Using TeXShop, LaTeX and ConTeXt Simultaneously

Version 4.65 of TeXShop introduced several changes, so a summary explaining how to use them is in order. In this section, we suppose that TeXShop has been used to typeset LaTeX documents using pdflatex, xelatex, or lualatex and we want to add ConTeXt to the mix. ConTeXt is in TeX Live, but that version tends to be out-of-date because TeX Live is only updated once a year. It is possible to get the latest ConTeXt by going to the ConTeXt Garden Wiki at https://wiki.contextgarden.net/Main_Page and installing the version available there for Intel Macs, or the version for Arm Macs. This process is clearly described on the Wiki, and very easy. The distribution can be installed anywhere you like; a natural place is in your home directory at ~/bin/context. A simple command allows this distribution to be updated, say weekly.

If you install ConTeXt this way, go to TeXShop Preferences under the "Engine" tab and change the third item at the top, which is labeled "Alternate Path", to the path to ConTeXt binaries. If the binaries were installed in ~/bin/context, this path might be ~/bin/context/tex/texmf-osx-arm64/bin.

Your latex files will continue to work as always.

At the top of each ConTeXt source file, add the magic lines

```
% !TEX TS-program = ConTeXtTest
% !TEX useAlternatePath
% !TEX useConTeXtSyncParser
```

In the header of the ConTeXt document, add the command

```
\setupsynctex[state=start,method=min]
```

Then typesetting for the ConTeXt file will also work as usual, and syncTeX will work as usual. But this typesetting will use the new ConTeXt Garden distribution, and syncTeX will use new routines in ConTeXt rather than the usual syncTeX code.

Done.

Using TeXShop, LaTeX, and ConTeXt with an External Editor.

As before, install ConTeXt from the ConTeXt Garden Wiki and enter its binary path into the "Alternate Path" Preference item in TeXShop. Your LaTeX files will continue to work unchanged. TeXShop does not open the source file when opening a project in external editor mode, so information in magic lines is not read. Therefore, if you want to use a ConTeXt source with an external editor, you must tell TeXShop to use the alternate binary path to the ConTeXt binaries. Here is a short applescript macro which does that and simultaneously tells TeXShop to use the new ConTeXt synctex methods:

```
--Applescript direct

set frontName to #DOCUMENTNAME#
\tell document frontName of application "TeXShop"
    AlternateBinPath
    SyncConTeXt
```

```
     end tell
```

Enter this macro using TeXShop's Macro Editor. Recall that macros appear in the menu bar even in external editor mode. Execute the macro just once as soon as the ConTeXt source is loaded. Then TeXShop will call the correct version of ConTeXt when it typesets.

Now we want to activate two additional features. We want "Goto Error" and "Sync from Preview to Source" in TeXShop to go to the appropriate line in the external editor, and perhaps open the appropriate source file if it is not open. And we want the editor to have a "Sync from Source to Preview" command, which takes us back to the appropriate location in TeXShop's Preview of the Document. This command should be activated by clicking at a spot in the source text while holding down the command key.

These extra features require additional code in TeXShop, which version 4.65 provides, and additional code in the Editor, which must be supplied by the author of the editor. Both Sync operations will be performed by TeXShop, so the extra features required for the external editor are extremely modest. The commands "Goto Error" and "Sync from Preview to Source" were provided by version 4.24 of TeXShop. We first explain some changes to simplify what was said there.

One important change is that we earlier recommended setting two hidden preference items:

```
     defaults write TeXShop TextMateSync YES
     defaults write TeXShop OtherEditorSync YES
```

This is no longer necessary. Instead the necessary commands to TeXShop are provided by AppleScript.

Version 4.65 of TeXShop provides eight additional applescript commands. These commands are only needed in external editor mode. Here are the commands:

```
     SyncRegular
     SyncConTeXt
     SyncWithTextMate
     SyncWithOtherEditor
     AlternateBinPath
     StandardBinPath
     SyncRootName
```

The first command is rarely needed. If the user is working on a ConTeXt document, the second command says to switch to ConTeXt Sync.

The third command says that the external editor is TextMate. The fourth command says that it is some other editor.

The fifth and sixth commands have already been introduced, and tell TeXShop to use the Alternate Binary Path or the usual binary path /Library/TeX/texbin.

TextMate installs a command line program in /usr/local/bin which is able to control some of the features of the editor. For instance, the command /usr/local/bin/mate --line 50 /Users/koch/Syllabus.tex opens Syllabus.tex in TextMate (if not already open) and highlights line number 50. This makes it easy to support syncTeX from TeXShop to TextMate because the standard TeXShop routine can be used until the last moment when a line in the Source Window would be highlighted; this final step can be replaced by a call to "mate".

Thus if you use TextMate, configuration is done as soon as you tell TextMate to install the mate program in /usr/local/bin. TeXShop 4.24 will sync from the Preview window to TextMate in the standard way: click on a spot in the pdf display while holding down the command key. As a bonus, Goto Error works from the Console window if the user is using TeXShop to typeset. Click this button to be taken to the first error, and click again and again to cycle through the first few errors. If you called SyncConTeXt, this sync operation will use ConTeXt. Note once more that TeXShop must be in "external editor" mode to use these features.

