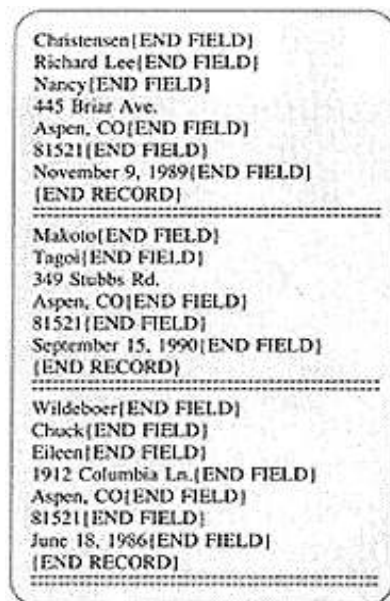


A Macro to Clean Up Your Secondary Merge Files
WordPerfect Magazine
By Elden Nelson

The *Please Wait* prompt disappeared. I anxiously scanned the document, then broke into sobs. My merge had failed. Misfired. It was mangled. I'd spent hours – no, days – entering the information for my secondary merge file, and now the merge didn't work. I fought the impulse to call Customer Support. I already knew their response. They'd tell me that in the midst of all the flexibility of WP's Merge feature, beneath all its power, lies the Cardinal Rule of Merge: Each record must have an equal number of fields. If a secondary merge file doesn't you're asking for trouble.

The task

Since secondary files may have hundreds of records, sometimes with more than 20 fields in each record, it can be nearly impossible to follow this rule to the letter. Look at Screen 1 below. At first glance, it looks like a good secondary file: the {END FIELD} codes are always at the end of a line, immediately following the field text. {END RECORD} codes are at the end of every record, followed by [HPg] codes (represented by a double-dashed line).



```
Christensen{END FIELD}  
Richard Lee{END FIELD}  
Nancy{END FIELD}  
445 Briar Ave.  
Aspen, CO{END FIELD}  
81521{END FIELD}  
November 9, 1989{END FIELD}  
{END RECORD}  
-----  
Makoto{END FIELD}  
Togo{END FIELD}  
349 Stubbs Rd.  
Aspen, CO{END FIELD}  
81521{END FIELD}  
September 15, 1990{END FIELD}  
{END RECORD}  
-----  
Wildeboer{END FIELD}  
Chuck{END FIELD}  
Eileen{END FIELD}  
1912 Columbia Ln.{END FIELD}  
Aspen, CO{END FIELD}  
81521{END FIELD}  
June 18, 1986{END FIELD}  
{END RECORD}  
-----
```

**Screen 1. An excerpt from
a secondary file**

Now take another look. You'll notice that all three records have a different number of {END FIELD} codes. This secondary merge file is a disaster waiting to happen.

With so many lines of data and so many codes in each record, it's easy to accidentally slip in an extra {END FIELD} code or to leave one out. Finding the record with the wrong number of fields can take longer than it took to type the information in the first place. That's where the included CHECKSEC.WPM macro comes in. It's a macro that lets you specify how many fields a record should have, then checks each record to find how many fields it does have.

Any time the macro finds a record with the wrong number of fields, it leaves a comment.

This comment describes how many fields that record has and how many it ought to have. You can then find these irregular records using the Search (F2) feature. More on this later.

The macro

The second line of the macro is {ASSIGN}wrong~~. This line clears the variable "wrong." Since the macro uses this variable to keep track of whether any records in the secondary merge file have an irregular number of fields, the macro clears it right at first.

Lines 4 and 5 prompt you to enter how many fields ought to be in each record. First, line 4 asks the question and assigns the answer to the fields variable. Then, line 5 checks to make sure the response is a number. It does this by dividing your answer by itself. If zero or a letter is entered, this equals something other than an integer, which the {IF} command interprets as false.

If the expression is false, the macro goes to line 40, which is the second part of the IF-ELSE-END-IF routine that begins on line 5. These commands tell you to try again, then return the macro to the start.

