TECHSHOW2018

Using Automation to Streamline Your Practice

WRITTEN BY:

Julianna Maria & Victor Medina

PRESENTERS:
Julianna Maria
Victor Medina:@victormedina

March 8, 2018

The Ghost in Your Machine: Document Automation on Your Macintosh Computer.

It has now been a substantial number of years since Macintosh users have been able to natively run HotDocs software. That doesn't mean, however, that we are without options for document automation. The purpose of this paper is to explore the currently available options, discuss their strengths and limitations, and discuss ways that these tools can assist today's practitioner in his or her day-to-day practice. Finding the ghost in your machine isn't hard. Various vendors offer a collection of tools to help you, and all of these tools are aimed at producing documents with a minimum amount of input from the user, which means not only efficiency, but less opportunity to make mistakes. In some circumstances, it can also mean that the attorney is freed up to hand tasks off to associates or paralegals who, armed with an automated template containing the more experienced lawyer's thought processes, can complete the task of document preparation largely on their own.

How to Automate a Document

In order for us to have a fruitful discussion, we need a glossary of sorts in order to make sure that were all using the same words the same way. We therefore begin with a brief discussion of database theory, because it is there that the source of most of the necessary vocabulary can be found.

Databases consist of a collection of "records." A simple example of a database with which all current attorneys are familiar (and our children and grandchildren are not) is the telephone book. Each page of the now largely defunct telephone book consists of hundreds of records. Those records consist of three pieces of information: a telephone user's name, their street address, and their telephone number. Database theory calls each of those discrete pieces of information a "field." In the case of our phone book example, one field holds the name, another holds the street address, and a third holds the telephone number.

Turning back to the telephone-book-as-database analogy, we have a database (the telephone directory) that consists of records (each listing) that are made up of fields (name, address, telephone number). And those basic statements are true about every database. Every database consists of a collection of records, and records consist of a collection of fields. The fields do not vary from record to record, but the contents of them do.

Your memory is filled with various forms of databases you have regularly encountered throughout your life. Another example, also alien to most of our kids, is the card catalog in the library. The big wooden cabinet containing all of those drawers of cards is itself a database. Each card is a record. And each record consists of the same kind of information, the exact contents of which vary from card to card: the author's name, the book's title, the publisher's name, year of publication, and so forth. Each of these things printed on the card are the content of the fields. Another example? Your personal address book. Little binder is a database, the repeating sets of boxes are records in the boxes themselves (name, address, telephone number, birthday, etc.) are all fields.

The singular advantage that a computer has when it comes to databases is that the computer can search, sort and manipulate that data far more efficiently than we can turn pages or finger through index cards.

Returning to our simple telephone book example, a slightly more advanced version would break the name field into two parts: one for the person's first name, and one for the person's last name. Why? Because doing so will give the user of that database more power and flexibility in searching, sorting, and controlling the way in which the data is presented, both on screen and on paper.

Once you begin to see the information around you in your practice as a collection of fields, you've made the first step in starting to automate your documents.

The second step in automating documents involves identifying a document you'd like to automate. It can be anything that you use on a recurring basis. The "recurring" element is extremely important. There's little point in doing all the work that we are going to describe below for something that you're never going to use again. But the document can be anything from a release, to an apartment lease, to a quit claim deed. Once you've settled on it, print the last one you used. Get out your highlighter, and highlight the content of that document that is unique to that particular version of the document. In the case of an apartment lease, the unique information will be things like the tenant's names, the number of their children, the amount of the monthly rent, the date of the lease, and the duration of the lease (probably in months). In the case of a litigation release, the unique information will be things like the plaintiff's name, the defendant's name, the amount of the settlement, the court in which the case is pending, the court number, and so on. In the case of a quit claim deed, the grantor, the grantee, legal description, the tax ID number, and the street address are among the unique elements.

When you're finished, the non-highlighted portions of your document should be the legal boilerplate that doesn't change much, if at all, each time you use this document. The highlighted portions of the document are the "fields" for the document.

Pro Tip: go back through your document and highlight pronouns and possessives. Him, his, her, she, it, their, them, etc. Get them all. All of those will vary from document to document, depending on what information is being used. Each one of those is also a field.

