

CLIENT WHITEPAPER

“Letter Generation”

Creating Missing Information Letters in GLRLETR

Graciously Submitted by:

John Tong
Savannah College of Art and Design
jtong@scad.edu
March, 2003

Creating Missing Information Letters in GLRLETR

Table of Contents

Introduction.....	3
Set up the project	5
Using Mapform.....	6
Using FormStamp	8
Using CaptureForm.....	9
1. Combining Multiple Data Fields	10
2. Multi-line Data fields.....	10
3. Date formatting.....	11
4. Selecting an E-mail address.....	12
Using FormDirector	13
GENERAL.....	16
APPENDICES	18
Appendix A:.....	18
Appendix B: Input Data File from GLRLETR.....	19

Introduction

The goal will be to create a First Missing Information letter that will replace a word MailMerge document that is currently generated via data from GLRLETR. The GLRLETR jobsub is accessed either through Direct Access on the starting Banner page, or through GJAPCTL.

The screenshot shows the 'Process Submission Control Form GJAPCTL 5.5 (PROD)' window. At the top, the 'Process' is set to 'GLRLETR' with the description 'Letter Generation Print Report', and the 'Parameter Set' is 'NOAPP'. Below this is the 'Printer Control' section with fields for 'Printer', 'Special Print', 'Lines' (set to 60), and 'Submit Time'. The 'Parameter Values' section contains two columns: 'Parameters' and 'Values'. The 'Parameters' column lists: 01 Application Code, 02 Word Processor Extract Option, 03 Print ALL Pending Letters, and 04 Letter Code. The 'Values' column lists: FINAID, 1, N, and FA_NOAPP. Below the parameter list, it specifies 'LENGTH: 30 TYPE: Character O/R: Required M/S: Single' and provides a note: 'Application code for letter(s) you wish to print.' The 'Submission' section at the bottom includes a 'Save Parameters' checkbox, an 'As:' field, and radio buttons for 'Hold' and 'Submit' (which is selected).

Parameters	Values
01 Application Code	FINAID
02 Word Processor Extract Option	1
03 Print ALL Pending Letters	N
04 Letter Code	FA_NOAPP

Figure 1: GJAPCTL showing the GLRLETR job

It is used by several Banner modules and is fed by a Population Selection (AKA a PopSel) that is run through GLBLSEL (Letter Extract Job). From there GLRLETR is run, then the resulting DOC file is run through a Mail Merge that is printed and then assembled, folded, then sealed and mailed.

Our goal at the **Savannah College of Art and Design** is to improve and make our Financial Aid office more efficient by reducing the amount of manual work, and physical resources required to provide service to our students. Thanks to recent progress made in electronic submissions and availability of forms the 'paperless' or 'paper less' Financial Aid office is now becoming a possibility.

In our example of the first missing information letter (MIL1), our old workflow involved the Mail Merge process described above, the resulting 'packet' contained all the forms the student may have required. This was often a very thick packet, and was prone to all the vagaries of manual assembly of each packet. With the FormFusion we saw an opportunity to print a single page onto a standard 'z-fold' letter that included a URL to the necessary forms and sites. This form needs no envelope, and can be folded using a machine at a very quick rate. An additional benefit is that FormFusion allows us to include a letterhead onto the top of each letter, allowing us to use inexpensive blank 'stock' and eliminating the need to stock special letterhead.

Set up the project

The first step is to collect the following:

1. The input data. In this case the output with the '**02 Word Processor Extract Option**' set to '1' is a file that ends in .doc but is in actually a plain text CSV file. More on this later in the MapForm Section. See APPENDIX B
2. The mail merge form that is the 'template'. This would normally be fed the input data from #1. See APPENDIX A
3. A copy of the complete letter after the mail merge and on official letterhead. You will use this to 'model' the FormFusion letter.

