How to install JavaScript support files

D. P. Story

Last Revision Date: June 18, 2021   Version 1.0

1. Introduction

Because some of the Acrobat JavaScript methods have security restrictions, they can be only executed in the console window, or as part of a batch sequence, or from the JavaScript folder reserved by Acrobat for that use. The developer (of LaTeX packages or PDF form documents) may want to use these restricted JavaScript methods as part of document assembly. To use the restricted methods, they must be called from the *user JavaScript folder.* The scripts are contained in a JS file that Acrobat reads when it first starts up.

In this article, we provide detailed steps for locating the user JavaScript folder of Acrobat and for installing JavaScript files in that folder. Various packages distributed by package author D. P. Story (notably acrotex, aeb_pro, thorshammer, and docassembly), use the JavaScript files aeb.js and aeb_pro.js, which are placed in the user JavaScript folder. It is important to correctly install these files for the packages listed above to function as expected.

Other JavaScript files may be placed in the folder as well, for example, the config.js file is one that is commonly used for general purposes.

Before we get started, if you are using a version of Acrobat prior to version 8.1, you need not read any further. The security restrictions described below were not in effect for those versions.

2. Installing the JavaScript support files

The functionality of AeB (acrotex) and AeB Pro (aeb_pro, as well as other packages, depends, to some extent, on the proper installation of their JavaScript support files.

- AeB: For authors using the dvips + Acrobat Distiller workflow, aeb.js is required; this file is not used by pdflatex or xelatex users.

- AeB Pro: For authors using the dvips + Acrobat Distiller workflow, aeb_pro.js is required; for authors that prefer pdflatex, lualatex, or xelatex as the PDF creator *and who own* the Acrobat application, the aeb_pro.js is required with the useacrobat option of AeB Pro. When using the nopro option of AeB Pro, the JavaScript file aeb_pro.js is not used.

- For thorshammer and docassembly, aeb_pro.js is required to be installed.

The placement of the JavaScript files has changed over the years and is a function of the version number ⟨*ver*⟩. Start Acrobat and open the console debugger window by pressing Ctrl+J (for Windows OS) or Cmd+J (for Mac OS). Now execute the code app.getPath("user") in the console window.

To do this, press the getPath button provided below, which copies the necessary code to the console window.

$$\text{app.getPath("user");} \tag{1}$$

With the cursor on the line containing the script, press the Ctrl+Enter (for Windows OS) or Cmd+Enter (for Mac OS) key. This will execute the JavaScript and return the 'root' path to the user's application data for Acrobat.

The rest of the instructions depend on your operating system, Windows or Mac.

2.1. For users of Windows OS

There are several cases, jump to the numbered paragraph that describes your version (⟨*ver*⟩) of Acrobat.

❶ **For ⟨*ver*⟩, 8.1 ≤ ⟨*ver*⟩ < 10.1.1.** For the Windows OS, executing the script in display (1) returns a string of the form:[1]

$$\text{/C/Users/⟨\textit{username}⟩/AppData/Roaming/Adobe/Acrobat/⟨\textit{ver}⟩} \tag{2}$$

Navigate to this folder location. The correct location for the JS files is in the JavaScripts subfolder. If the folder does not exist, create it. The final path for the JavaScript support files is then

```
/C/Users/⟨username⟩/AppData/Roaming/Adobe/
     Acrobat/⟨ver⟩/JavaScripts
```

Finally, copy aeb.js (if applicable) and/or aeb_pro.js (if applicable) into this folder. Close Acrobat, the next time Acrobat is started, it will read in the JS files.

❷ **For ⟨*ver*⟩, 10.1.1 ≤ ⟨*ver*⟩ < 12.** For these versions of Acrobat security has have tightened up even more; the JavaScripts folder is now a subfolder of a special Privileged folder. Execute app.getPath("user") in the console window, as in display (1). The method brings back a string similar to the one in display (2). The correct path is obtained by simply inserting Privileged into the path in display (2), between Acrobat and ⟨*ver*⟩, as displayed below.

```
/C/Users/⟨username⟩/AppData/Roaming/Adobe/
     Acrobat/Privileged/⟨ver⟩/JavaScripts
```

But this is a mere description of the correct location, the path must exist! The path must be created, if it does not already exist.

