New Technical Notes
Macintosh

®

Developer Support

The Compleat Guide to TeachText
Platforms & Tools   **M.PT.TeachText**

Written by: Bryan Stearns, DTS Emeritus   April 1990

This Technical Note explains how to use TeachText to create release notes, complete with pictures, which every Macintosh owner can read. This Note assumes familiarity with ResEdit.

Background

TeachText is two, two, two applications in one, and Apple ships it with every Macintosh. It's a simple text editing training tool with support for the standard editing primitives, saving and printing, and it's also a tool which allows every Macintosh owner to browse read-only release notes or other documents which may contain text and pictures.
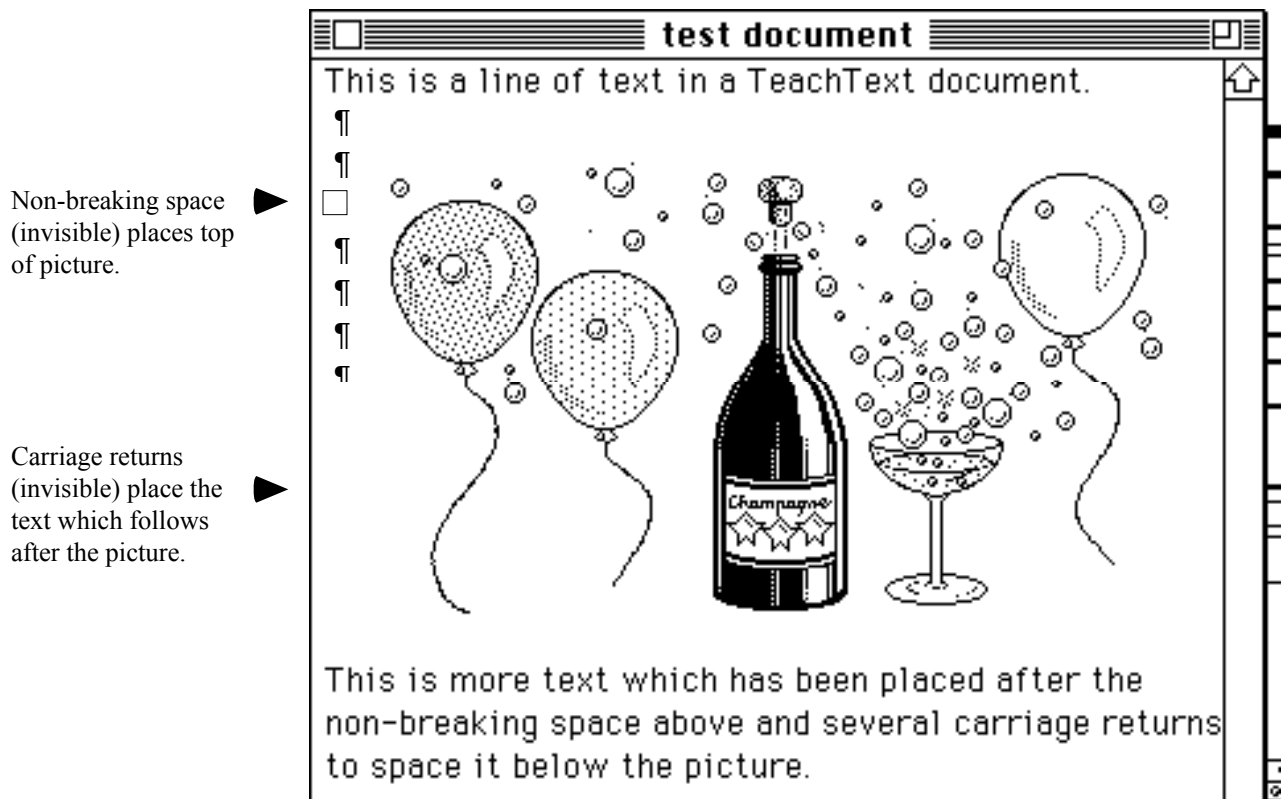
Since TeachText only allows a single open document at a time, it uses the document's file type to determine which of the two applications it should be. If the file type is "TEXT" (as are all files created by TeachText), it operates as a simple text editor, but if the file type is "ttro" (lowercase is significant), it only allows the user to scroll through the document or print its contents—modifications are not allowed, thus making the file read-only.

How TeachText Handle Pictures

TeachText operates on documents of the two file types previously described, and either may contain pictures. However, pictures tend to disappear when editing the document in which they are contained (to those hardy souls attempting to create documents with pictures who must put up with this during the creation process, my apologies), thus all documents which contain pictures should be distributed as read-only (i.e., file type "ttro").

