may cover content on the page, as it does above. Acrobat users can move the icon around, but AR users cannot



move the icon. The pop-up window is movable and scalable, but the icon cannot be moved. Therefore, you must be careful about placement.

Additional positioning of the icons can be made using standard  $\TeX$  commands such as `\raisebox` and `\smash`. For example, the blue icon above was created by

```
For example, \smash{\raisebox{1in}\raisebox{1in}{%
  \annotpro[nospace,color=blue,
    opacity=.25]{...clever message...}}}
```

Note that I've set the `opacity` to `.25`.

### 4.3. Key-values for stamp annotations

A stamp annotation is similar to a text annotation in the sense that it has a pop-up window in which the contents of the message is displayed; however, unlike the text annotation icons, the stamp appearance can be re-scaled using `\scalebox` and `\resizebox` of the `graphics` package; however, the keys `scale`, `widthTo`, and `heightTo`, as described below, are the recommended methods of re-scaling a stamp.

The following are the key-values associated with this annotation type.

- **name**: The stamps listed in the *PDF Reference* are `Approved`, `AsIs`, `Confidential`, `Departmental`, `Draft`, `Experimental`, `Expired`, `Final`, `ForComment`, `ForPublicRelease`, `NotApproved`, `NotForPublicRelease`, `Sold`, and `TopSecret`. If `name` is not specified, `Draft` is the default.

There are other stamps, not listed in the *PDF Reference*, but available in more recent versions of Acrobat. The file `stamps.pdf` lists all the stamps that I have access to. The names of these other stamps are recognized by the `name` key.

The dimensions of the stamps listed above are all the same, they are 150bp width and 40bp high.

Additional stamps are shipped by Acrobat, a listing of these appears in the file `stamps.pdf` (`stamps.tex`).

- **width**, **height**: If the value of `name` key is something other than one of the stamps listed above (one of the stamps listed in the file `stamps.pdf`), the width and height are not known to `annot_pro`. In this case, use the

height and width keys to set the dimensions of the bounding box. Adobe Distiller will resize the stamp to the stamp is the largest one that can fit in the bounding box, the stamp will be centered vertically and horizontally within the bounding box.

Here are a couple of examples, the bounding box is should as a black \fbox.

- This one  fits more or less exactly. I determined the dimensions of this stamp through some controls of the user

interface. Contrast this this stamp  obtained by setting

width=50bp and height=50bp. Notice the "best fit," and that the bounding box takes up space. We can use \smash to smash its vertical height, let's see how that looks, 

The code for producing these stamps are given in the margins.

You can resize these stamps using \scalebox and \resizebox like so.  The code for producing this stamp are given here 

- To create a stamp that takes up position against \smash and \makebox [O t] [l r] {text}; here is one of the standard stamps as listed in the PDF.  The code for this stamp is given in this note .

Try changing the magnification of the PDF page, you'll see that stamps are re-scaled as you zoom in or out, while the text annotations are not. I don't like text annotations for this reason.

- rotate: Stamps can be rotated, use this key to enter an angle of rotation, for example, rotate=30 rotates the stamp 30° counter clock-wise.
- 👉 Do not use the \rotatebox command of graphics package, this command *does not rotate* the stamp.
- scale: Use this key to re-scale the stamp; the value of this key is a number between 0 and 1. For example, scale=.5 reduces both width and height in half.
- widthTo: This key resizes the stamp so that the width is the value of this key; for example, widthTo=2in re-scales the stamp to have a 2in width.



- `heightTo`: This key resizes the stamp so that the height is the value of this key; for example, `heightTo=2in` re-scales the stamp to have a 2in height. Only one of the keys `scale`, `widthTo`, and `heightTo` are recognized for any stamp annotation. If all three are specified, only `scale` is used. If `widthTo` and `heightTo` are both specified, then `widthTo` is used.
- ☞ The use of these keys is the recommend way of re-scaling a stamp. These methods are compatible with the `rotate` key.
- `customStamp`: You can create a custom stamp from any image you wish to use, here is  one such example. The code for the above stamp is given here .
- `ap`: When the `customStamp` key is used, you have to supply an indirect reference to the appearance of the stamp. This reference is made through the `ap` key. The example above demonstrates the use of the `ap` key.

**Important:** When using the stamps of Acrobat always perform a SaveAs on the file when you have finished building the file. This imports the appearances of the stamps into the document and saves them.

- ▶ The creation of a custom stamp requires a details list of steps, at some point I'll write a white paper on the subject.

Here's an example of the keys `rotate` and `widthTo`:



#### 4.4. Key-values file attachment annotations

The file attachment annotation has no pop-up window, the value of the required parameter is used as a description of the attached file, and appears in the file attachment window.

The key-value pairs special to this type of annotation are as follows.

- `name`: The name of the icon to use, permitted values are `Graph`, `Paperclip`, `PushPin`, and `Tag`. If the value of `name` is not specified, the default is `PushPin`. The `annot_pro` package knows the dimensions of each of these icons, so you need not worry about them. They can be re-scale using standard commands from the `graphics` package, though, there may be little need of doing so.
- `file`: The value of the `file` key is the *absolute path* to the file to be attached. I've devised a helper command `\defineAPath` that can be used to define the path to your file. For example, we can define a path to wherever the files are, like so

```
\defineAPath{\graphicsPath}
  {C:/AcroTeX/acrotex/aepro/annot_pro/examples/graphics}
```

The command takes two arguments, the name of the command to be defined, and the path.

We can create a file attachment  like so, ho, ho, ho. Here is the code for this file attachment . Clicking the file attachment icon will open the file, in recent versions of Acrobat, the file is listed in the file attachments panel. Open it using the user interface, and you'll see the file listed, as well as a description, as passed to the `annot` as the second argument of `\annotpro`.

The file attachment icons can be resized using any of the `graphics` commands, `\scalebox` or `\resizebox`.

#### 4.5. Setting Global Options with `\setAnnotOptions`

Global options are by using the `\setAnnotOptions` command. In the preamble of this document I placed

```
\setAnnotOptions{subject={AcroTeX Communiqu\'e},title={D.P. Story}}
```

That way, I didn't have to constantly type in my personal name for each example. These options can be overwritten by specifying options locally, if I say, `\annotpro[author=Don Story]{Hi there!}`, the author is now my alter ego, Don Story.

That's all for now, I simply must get back to my retirement. 