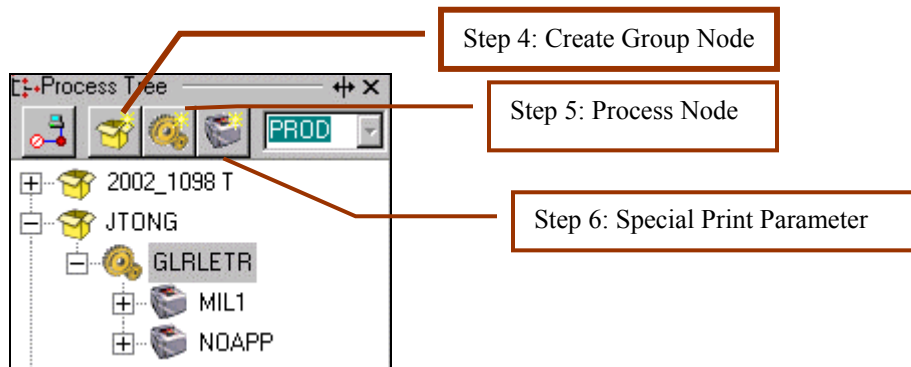
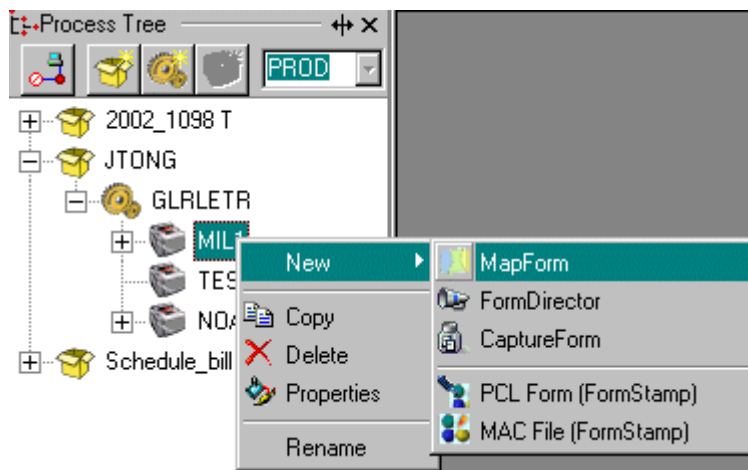


Figure 2: Setting up FormFusion nodes.

4. Create a 'Group Node' in FormFusion for your projects, in my case JTONG.
5. Create a 'Process Node' for the job that will be using FormFusion. In this case GLRLETR.
6. Create a 'Special Print Parameter' for this letter. In this case I named mine MIL1. Here you can set up a lot of details for the letter. Such as, Number of copies, the paper tray, and other features that we will explore in the "Set up FormDirector" section. You should use a different SPP for each letter. This will allow you to produce a different output, and set up a different input map for each letter. See the "General Tips" Section for additional details and functionality.

Using MapForm

After collecting the needed information and setting up a workspace for our letter, the first step is to set up MapForm to handle the input file format. A typical report output from Banner supplies data that is mapped to a 'position'. For example a student's name is always the second row down from the 3rd character in all the way to the 60th character over. GLRLETR outputs its data in a format called CSV, or a Comma Separated Value file. Data is set up one record per line, each field is separated by commas, and if there will be commas in the data field, it is wrapped in a double-quote, for example "\$1,200" might be something you might encounter. See Appendix B for the data file I came up with.



Right click on the SPP you created and select New, then MapForm. Then double click it to bring up MapForm.

Figure 3: Creating a new Mapform

Right Click the MapForm, and go to Properties. Click the "This Form" tab and change the input type to 'Comma Delimited'.



Figure 4: Importing the CSV file.

To import the data file that the GLRLETR will output, click on the red Import LIS button above. **Note** you will have to change the 'Files of Type:' filter, since our output ends in a .DOC .

When it asks you to change the characters per line, say NO.

Using FormStamp

It is now time to create the form template using FormStamp. Once again, right click on the SPP (MIL1), but this time select the PCL (FormStamp) option. You can name it whatever you'd like. It would be best to name it something descriptive, perhaps MIL1Front. FormFusion allows you to design double-sided forms and add forms conditionally to the end of existing forms (see the other tutorials).

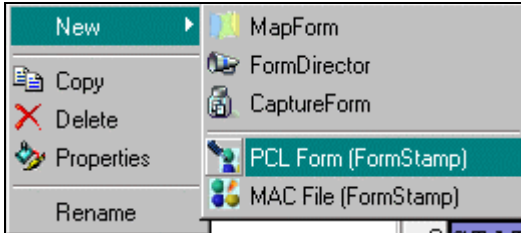


Figure 7: Creating a new FormStamp

Right click on the PCL Form after you name it and go to the Form Layout Tab. The forms we're using are 'Legal' sized (8.5inx14in).