A document's pictures are stored as purgeable `'PICT'` resources in the resource fork of the document. Whenever a file is opened, each of these picture resources is loaded in numerical order, and its size is read into an array (so TeachText can later test to see if a picture needs to be drawn into the window without loading the picture). After the picture resources are loaded (and every time the window is resized thereafter), TeachText scans the text of the document for non-breaking space characters (ASCII $CA, entered as Option-Space Bar and

usually used instead of a space to prevent related words from being split across line boundaries). In TeachText documents, a non-breaking space character represents the line on which the top of a picture resides. Figure 1 illustrates this relationship.

This is a line of text in a TeachText document.

¶
¶
☐ — Non-breaking space (invisible) places top of picture.
¶
¶
¶
¶

— Carriage returns (invisible) place the text which follows after the picture.

This is more text which has been placed after the non-breaking space above and several carriage returns to space it below the picture.

Figure 1–Picture With Non-Breaking Space and Surrounding Text

If there are more non-breaking space characters than `'PICT'` resources, TeachText ignores the extra non-breaking spaces. Likewise, if there are more `'PICT'` resources than non-breaking space characters, TeachText ignores the extra `'PICT'` resources. Every time an update event occurs, TeachText checks each picture in the array, and if any of the pictures in the array overlap the current update region, it draws that picture.

As it happens, TextEdit is particularly messy about redrawing large portions of the screen when a user is entering text, and this makes editing documents with pictures rather clumsy. Since resizing the window causes another scan for non-breaking space characters as well as an update event, sizing the window in any way causes TeachText to "refresh" the pictures.

Creating Release Notes With TeachText

So how does one use TeachText to create release notes? It's easy. Get those creative juices flowing, grab a cup of strong coffee (or your favorite highly-caffeinated beverage), and read on.

Write the Text

You can handle this part yourself. Use any word processor or text editor that supports saving to text-only files (i.e., those files of type "TEXT"). You can even use TeachText if you so desire. Don't worry about fonts or styles, since TeachText only gives you the default application font in plain style. Don't put carriage returns after each line either, since TeachText automatically wraps lines, just like a real word processor (the TeachText window

conforms to the size of the current screen, so don't depend on the breaks you see either). Don't worry

about non-breaking space characters at this point either; you'll get a chance to add them later. Just think about what pictures you want (if you want them at all) and in what order you want them. When you are finished with the text, save a text-only file. If your word processor gives you the option of putting carriage returns after lines or after paragraphs, choose the after paragraphs option.

Draw the Pictures

First make a backup of your Scrapbook file (you should find it in your System Folder) if it contains anything you consider important. After backing it up, throw away the original copy (this makes things much easier later on in the process), but don't worry, if you made a backup you can use it to restore the original when finished. If you prefer, you can just rename the Scrapbook file, which effectively makes a backup copy.

Unfortunately, the ideal method for creating a picture involves both a paint program and a draw program. In addition, you should use Geneva 12 point font (or another System font like Monaco 9, Chicago 12, or Geneva 9) in your picture since that is the font that the rest of the text in the TeachText document uses. Once you are finished with your pictures, save them to a document, then do one of the following:

1. If you used a painting program to draw your pictures:

    Select your picture with a Lasso tool to ensure that only the minimum size of the image is copied. This takes up less space on disk and centers the picture in the document better for the user. Copy the picture then paste it into the Scrapbook. Repeat these steps for each individual picture you wish to include in the document.

2. If you used a draw program to draw your pictures:

    Copy each of your pictures into the Scrapbook. Launch a paint program, then copy each picture from the Scrapbook into the paint program. Once every picture is in a paint document, open the Scrapbook and clear each of your pictures from the Scrapbook. The Scrapbook should say "Empty Scrapbook" when you are finished (unless you did not start with a fresh Scrapbook). Now follow the procedure in the steps for a painting program to copy and paste each of your pictures back into the Scrapbook.

At this point, regardless of which program you originally used to create your pictures, they should all be in the Scrapbook and in bitmap form (after being copied with a Lasso tool from a paint program).

Because of a quirk in the Printing Manager and PostScript®, you have to perform one more step. Launch a draw program, then copy each picture from the Scrapbook into the draw program. Once every picture is in a draw document, open the Scrapbook and clear each of your pictures from the Scrapbook. The Scrapbook should say "Empty Scrapbook" when you are finished (unless you did not start with a fresh Scrapbook). Now copy each picture back to the Scrapbook. This process makes the pictures "transparent" when printed, and this is important to avoid a problem with white, horizontal stripes running through your pictures.

Add the Pictures

Now to add the pictures to the TeachText document. Launch ResEdit and open the text-only TeachText document (you may want to work on a backup copy). ResEdit may warn you that

the file does not have a resource fork and opening it will create one. This is fine, since you want a resource fork. If ResEdit does not warn you, then the file already has a resource fork (this means that there may already be resources there).

If the ResEdit window you get (whose title is that of the document name) contains any four-letter words (no, not those four-letter words, but words like `'MPSR'`, `'ETAB'`, etc.) other than `'PICT'`, then you should select them and clear them from the document. If you have already added some pictures to this file (and are replacing some of them), you should be especially careful, since it is easy to accidently delete the wrong one.

