

Document Stapling with C#

Development of an Application to Collate Study Outputs

Mark Vanderstay, Amgen Ltd, Cambridge, UK

ABSTRACT

When multiple study outputs (Tables, Listings or Graphs) need to be reviewed by a study team, it is often beneficial to collate those files into one Microsoft Word document. At Amgen this process is known as 'Stapling' and is an extremely laborious process when done manually.

This paper looks at the development of an application that simplifies the stapling process. The requirements of the application will be outlined together with the adopted solution.

INTRODUCTION

There are several methods of collating, or 'stapling' outputs. Primarily one uses Microsoft Word and either inserts each file manually, or sets up a macro to automate the process. It would, of course, be possible to use other software packages, even SAS, to do the same thing.

This application was developed using C#, an object-oriented programming language built upon Microsoft's .NET Framework. C# was chosen because it allows programmers to build client applications and also because of the author's requirement for this application to be client-based.

SYSTEM REQUIREMENTS

In order to limit development a list of requirements was necessary; similar to stating the hypothesis of a research project before the analysis begins. So as part of the development process of this application, potential users were questioned on their requirements. Thankfully, the list was short;

- 1- To be able to collate both .rtf and .cgm files
 - 2- To be able to arrange the files into the appropriate order
 - 3- To staple the documents and build a table of contents.
- Each requirement will be examined in turn.

COLLATING BOTH RTF AND CGM FILES.

The first step was to enable users to create a list of outputs to be stapled. How this list should be displayed and how users could interact with it were key usability points. Users needed a textual list so they could see the names of the files to be collated, rather than a list of icons. Several methods of adding files to the list would be possible; a menu, a context menu and 'drag and drop'. It was decided that all three methods could easily be programmed so there was no need to choose any one method over another. Development began with the initial form, which was deliberately kept compact to reflect the simplicity of the application and implementation of the drag and drop collation (Figures 1 and 2).

PhUSE 2007

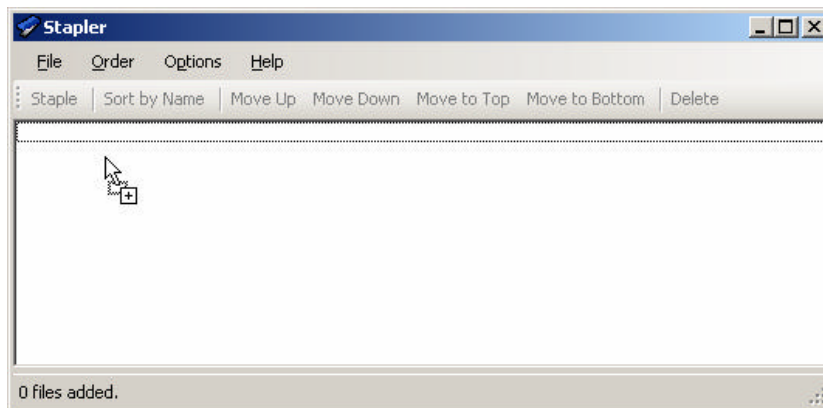


Figure 1: Drag...

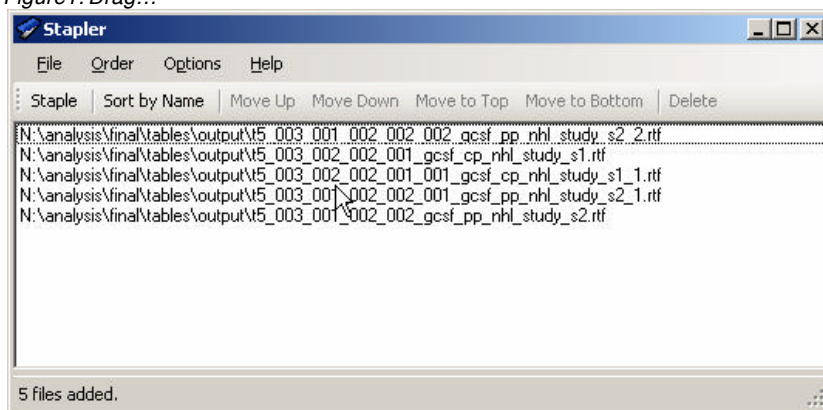


Figure 2: ...and drop.

Once this had been completed it was a simple matter of extending the file adding methods to the File menu. The addition of a File Browser dialog meant that users could browse the files and folders on their PC or network to locate the outputs they required. The same procedure was followed for the right-click (context) menu.

