COMINGLING CONTENT CONTROLS AND FORMFIELDS IN WORD DOCUMENTS

(A white paper by Gregory K. Maxey)

If you search the net on comingling Content Controls (CCs) and Formfields (FFs) in a Word document, you will find that most experienced Word users advise not to do it. Some more strongly than others and I am more in the middle ground. I would have to have a very compelling reason to comingle CCs and FFs and never for convenience.

The two are certainly not a perfect marriage and there is no indication that Microsoft made a focused or even adequate effort to:

* fully develop content controls in the first place (or since) or,
* replace legacy formfields with content controls or,
* ensure that they can coexist happily together.

Upon the introduction of content controls with Word 2007, there was an obvious omission of a CC control checkbox. This omission alone is probably the biggest contributor to Word 2007 form designers scratching their heads and deciding to comingle the available CCs with a FF checkbox in a document protected for filling in forms. As the old saying goes, “Old habits die hard.”

Note: In this article, I make frequent reference to “protected for filling in forms.” “Filling in forms” is the protection method designed to restrict editing when using FFs in a document. You can create protected forms using only CCs that do not require this method. See:

[Create Form with Content Controls](http://gregmaxey.mvps.org/word_tip_pages/content_control_fillin_form.html)

So, you might ask, what is the problem? What are the consequences to the document or document users of comingling CCs and FFs?

There are no criminal consequences, fees, fines, or safety hazards to the document or the document user when you comingle CCs and FFs in document. However, there are observed anomalies and other consequences and the question you should ask is, “Why would I ever want to comingle CCs and FFs in the same document in the first place?”

If you find a good answer then do so. However, if you do you should make yourself aware of these anomalies and consequences to the document and their effects on the document user.

Do you think you have a good reason to comingle? Let’s see if you can be convinced otherwise.

1. If one or more of your document users is limited to Word 2003 or earlier then comingling CCs with FFs is dead in the water. Stop here and stick with FFs!

2. “On or more of my users is limited to Word 2007. I want to take advantage of the picture CC and I need checkboxes in my form.”

* Perhaps you could redesign your form using CCs exclusively. Could you use a single CC dropdown list with “Yes/No/Maybe” type entries to replace a series of FF checkboxes intended as a single exclusive user selection?

 CC Dropdown with choices: Choose

* If you must have the look of a checkbox (e.g., for a printed document) consider a series of dropdown lists with check and uncheck symbols as the list members displaying the unchecked symbol as default.

 CC Dropdowns with symbols:  Yes  No  Maybe

Note: I have programmed the pseudo checkboxes above to toggle state if you enter them via a left mouse click. This is not required, but if used you must have a macro-enabled document. Unfortunately, subsequent mouse clicks while the CC has focus are not detected. To change the state again, you will have to ether exit the CC and reenter it via a mouse click or select the desired state from the list.

Bonus Tip: To create the pseudo dropdown list checkbox symbols illustrated above, set the content control font to Wingdings and use ALT+0168 (on the numeric keypad) in the Add>Display Name field to create the unchecked symbol and ALT+0254 to create the checked symbol. Remove the default “Choose item” list entry. After creating the control, you can save it as a buildingblock for reuse.

* Perhaps the best option, suggested in user feedback, is a variation of my [Add Interactive Checkboxes to a Document](http://gregmaxey.mvps.org/word_tip_pages/add_interactive_toggle_objects_to_document.html). This method employs rich text controls containing macrobutton fields that toggle between two symbols.

 Richtext CCs displaying one of two symbols:

Notes:

1. This method requires a macro-enabled document and produces a pseudo content control checkbox that looks and performs in Word2007 similar to an actual checkbox content control in Word 2010/2013. Code is used to confirm/set the user’s field button click event to 1 (if changed, user option is reset when the document is closed).

2. In a document protected for filling in forms (like this one), when you click the checkbox to toggle the state, the focus moves automatically to the next field in a document. I don’t consider this important, because the sole purposed of these controls is to eliminate any need to use CCs comingled with formfields.

3. You cannot tab out of a richtext CC. There is a limited work around if place the controls in individual table cells. See: [Content Control Grouping and Navigation](http://gregmaxey.mvps.org/word_tip_pages/content_control_grouping_and_navigation.html).

Here are the three methods compared with the FFs. Are you convinced?

FFs: [x]  Yes [x]  No [x]  Maybe

CC Dropdown with choices: Choose

CC Dropdowns with symbols:  Yes  No  Maybe

CC Richtext with macrobuttons: Yes No Maybe

3. “I need the ability to use a combobox CC or a picture CC and still perform calculations with FFs”

If you and your document users can accept using a macro-enabled document, you can use CCs for performing many calculations. Use CCs to define variable values in conjuction with traditional formula field code, or use code to perform calculations.

|  |  |  |
| --- | --- | --- |
| Quantity | Price | Line Total |
| 2 | $6.00 | $ 12.00 |

Bonus Tip: See my Content Control Math tips page for more tips on using CCs in math calculations.

Note: I nested the formula field in a richtext CC in order to target the VBA “Fields.Update” method to the single field in the control.

If you are still not convinced and want to plunge ahead and comingle CCs and FFs, consider the following:

1. Comingling will require the document to be protected for filling in forms thereby disabling many of Words advanced features (e.g., spell checking). With CCs you can restrict editing to targeted ranges (the content controls) and retain use of these features.

2. You can tab from one FF to the next FF or the next CC. However, if the last field in the document is a FF and the first field in the document is a CC then when you tab from the last FF the next field selected in the first FF. CCs before the first FF are skipped.

3. You can tab from one CC to the next CC. However, without complex code in a macro-enabled document, you cannot tab from a CC to the next FF.

Note: Unlike with FFs, tabbing through CCs is not a continuous loop. A tab press when the last CCs in a document has no effect and the focus is not set to the first CC in the document.

4. Under certain conditions, CCs in a protected form temporarily lock up. For example, this will occur if you check or uncheck a FF checkbox and then click in a CC. When this occurs, you can unlock the CC by simply clicking in a protected area of the document. This would surely frustrate an unknowing user!

Note: Tabbing from a FF checkbox to the next content controls does not cause this behavior.

This document is a protected macro-enabled document. It contains code that eliminates most of the tabbing anomalies described above. You can tab through each CC (except the rich text CCs) and FFs in this document from the first CC on page 1 to the last CC control on this page. Unfortunately, since a tab press in the last CC has no effect, the user must physically select the last FF. A tab from the last FF, will set the focus to the first CC.

Click here to enter text.       Click here to enter text. Click here to enter text.

In summary, I agree with most other experience users who advise not to comingle CCs and FFs. However, I will not go so far as to say, “don’t use them together.” I would consider comingling only when there is no viable alternative. For instance, if I absolutely must have a picture content control and needed to perform calculations in a “macro-free” document.