From here, you design your form to look like the finished letter you have on letterhead. You can insert graphics, use text fields, and insert the data fields that we selected in MapForm. **HINT** : Right-click on the MIL1 SPP node and go under properties to set the default font to what you need. In my case the letter I was modeling used Times New Roman. See the Template for the finished product. (NOTE: Depending on your interest, you can use ANY font. So, you can get as fancy as you like).

See font/graphics tutorial here: http://www.evisions.com/pdf/tutorial/fonts_images.pdf

Using CaptureFormSQL

We find that not all the data we need was available on the MapForm, or that it requires a bit of modification to get it into the format that we desire. A lot of these formatting issues can be accomplished using CaptureFormSQL. From here, we will tackle the following situations:

1. Combining multiple data fields into a single data field. For example the City, State, ZIP, and First, Middle, Last Name. Even building multi-line fields such as a single address field that includes, Name, Lines of address, and City/State/ZIP.
2. Displaying Multi-line data fields. For example, displaying all the missing documents of a Missing Information Letter.
3. Pulling a current date and formatting it.
4. Selecting an Email address.

To create a new CaptureForm query, you should right click on the SPP that you have created and then go to New, then CaptureForm. Under the CaptureForm node you will see an Apple Icon called “Variable Storage”. You need to create the variables that will hold the completed queries here.

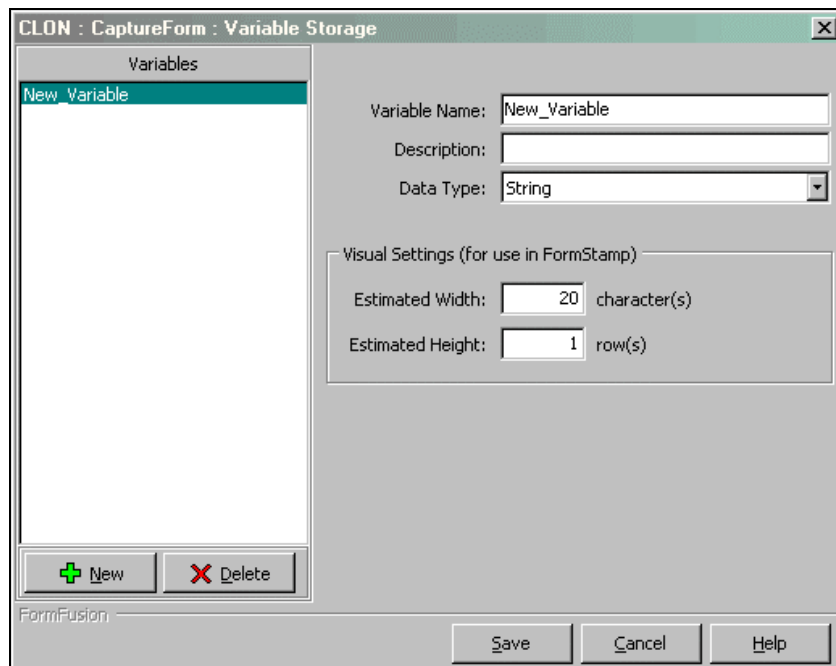


Figure 8 : New CaptureForm Variable

Create a variable called “CSZ” for City, State, and Zip. The sizing SHOULD be the sum of these three fields in the database, plus 2 bytes for separators. One may have to modify the estimated width to be smaller in order to be able to position it on the Form though because the Form will NOT allow you to place field items that run off side of the page.

1. Combining Multiple Data Fields

Now we create the CaptureForm Query. Right click on CaptureForm icon and click New, then “New CaptureForm Query”. Name it “Make CSV”. I used the following query.

```
select trim(:city) || ', ' ||
       trim(:state) || ' ' ||
       trim(:zip)
into :csz from dual
```

There are some things to note about this query. The first is the use of “TRIM” this trims the white space around the input of :city, :state, and :zip that we specified in our MapForm. Another is that I used the double bars “||” to concatenate in the separators, in this case, a comma and a space between City and State, and just a space between State and Zip Code. The last is that the “dual” table is a virtual table that doesn’t really exist. It serves as a dummy table to allow tricks such as this in SQL.