The user requirements had already highlighted that only RTF and CGM files were required, so checks were put in place for the types of file being added to the list.

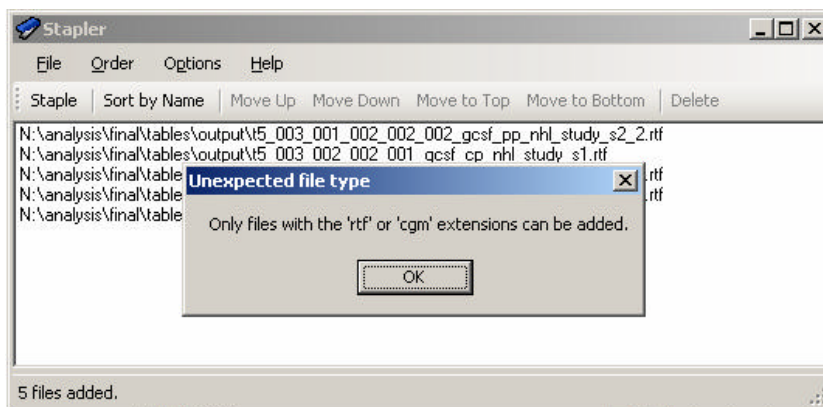


Figure 3: File type checking.

ORDERING THE FILES IN THE LIST

By default, files added to the Listbox component appeared in the order they were added. Whilst stapling files in this order was feasible it meant a greater burden on the user. An alphabetical list would have its' order determined by

PhUSE 2007

the filenames within it and since there are Amgen output numbering guidelines then this is perhaps the most obvious solution.

SORT BY NAME

This was the easiest of the ordering methods to implement. Since the Listbox included in the .NET framework has a Sort by Name property then the alphabetical sort order could be enabled and disabled by means of a button on the interface.

NON-STANDARD OUTPUTS

Adhoc or exploratory outputs that are not part of a study and do not conform to the standard numbering conventions meant a more flexible solution to the sorting implementation was necessary. Each item in the listbox has an index which determines its location within the list. As users may therefore need to place individual files at specific places within the list the only possible solution was to allow the changing of specific indices. The method chosen was as follows;

1. User highlights a file by clicking its name in the list.
2. User then clicks buttons to move file up or down.

PhUSE 2007

Once this approach was working successfully it was expanded to allow the selection of multiple items in the list. Shortcut buttons to move highlighted items directly to the top or bottom of the list were also added.

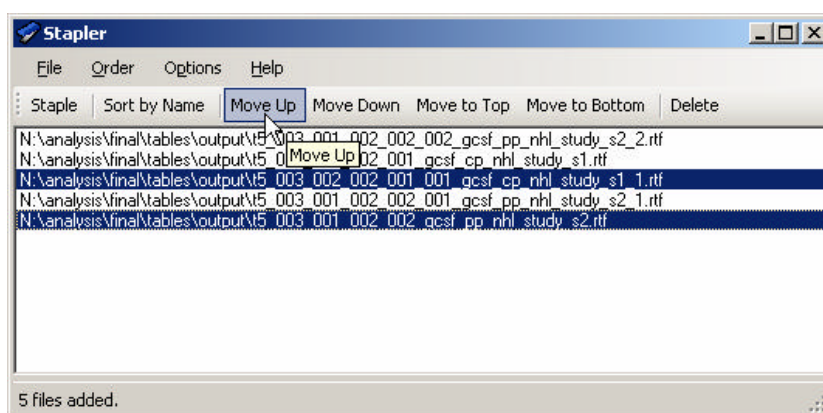


Figure 4: Moving multiple items up and down the list meant changing their indices behind-the-scenes.

Combining each of the sorting methods has given users complete control of the stapling order.

STAPLING THE DOCUMENTS

Stapling the files into one document was the ultimate aim of the application. Once a user had decided that the file order was correct it should simply be a matter of them 'pressing the button' to collate their chosen files. Whilst the user experience would be simple, a lot of activity is happening in the code.

Microsoft Word is used to create the document and is started as soon as the user clicks on the 'Staple' button. The first file in the list is opened with Word and each subsequent file is then inserted, following the manual process for creating this type of document.

It was decided that Word would remain hidden from the user to prevent them from interrupting the stapling process.

Once all files have been stapled, the user would be prompted for the location to save the new document. When the save is complete Word is exited.

It was decided that for all file operations, including stapling, a temporary copy of the file should be made which is subsequently deleted after use. Despite adding to the resources required by the application this action would reduce the risk of corrupting any critical outputs.