Now, make a list of each of the highlighted items, giving each one a succinct description (name, street address, telephone number, settlement amount, monthly rent, etc) and take care to ensure that you are not making any duplicate entries. By way of example, just because the plaintiff's name may appear more than once in the release, you don't want it to appear more than once on the list you are now making. In database language, what you are now doing is giving each field a name. Because we are later going to use those field names, this is a good time to be certain that your description is both brief and unique. Two words is often a good

goal, but not always achievable. "Unique" is absolutely critical while you cannot have two separate fields with the same name, you can use the same field more than once.

Using a contrasting pen, circle each highlighted portion of text in your document, and draw a line to the margin. At the end of that line, in the margin, write the field name from your list that describes the highlighted text.

When you have completed this task, you have the raw material for automating that document. The precise steps for automating it will now vary, based on the tool you select, but you now hold the key to that automated document.

The various tools available all have different methods of identifying what text in the document will always be in the document, and what text constitutes fields. You will need to use the appropriate field identification strategy for your tool in order for the document automation process to work. Some tools put field names in angle brackets, like this: <first name>. Others use chevrons: «first name». The precise way of signaling the beginning and end of the field name is unique to each program. Once you have learned the way your tool works, completing the automation of your document involves loading the digital version of the document you highlighted into your word processor and replacing the highlighted text with field names (including the correct characters indicating the start and end of each field name). From there, you will save the document as a master or template document and begin the process of testing. That is, you will now run your document automation tool, with this document, and produce sample output. Carefully inspect the results and find places where additional spaces are necessary or additional spaces can be deleted, verify that all of the field information was correctly inserted by the tool, and look for instances in which a field name remains in the document (perhaps, because you misspelled the field name). When you have successfully completed two or three practice runs with the document, you're finished.

Congratulations. You just automated your first document.

## Document Automation Option 1: MacSimplePrompter

If you decide that you're going to get serious about document automation, the most full-featured tool on the Macintosh right now is MacSimplePrompter, $299, http://www.wordsite.com/products/sp.htm. It's also the most expensive, but you get what you pay for.