2. Multi-line Data fields

The next element to tackle is the Award Requirements list. This is a possibly very long list of items that the student needs to return to be processed. The output from Banner Letter Generation for this field is a long text string with “vertical tabs” separating each item. These displayed correctly without modification in the Word merge documents, however do NOT display correctly in a data field in FormFusion. They are translated as an unprintable character represented by a small box.

To get this to represent properly we must create an :AwardReqs variable with the estimated size to be perhaps 50 characters across and 16 lines down. These values will allow you to see what the area that may be printed on to list all the missing award requirements. After the variable is created, we need to create a CaptureForm query called “Make Award Reqs” that will ‘translate’ all those vertical tabs to carriage return/line feeds. We use the REPLACE command in this case.

```
SELECT REPLACE( :AWARD_REQ , chr(11), chr(10)||chr(13))
INTO :AWARDREQS
FROM DUAL
```

3. Date formatting

To display the date at the top of the letter in a user-friendly format, such as “March 19, 2003” as opposed to “2003-MAR-19” we must do a little manipulation. We must create a variable to hold the date, of course it should be of the data type “Date”. We can then select the current date into the variable via a simple CaptureForm query we create called “Make Date”.

```
select sysdate
into :datevariable from dual
```

Place the date data field onto your letter, then right click on it and go to “Format Object”. From there click on the Ellipses (...) to the right of the data conversion field. From here select the Date/Time option. Select <NONE> for the Time format, since we don’t care about the time in this case, then Select the format that you want for the date. **NOTE:** Do it in this order, because there is a bug that prevents the display and the formatting from updating if you just change the time section. I chose “MMMM dd, yyyy” and erased the date/time separator (which starts as a comma).

4. Selecting an E-mail address

The final more or less tricky part that we will cover is how to select a receiver's e-mail address. This is only complicated because there may or may not be e-mail address associated with the person, AND we would like the 'official' school address to be used first if available. The reason for this is that we want a CaptureForm that is generic enough to be used for ALL the FinAid letters, from NoApps (no applications, in which case they haven't even applied to the school, let alone been assigned a school address) to Award letters for continuing students. Writing a slightly different version for each would have been time consuming.

Once again, create yourself a Storage Variable called :email, and create a CaptureForm Query as follows:

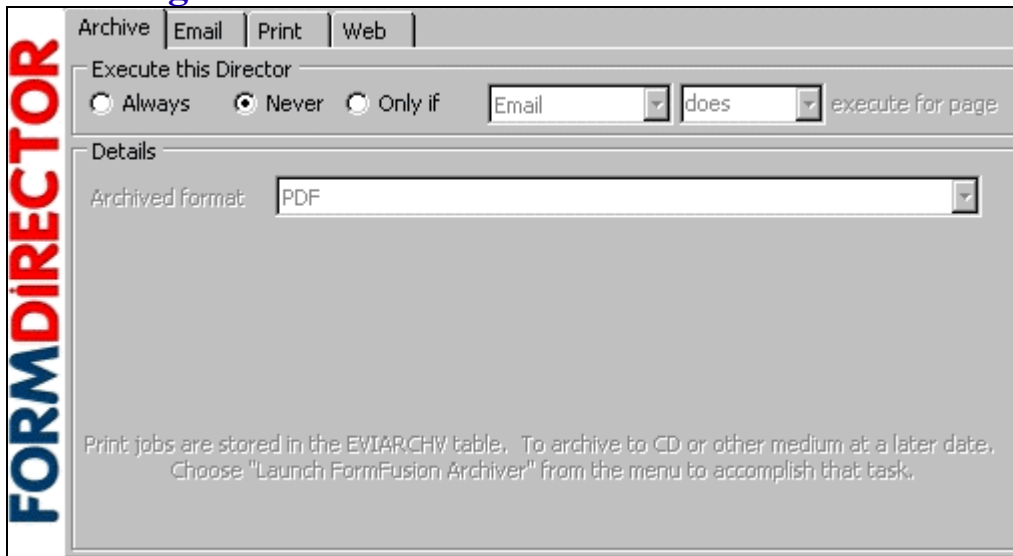
```
SELECT DISTINCT
      nvl(a.GOREMAL_EMAIL_ADDRESS,
          b.GOREMAL_EMAIL_ADDRESS)
INTO :email
FROM GOREMAL a, GOREMAL b, spriden c
WHERE c.spriden_pidm = f_getpidm(TRIM(:ID))
      AND b.goremal_pidm (+) = c.spriden_pidm
      AND a.goremal_pidm (+) = c.spriden_pidm
      AND a.GOREMAL_STATUS_IND (+) = 'A'
      AND b.GOREMAL_STATUS_IND (+) = 'A'
      AND a.GOREMAL_EMAL_CODE (+) = 'STEM'
      AND b.GOREMAL_EMAL_CODE (+) = 'EMAL'
```