LAYOUTS AND RESIZING

Each file to be inserted may have either portrait or landscape formatting. So for the .rtf files in the list it is first necessary to check this parameter. This is accomplished for RTF files by opening the file and checking the rtf code within. For CGM files a different approach was needed; the dimensions were first obtained from the image within the file and then a simple algorithm to determine whether the image was wide or tall gave the portrait or landscape attribute. The image was then reduced to 90% of its original size in order fit on the page along with its title (see Table of Contents below).

The layout achieved when the Stapler is used can be seen in Figure 5.

TABLE OF CONTENTS

Building a table of contents in Word is accomplished by setting the appropriate font style to headings within a document. To replicate this in an automated application Word is used to highlight the first paragraph within each imported document then set its font style accordingly. As the first paragraph in standard Amgen output is the output title it followed that the Table of Contents, when built, would contain the titles of each imported output. However, CGM files, which do not contain readable text, needed to be treated differently. When a CGM file is imported, the file location is first added to top of the page, the font style is then set before the CGM is inserted. The result of this is that CGM files still appear in the Table of Contents.

PhUSE 2007

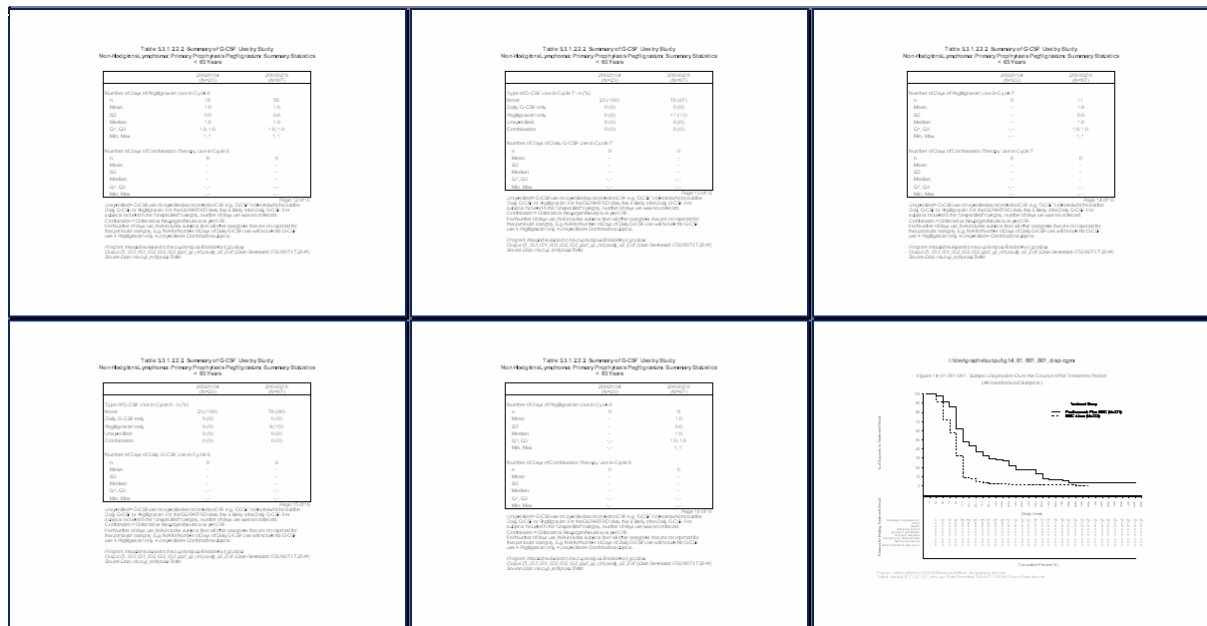


Figure 5: Final layout showing Stapled RTF tables and a CGM graph appropriately resized.

CONCLUSION

This application was initially thought of as a 'quick-win', after all, it's basically a re-working of several VBA macros present within Amgen with some new features added in. However, even with its small requirements list the development was still rather complex. This was mostly due to the (seemingly undocumented) nature of COM interoperability and its inherent complexities. The Stapler is now ready for use within Amgen UK and so far it has been well-received with users commenting on how much easier and less time-consuming the Stapling task now is. Roll-out in the US will happen soon.

ACKNOWLEDGMENTS

I would like to thank the following for their efforts in testing the Stapler application;
 Alison Kirk,
 Rob Allis.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Mark Vanderstey
 Amgen Ltd.
 240 Milton Rd.
 Cambridge Science Park
 Cambridge
 CB4 0WD

Email: markv@amgen.com

Brand and product names are trademarks of their respective companies.