If you use some other editor, you must create an executable shell script in /usr/local/bin named "othereditor". When TeXShop calls this script, it sends two parameters to the call: $1 contains the line number (as a string), and $2 contains a full path to the desired tex source file. The othereditor script should then call the external editor asking it to open the desired source if it is not already open, and move to the appropriate line. For some editors this may already be implemented; otherwise it may be necessary to bribe the author.

To show how this works in detail, we will assume that TextMate is not built into TeXShop and handle it using an othereditor script. And we will write an othereditor script for BBEdit, whose authors have provided the necessary code in their application.

For TextMate:

```
#!/bin/sh
/usr/local/bin/mate --line "$1" "$2"
```

For BBEdit:

```
#!/bin/sh
/usr/local/bin/bbedit "$2:$1"
```

One final thing. Rather than using a TeXShop Macro to call the Applescript commands listed above, it is possible to create shell scripts which call these commands, and then call then using a Terminal command. For example, the shell script below calls SyncConTeXt:

```
#!/bin/bash

osascript  <<EOD

tell application "TeXShop"
    set the front_document to the front document
    tell front_document
          AlternateBinPath
          SyncConTeXt
          return 0
    end tell
end tell
EOD
```

Syncing from an External Editor to the Preview Window

When using an External Editor, we would like to click at a spot in the source file while holding down the command key and then call TeXShop to display the corresponding spot in the pdf output. This will require additional code in the external editor\. When the spot is clicked, the external editor needs to determine the line clicked, the index of the click point (that is, the number of characters in the line before the click), and the full path to the source file. Call these parameters "linenumber", "indexnumber", and "full path to source file". Each is a string; in particular the linenumber is a string like "51", not the integer 51.

The external editor then sends this information to TeXShop, which will handle processing the synctex file and highlighting the corresponding spot in the pdf. There are two ways to send this information. If the external editor supports AppleScript, it can execute the following AppleScript message:

```
tell application "TeXShop"
     activate
     set the front_document to the front document

     tell front_document
          sync_preview_line theLine "linenumber"
          sync_preview_line theIndex "indexnumber"
          sync_preview_line theName "full path to source file"
          return 0
     end tell
end tell
```

If the external editor can call a shell script, the AppleScript message above can be embedded in such a script. Create the following file

```
 #!/bin/bash

MyShellVar=$1
MyShellVas=$2
MyShellVat=$3

osascript  <<EOD

tell application "TeXShop"

   activate
   set the front_document to the front document
   set MyAppVar to $MyShellVar
   set MyAppVas to $MyShellVas
   set MyAppVat to "$MyShellVat"

   tell front_document
           sync_preview_line theLine MyAppVar
           sync_preview_index theIndex MyAppVas
           sync_preview_name theName MyAppVat
           return 0
   end tell

end tell
EOD
```

Name this file something appropriate, like "CallTeXShopSync", set its execute bit, and place it in /usr/local/bin. Then the external editor can communicate via

```
CallTeXShopSync "linenumber" "indexnumber" "full path to source file"
```

All of this works both with ordinary pdflatex, xelatex, or lualatex sources and with ConTeXt sources. It is useful to know more about these AppleScript calls. When I wrote them long ago, I couldn't figure out how to send more than one parameter to an AppleScript command. So I used a kludge. The command "sync_preview_line" sends "linenumber" to TeXShop, where it is remembered in an instance variable. The command "sync_preview_index" sends "indexnumber" to TeXShop in the same way. The command "sync_preview_name" sends "full path to source file" to TeXShop **and then** calls an internal procedure in TeXShop to finish the sync operation. So while it looks like we are just passing in three parameters, the final step uses these parameters to complete the task.

Incidentally, the "SyncConTeXt" command included in an earlier macro tells TeXShop to use the new ConTeXT Sync Methods when the internal procedures finishes the sync operation.

You may wonder why the applescript commands pass in the "full path to source code." TeXShop usually knows this information, which it learned when the file was opened in external editor mode. That was the "root document", which is the document typeset by TeXShop and displayed in the Preview window. But this root document may input other source files and the sync operation might have occurred by clicking on a line in the input file. The code which processes the synctex information needs to know which file the data came from.

It is possible to test this code before your external editor is modified, to see if that extra work is justified. For this testing, the case where additional source files are input is an added complication best ignored. Instead, we can let TeXShop just look for synctex information in the root file. That is the purpose of the final SyncRootName call. We can then write a shell script as follows:

```
  #!/bin/bash

     MyShellVar=$1
     MyShellVas=$2

     osascript  <<EOD

     tell application "TeXShop"

        activate
        set the front_document to the front document
        set MyAppVar to $MyShellVar
        set MyAppVas to $MyShellVas

        tell front_document
                sync_preview_line theLine MyAppVar
                sync_preview_index theIndex MyAppVas
                SyncRootName
                return 0
        end tell

     end tell
     EOD
```

Call this shell script "doSync", set its execute bit, and put it in /usr/local/bin.

To test, find a spot in the source and inspect to find its line number and index. A guess is usually good enough for the index. Suppose the line number is 71 and the index is 5. Open Terminal and issue the command

```
     doSync "71" "5"
```

If everything works, the correct spot in the Preview display should be marked by TeXShop.

Richard Koch
Department of Mathematics

University of Oregon
Eugene, Oregon 97403