This query executes what is called an 'outer join' to SPRIDEN as a base table. Analysis of this query is outside the realm of this document, however a friendly DBA and quite a few web sites may be able to lend assistance. It should be pointed out however that the above will use the "STEM" (aka Student Email) address first if available, or default to the Secondary "EMAL" address if it is available or remain empty if none are available. **NOTE:** There is the use of the TRIM command above around the :ID data field. This is VERY important as superfluous trailing spaces or left padded spaces will cause this query to fail.

Using FormDirector

FormDirector will handle the multiple printing, e-mailing, and archiving functions that we will require. In our situation, we ALWAYS want to mail a letter to the student to ensure that they receive this information. We will want to mail a copy of the letter in .PDF format to the student. Finally we wish to keep a copy for ourselves to allow reprinting and for verification purposes. **NOTE:** We added a small CC: section to the bottom of the form to show the e-mail address that it was mailed to if available.

Archiving



The screenshot shows the 'Archiving' tab of the FormDirector software. The interface includes a vertical 'FORMDIRECTOR' logo on the left. At the top, there are four tabs: 'Archive', 'Email', 'Print', and 'Web'. Below the tabs, the 'Execute this Director' section contains three radio buttons: 'Always', 'Never' (which is selected), and 'Only if'. To the right of these buttons are two dropdown menus, one containing 'Email' and the other containing 'does', followed by the text 'execute for page'. Below this is the 'Details' section, which features a dropdown menu for 'Archived format' set to 'PDF'. At the bottom of the window, there is a note: 'Print jobs are stored in the EVIARCHV table. To archive to CD or other medium at a later date, Choose "Launch FormFusion Archiver" from the menu to accomplish that task.'

Figure 9: Archiving Tab of FormDirector

Archiving saves a copy of your generated documents in several tables on the database. EVIARCHV and EVIBLOB tables to be specific. They can be accessed later via the ArchiveDirector. Alternately the 'Web' tab might be used in the future to create a directory full of files to associate with students via ApplicationXtender.

Email

The screenshot shows the 'Email' tab in the FormDirector interface. On the left, there is a vertical logo that reads 'FORM DIRECTOR'. The main window has a title bar with 'Archive', 'Email', 'Print', and 'Web' buttons. Below the title bar, there is a section titled 'Execute this Director' with three radio buttons: 'Always' (selected), 'Never', and 'Only if'. To the right of these buttons are two dropdown menus, one containing 'Archive' and the other 'does', followed by the text 'execute for page'. Below this is a 'Details' section with several fields: 'To:' with the value ':email', 'From:' with 'finaid@scad.edu', 'CC:', 'BCC:', 'Subject:' with 'Financial aid at the Savannah College of Art and Design' and a 'Body...' button, 'Return Receipt:', 'Attachment format:' with a dropdown set to 'PDF' and a 'Password ...' button, 'Filename of Attachment:' with ':id .pdf', and a checkbox for 'Custom Script'.

Figure 10: FormDirector Email Tab

This is a very important tab. Note that the To: field is a data field variable “:email”. Additionally, we added some general text to the letter that asks the student to review the PDF for the appropriate actions. This was done with the ‘Body...’ button.

NOTE: There is a SPACE in the Filename of Attachment between :id and .pdf . This is because the use of any data variable will cause the next character **to be ignored**. It is *Very Important* to remember this. If the filename was “:id.pdf” the file that the student receives would look like “000667202pdf” instead of “000667202.pdf”. This goes for the Subject line, To: field, OR anywhere above.