MacSimplePrompter works exclusively with Microsoft Word and uses square brackets to mark field names, as shown in the following screen shot.
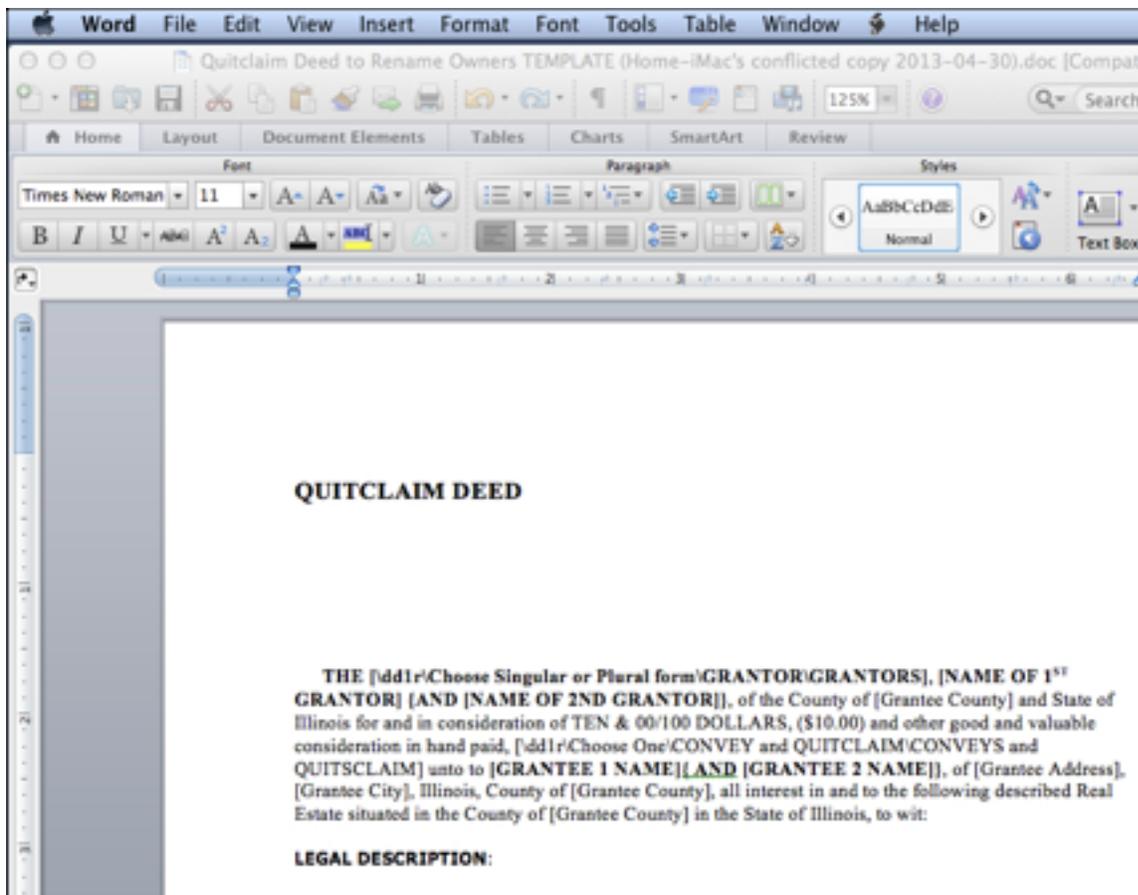
Simple fields are just words in square brackets, such as [GRANTEE 2 NAME] in the above example. Note, by the way, that this field name is in all capital letters and is bolded. When MacSimplePrompter works its magic, the resulting text will be formatted the same way. When the program is invoked, the user is prompted to provide information that then constitutes the user filling in the field. Simple fields generate a prompt like this:
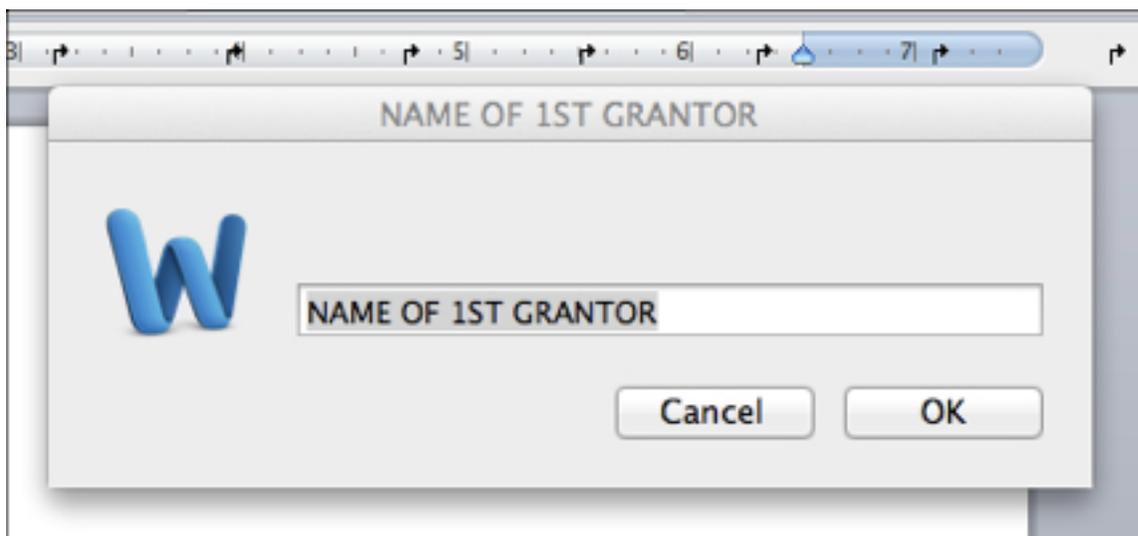