Navigate to the folder, one folder above the path shown in display (2), that is, navigate to,

```
/C/Users/⟨username⟩/AppData/Roaming/Adobe/Acrobat/
```

---

[1] This string is for Windows OS beyond WinXP.

If a `Privileged` folder exists, enter into it; otherwise, create it, then enter it.

> `/C/Users/⟨username⟩/AppData/Roaming/Adobe/Acrobat/Privileged`

Now create a folder named ⟨*ver*⟩, unless it already exists, and enter it. The version is always the major release 8.0, 9.0, 10.0, or 11.0.

> `/C/Users/⟨username⟩/AppData/Roaming/Adobe/`
>   `Acrobat/Privileged/⟨ver⟩`

Finally, create the `JavaScript` folder, unless it already exists, to obtain the final path to the user folder JavaScripts:

> `/C/Users/⟨username⟩/AppData/Roaming/Adobe/`
>   `Acrobat/Privileged/⟨ver⟩/JavaScripts`

Copy `aeb.js` (if applicable) and/or `aeb_pro.js` (if applicable) into this folder. Close Acrobat, the next time Acrobat is started, it will read in the JS files.

❸ **For ⟨*ver*⟩, ⟨*ver*⟩ > 12.** Beginning with the next major version following version 11, Adobe changed its versioning scheme. There are now two tracks, *subscription* and *classic* (also called *perpetual*). For subscription, the version, as I understand it, will always be DC (Document Cloud), and for the classic, for which you've purchased a *perpetual license*, the ⟨*ver*⟩ is a calendar year. The first classic version is 2015, the next one is most likely 2017, since major releases occur about every two years.

    Execute the code of display (1) by pressing the button provided to obtain a path of the form,

> `/C/Users/⟨username⟩/AppData/Roaming/Adobe/Acrobat/DC`

for the subscription license or,

> `/C/Users/⟨username⟩/AppData/Roaming/Adobe/Acrobat/2015`

for the classic (or perpetual) license. Of course, this last path is for the first classic version.

    Now, to obtain the path to the `JavaScripts` folder, follow the instructions for numbered paragraph ❷, but with ⟨*ver*⟩ equal to either DC or 2015 (a calendar year).

2.2. For users of Mac OS

Mac OS does not have the security restrictions that Windows OS does, as described in paragraphs ❷ and ❸.

    Follow the instructions of paragraph ❶, by first pressing the button provided in display (1), then, in the console window, execute this script as described in the paragraph following display (1). After working through ❶, one obtains any of the following paths depending on the version (⟨*ver*⟩) you are using:

For ⟨*ver*⟩ = 8.0, 9.0, 10.0, 11.0:

```
/Users/⟨user⟩/Library/Application\ Support/Adobe/
    Acrobat/⟨ver⟩/JavaScripts
```

For $\langle ver \rangle > 12$, the path is,

```
/Users/⟨user⟩/Library/Application\ Support/Adobe/
    Acrobat/DC/JavaScripts
```

for a subscription license, and,

```
/Users/⟨user⟩/Library/Application\ Support/Adobe/
    Acrobat/2015/JavaScripts
```

or more generally,

```
/Users/⟨user⟩/Library/Application\ Support/Adobe/
    Acrobat/⟨year⟩/JavaScripts
```

for the perpetual license.

In all cases, if the JavaScripts folder does not exist, it must be created. Copy any JS support files into the JavaScripts folder. The script files are read the next time Acrobat is started.

3. Testing the placement of the JavaScript support files

Assuming you have successfully followed the instructions and created the JavaScripts folder, test the validity of what you have done by closing Acrobat then opening it again. Open the console window (Ctrl+J or Cmd+J, for Mac OS) and execute this command in the console window:

```
app.getPath("user","javascript");
```

If you have followed the instructions, the return string should read the complete path to the JavaScripts folder.