Now open your Scrapbook file (the one with all the pictures in it). Its ResEdit window should contain `'PICT'`, `'SMAP'`, and `'vers'` resources. Select `'PICT'` (don't double-click), and copy this resource to the TeachText document by bringing it's window to the front and selecting Paste from the Edit menu. If you do this step correctly, your pictures and text should all be in the same document. Now save the TeachText document so you don't have to do this step again and close the Scrapbook.

Now you need to put the pictures into the proper numerical order so they show up in the correct order in the TeachText document. Numbering starts at 1000 (i.e., first picture should be 1000, second picture 1001, etc.). To order these pictures, double-click on the `'PICT'` in the TeachText document's window. You should get another window which contains each of the pictures you copied into this document. Use the scroll bar until you find the first picture you want to appear in the document. Select it (by clicking on it once), and choose the Get Info or Get Resource Info option to get information on the resource. ResEdit displays an information window about the selected resource with space to enter a name and an ID (there is already a random ID number assigned). Change the ID to 1000 and give the picture a name too (i.e., "Figure 1", etc.). Near the bottom of this window you can see the resource attributes. Be sure that the "Purgeable" attribute is checked, then close the window. Repeat this process for each succeeding picture, giving each a successive number (i.e., 1001, 1002, 1003, etc.). When you are finished with all of the pictures, save the file and quit ResEdit.

That is the difficult part; the rest is icing. Go get some more coffee or whatever it is you are drinking.

Edit the Text to Make It Look Pretty With the Pictures

Launch TeachText and open your document. Find the location where you want to place the first picture and position the text cursor there. Enter a carriage return or two (more if you want more space before the picture) then a non-breaking space character (Option-Space Bar, remember), which will be invisible. Now resize the window, and *voilà*, when the window redraws, your picture will be just below the non-breaking space character. Now enter as many carriage returns as necessary to provide space for the picture. When you enter the first carriage return, TeachText will erase the picture, so you will need to resize the window again to verify your spacing.

Once you have enough room for the first picture (you probably want to leave an extra blank line or two after it too), move on to the next desired picture location and repeat the process. Continue this process (and don't forget to save the document along the way) until you have placed all of the pictures. When you finish placing the pictures, you should save the document again and try printing it on both an ImageWriter and LaserWriter if possible. You may wish to tweak the picture spacing or location to keep them from crossing printed-page boundaries.

When you are satisfied with the results, Quit TeachText.

Make the File Read-Only

Make a copy of the file (to save a step if you decide to edit it again) then launch ResEdit. Now choose Get Info from the File menu and change the file type from "TEXT" to "ttro" (the lowercase is significant) and check to make sure the creator type is "ttxt". Now quit ResEdit and save the changes to the document when prompted.

That's all there is to it. (Now that wasn't that bad, was it?)

A Few Hints On Creating Good Documents With Pictures

The following hints should help to make your TeachText document creation faster and more efficient as well as make the final document as nice as possible for the user.

- Always use the Lasso tool in paint programs to select pictures to appear in TeachText documents; it makes them smaller.

- Keep pictures as small and simple as possible; the document takes up less room on disk and scrolling is faster.

- If two pictures appear on top of each other, you probably have two non-breaking space characters on the same line. Simply delete one to fix it. It is generally a good idea to put non-breaking space characters on a line by themselves with a blank line before it. In addition, always leave room for an extra line after the picture so you do not have the picture running into the text which follows it.

- If you need to use the non-breaking space character as a non-breaking space, you can. Since TeachText assigns the numbered `'PICT'` resources to the non-breaking space characters in the document, you can simply skip a resource number to use a non-breaking space character as a non-breaking space in the text. For example, if you had four non-breaking spaces in the document and you wanted pictures at the first, second, and fourth, you would number your `'PICT'` resources 1000, 1001, and 1003. The third non-breaking space character would normally have `'PICT'` resource 1002 assigned to it, but since there is not a resource with this ID, it simply acts as a non-breaking space in the document.

- Do not worry about how horrible everything looks when you are editing; users will not be able to edit your document (unless they have read this Note), so they will not see the awful flashing, disappearing pictures, etc.

- Make the document read-only even if you do not use pictures. Distributing read-only documents to users gives the consistent impression that Release Notes are not to be modified.

- If your pictures are not appearing as you think they should, and if you cannot figure out what might be wrong by following the sequence in this Note, then try the following: Open the document with ResEdit. Click once on the `'PICT'` list and choose Open General from the File menu of ResEdit 1.x. You should get a window with a list of all of your pictures, in order, and numbered sequentially from 1000. If this is not what you get, then you have missed a step along the way and need to make sure all your pictures are in the resource and numbered sequentially.

Further Reference:

- *Macintosh ResEdit Reference*

PostScript is a registered trademark of Adobe Systems, Incorporated