MacSimplePrompter also offers a list from which the user can choose, such as this one asking for which pronoun to use to describe the Grantor:
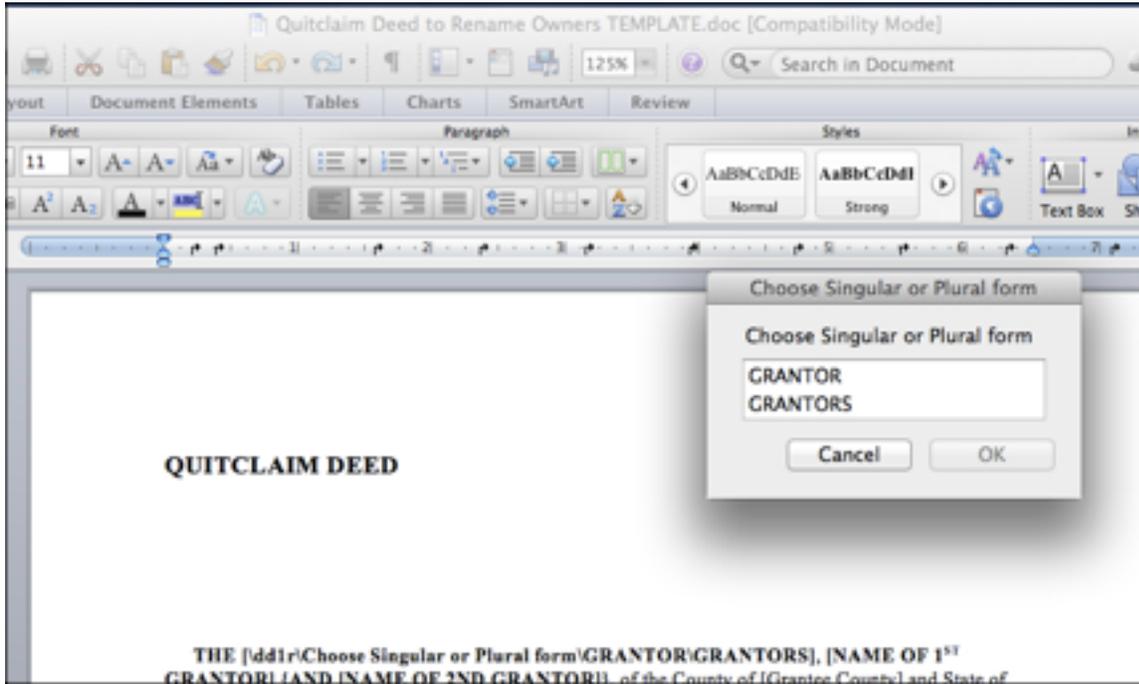
The program also allows optional inclusions, to address circumstances in which a field may not apply:
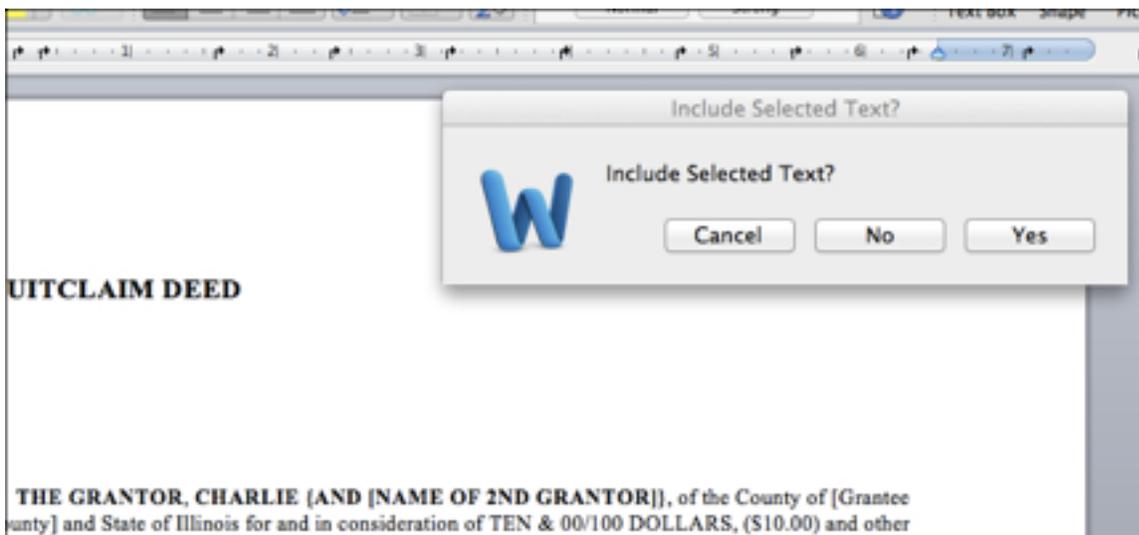
It also offers a large collection of other options, such as letting the user select multiple items from a list (even offering control over whether commas get inserted in between the selected items and whether the word "and" gets put in between each item, or only before the last one. It includes a boilerplate feature that, based on the user's answers to questions, can insert