If you answer correctly, the macro continues to line 6. This line determines how many pages are in the document, which is information the macro uses to know when to quit.

Until now, the macro has gathered preliminary information, such as how many fields should be in the record, how many pages are in the merge file and so forth.

Lines 7-29 are the heart of the macro. These lines loop over and over as the macro counts each field in each record in your secondary merge file. Line 7 marks where the loop begins.

Lines 8-10 determine if the cursor is on the last page of the document. If so, the macro goes to the series of commands beginning on line 30.

These commands tell you whether or not to make changes to the secondary file.

Line 11 gives you a status report, telling you which record the macro is working on. This can be comforting, since you won't be able to see the macro do its work – which may take a few minutes on longer secondary files. Line 12 sets variable "check" to 0. This variable keeps track of the number of fields in the current record.

Next, if the cursor isn't at the end of the document, the macro continues. The {SYSTEM}right~=56884~ command on line 15 checks if the cursor is at an {END RECORD} code. If so, line 16 checks whether this record has the right number of fields.

If the record doesn't, line 17 makes a note of it by assigning a value to the variable "wrong." Lines 18-20 put a comment box in that record, noting how many fields it has and how many it should have. Finally, lines 21-22 go to the next record and start the process over.

When the {IF} statement on line 15 finds that the cursor isn't at an {END RECORD} code, lines 24-29 check for an {END FIELD} code at the end of the line.

If one exists, 1 is added to variable check, so it always contains the number of fields in the current record. The cursor then moves to the next line in the secondary merge, and the macro starts looking for the next {END FIELD}, the next {END RECORD}, or the end of the document, all over again. It's a tedious process, but WordPerfect doesn't seem to mind.

Using the macro

Retrieve the secondary merge file you want to check. Determine how many fields ought to be in each record.

Above all, make sure that the secondary file follows the correct format: hard page codes must separate each record and {END FIELD} codes should never have any text, spaces or anything else following them on the same line.

When you're ready to check the file, press Macro (Alt-F10), type "checksec" and press (Enter).

The macro asks how many fields should be in each record. Type the correct number and press (Enter). The macro then displays its progress in the lower left corner of the screen. When it's done, the macro displays one of two messages.

If no problems were found, it tells you that No irregular records were found, in which case you can be confident you have an identical number of fields in each record.

If problems were found, you'll be prompted to Press Search (F2) twice to find each irregular record. Records that don't have the correct number of fields have comments in them (see Screen 2 below). You can find these records by searching for comment codes. To do so, press Search (F2) twice. The macro has [Comment] preset as the search string.

Once you've found an irregular record, fixing it is simple. The macro can't tell you which field is missing (or which one is superfluous). However, finding the irregular record is nine-tenths of the game.

When you've corrected the document, use Search and Replace to delete all of the comment codes. Do so by pressing (Home), (Home), (Home), (Up) to go to the top of the document, then pressing Replace (Alt-F2), (N) No. The search string should still be [Comment].

If not, press Text In/Out (Ctrl-F5), (1) Comment. Now, press Search (F2) twice. This removes all comments from the secondary file.

Note: You may want to run this macro again to be sure you've caught everything. Better safe than sorry, especially with complex secondary files.

Christensen[END FIELD]
Richard Lee[END FIELD]
Nancy[END FIELD]
445 Briar Ave.
Aspen, CO[END FIELD]
81521[END FIELD]
November 9, 1989[END FIELD]
[END RECORD]

Makoto[END FIELD]
Tagoi[END FIELD]
349 Stubbs Rd.
Aspen, CO[END FIELD]
81521[END FIELD]
September 15, 1990[END FIELD]

This record has 5 fields. It should have 6 fields.

[END RECORD]

Wildeboer[END FIELD]
Chuck[END FIELD]
Eileen[END FIELD]
1912 Columbia Ln.[END FIELD]
Aspen, CO[END FIELD]
81521[END FIELD]
June 18, 1986[END FIELD]

This record has 7 fields. It should have 6 fields.

[END RECORD]

Screen 2