paragraphs or entire pages of text at a time. There are also options for collecting date information from the user and reformatting it in several different ways.

MacSimplePrompter is a very powerful toolset that is not very difficult to learn to use. In fact, the process described above for identifying the fields in a document is more complicated then the process of using MacSimplePrompter to automate the document.

The developer, Bill Coan, is very responsive by email and offers a 30 day free trial.

Document Automation Option 2: TextExpander

A tool that no Mac using attorney should be without is TextExpander, $34.95, http://smilesoftware.com/TextExpander/index.html is a less expensive, but somewhat less full-featured option. TextExpander's principal use is to permit you to type just a couple of characters that the program then "expands" into something much larger. The things that get "expanded" are called "snippets." So typing "vjm" could cause the computer to immediately substitute "Victor J. Medina" and things like "ddate" can cause the computer to type out the entire date in whatever format you choose, such as "December 19, 2013" or "12/19/2013" or "Wednesday, December 19, 2013." You have control over just about everything. In David Sparks' excellent iBook, *Paperless*, he recommends using the date in YYY-MM-DD format as the first part of each file name. In Mark's office, they use a TextExpander item to do that for them, just by typing ".dt"

In recent updates, the program added the ability to prompt the user for information as part of the expansion, and that means you can automate certain documents using this $35 tool.

Below is a screenshot from TextExpander showing a notice of motion snippet. By simply typing the snippet "notmot2" Mark can produce the text of a notice of motion (meaning everything but the caption:
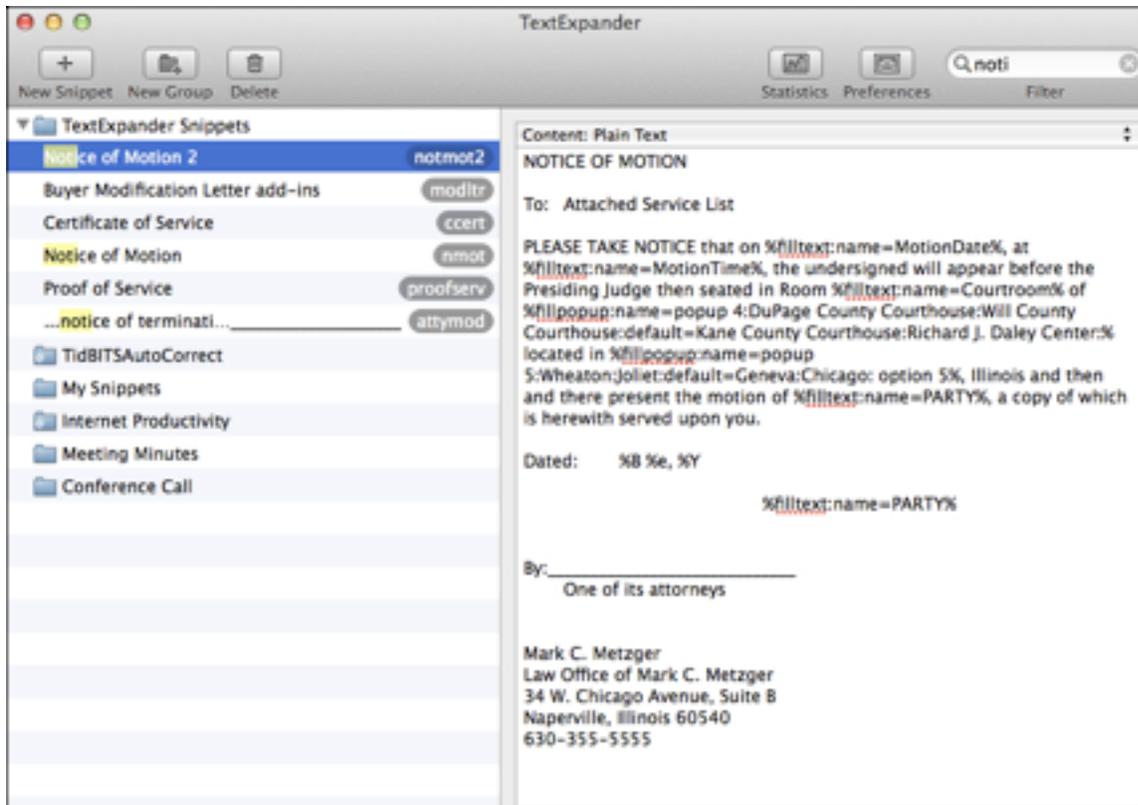
Invoking the snippet puts the following on the screen:

The gray text in the boxes is the name of each field. They could be used several times, but they'd only need to be filled in once. The courthouse and city fields are formatted as drop down menus and all Mark's paralegal has to do is choose any of the five counties in the area and insert the client's name to complete the text of a notice of motion. Notice that the snippet also took care of automatically including the date the snippet was run as today's date.

Pro Tip: Combining TextExpander with MacSimplePrompter gives you astonishing power to do things like allow the user to make drop down menu choices about what document elements to include, thereby generating a MacSimplePrompter-ready document to be expanded.

Document Automation Option 3: Word and Excel

Microsoft Office includes a mail merge feature, something which first appeared in version 3.0 in the late 1980s. Mail merge was probably the first document automation technology, and the feature was simply the process of treating the name, address and salutation ("Dear Mr. Smith") as fields, and allowed the user to generate a form letter to an entire mailing list simply by conducting the merge. In other words, it would take a mailing list

consisting of name, address and salutation fields and allow the user to turn a list of 100 names and address into 100 personalized letters.

Starting somewhat later, around the time that buying Office was more economical than buying Word alone, Microsoft added the ability to use an Excel spreadsheet as the data source.

The discussion of document automation above presumes that we're not looking for a way to make 100 different versions of something in pass (although you might, if you were looking to address customized interrogatories to the parties in a mass tort case, for example), but rather to use the computer to produce a single, highly customized document.

As it turns out, we can hijack Word's mail merge feature to get that done.

Start with Excel. Going back to your field list, put each field name in first row of the spreadsheet, one in each column. Also, because you'll need it later, in the second row, put in data for a sample document (that is pretend you're being asked for the information in the first row, and enter it into the second row). Now, save the spreadsheet with an appropriate name, at the place from which you intend to use it from now on (In other words, don't save it on the desktop now and move it later. Doing that will break everything later). If you don't save the spreadsheet before going on, Word won't know how to find it.

Next, load the document you're automating into Word. Invoke Word's Mail Merge Manager, which can be found in the Tools menu. This will bring up a small box with lots of menus in it:
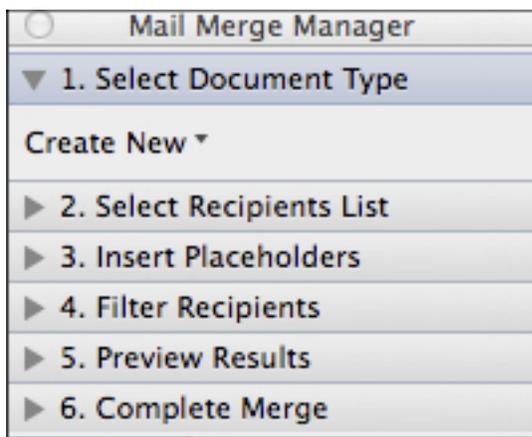
Under 1, click the "Create New" menu and choose "Form Letters." That will cause the box to enlarge with words to the effect that you're using the current document. It will also cause Part 2 to open, asking you to "Get List", which is Word's way of asking you where the data you're going to put into the document is going to come from. Click on the Get List menu and choose "Open Data Source." You will then be presented with a dialog box. Navigate to where you saved the spreadsheet you just made and select it.

Now, leaving that little mail merge box alone, look at your document there in the background. Find the first piece of text you highlighted on the paper version of the document. Select that text and delete it. Return your attention to the Mail Merge box floating above your document and click on "3. Insert Placeholders." This will cause part 3 of the box to open and display a really ugly index card/tab interface. The "Contacts" tab will now display a scrolling list of your field names. Pick the field name that stands in for the data you just deleted and drag it onto your document, right where the text you deleted use to be (or wherever else you want to put it). You'll see your field name now in the document, surrounded by chevrons. Do that with all of your fields, remembering that you can use the same field several times if you need to do so.

Save the document (this time giving it a new name, preferably with the word "TEMPLATE" in the name).

Return your attention to the Mail Merge box and open "5. Preview Results." Now, click on the button just below the down-pointing triangle and the number 5 – the button with the red chevrons above the letters, "ABC." Doing so will cause Word to display the text you entered in place of the field names. This is a good way for you to be sure you've formatted things correctly. If there are things you want to tweak, click the chevron/ABC button again so you see the field names and do your tweaking, remembering to save your work.

Once you're satisfied, go to item 6 in the Mail Merge box and click the second button, which will generate a new document for you, which is your completed, automated document result. Because your document automation is now complete, when you want to use the document in the future, you will simply complete these steps:

- Open the now-existing spreadsheet and enter the data you want in the document in row 2. Save it.

- Open the Word template you created. In most cases, the Mail Merge Manager will automatically open, because Word usually recognizes the document as a merge document. Go to 6, "Complete Merge" and click that second button.

You only need to do all of the field manipulation and setup once.

Pro Tip: You can also use a FileMaker Pro database as a data source, which means you can use all the rich interface development tools to build a nice interface to prompt your user and perform error-checking to eliminate certain kinds of errors in the data collection process. Or, maybe you've developed or purchased a customized database, based on FileMaker that contains the data you need. Word will be happy to Hoover that data out for you, and put it into your documents.

Document Automation Option 4: I hate you guys and I'll just take the last one I did and do some cut and paste stuff in Word.

At this point, some of you are undoubtedly thinking, "Wouldn't it be easier to do this with just cut and paste with the last version of the document I made?" To that, we say, "Maybe, but probably not." Using the merge, even in the hijacked way described here, means that the person running the merge won't be interacting with the document. That minimizes the possibility of errors getting introduced.

The merge makes sure that everything gets merged; the cut/paste approach interrupted by a phone call puts your document in jeopardy of embarrassing mistakes like wrong names and settlement or rent amounts being in the final document. Why? Because if the person running your cut/paste operation forgets to find and replace even one of your fake "fields" there's no way to know for sure that they missed it, because there's already text or numbers in the document. That means you spend lots of time proofreading. If a field is left blank in your spreadsheet, there's either going to be a really ugly field name sitting in your document with chevrons on it or just a blank space. You'll notice that. What are the chances you'll casually notice $750 instead of $950 in the rent field, which is what your interrupted cut/paste will produce?

There's one more advantage to this approach that doesn't exist in the cut/paste world: you can make Excel do work for you. Suppose you have a document that uses pronouns. Rather than making the user enter all of the necessary pronouns in the spreadsheet, you could add a column to the spreadsheet right after the one collecting the person's name. Call it "Sex" or "m/f" and then simply have the user enter either "m" or "f" in row two. Now you can write Excel formulas that look at the answer to the m/f question and, based on the answer given, automatically put the correct pronoun into your pronoun field ("him" or "her") and automatically also determine the correct possessive forms as well. Excel will also let the user enter slashed dates (3/24/14) and allow you to reformat it to whatever date format you want or need.

So, why should you use Excel and the Word merge feature instead of just cutting and pasting? Because doing so saves you time, dramatically reduces errors, allows you to hand off the document production process and produces a better document.

You're probably still better off with one of the first two tools discussed above, but we'd rather you were automating than not.

*   *   *   *   *

Now go have fun doing it. And from now on, when you start to write a document, ask yourself if this one could or should be automated for future use. If so, put it on a list of documents to automate. Then work on them one at a time or get your assistant doing it, but do it. The benefits go well beyond the simple rationales we've listed here.

As a last resort, you can always pay someone to do the automation for you.

Are there other ways to make my Mac produce documents for me?

There are a host of other ways to invoke the ghost in your machine. You could use AppleScript to automate the collection of information, perhaps from a FileMaker Pro database and produce documents in Word all on its own.

You could feed data in a spreadsheet or database back into a government or other PDF form.

You could take any government form and fill it out, either directly or from data in FileMaker Pro or Excel. No typewriters required.

We're not going to show you any of those things here, though. For that, the ABA Techshow Board needs to invite us back.