Printing

The screenshot shows the 'Print' tab of the FormDirector configuration interface. On the left side, there is a vertical logo that reads 'FORMDIRECTOR'. The interface has a tabbed structure with 'Archive', 'Email', 'Print', and 'Web' tabs, where 'Print' is currently selected. Below the tabs, there is a section titled 'Execute this Director' with three radio buttons: 'Always', 'Never' (which is selected), and 'Only if'. To the right of these buttons are two dropdown menus; the first is set to 'Email' and the second to 'does not', followed by the text 'execute for page'. Below this is a 'Details' section with a label 'Print Script:' and a text input field containing the command: `lp -c -d$PRNT :TEMPPATH_ :PRINTFILENAME_`.

Figure 11: FormDirector Print Tab

This is another simple tab. Remember to modify the print script to point it to the correct printer.

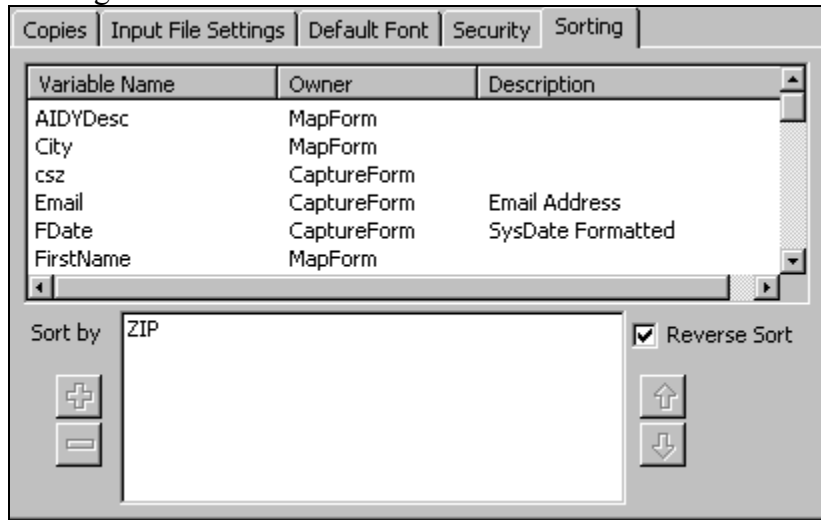
NOTE: The “:PRINTFILENAME_” variable is a built in variable. There is also a printer variable “\$PRNT” that allows users to pass the printer to be selected by the end user as well. The :TEMPPATH_ variable is the directory where the temporary PCL file is created by FormFusion. What is sent to the OS after it is translated by the above is:

```
lp -c -dfinanceprinter /tmp/formfusion_temp_file.pcl
```

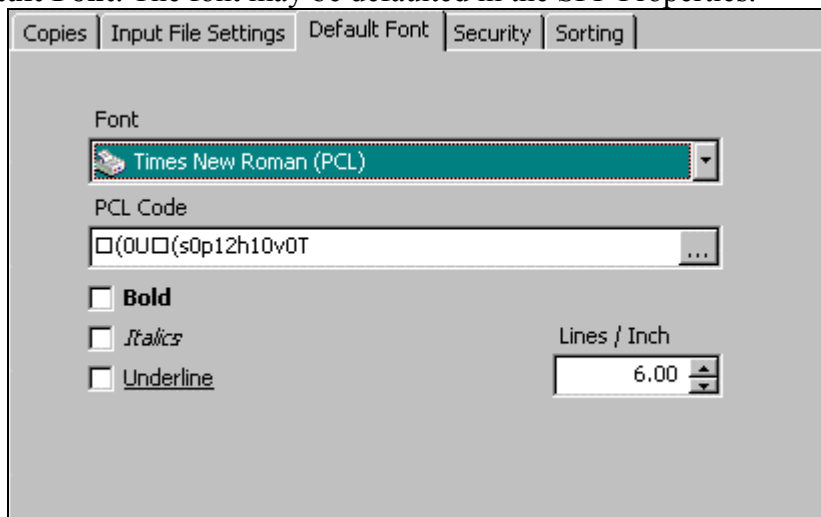
GENERAL TIPS

Some additional situations you may encounter.

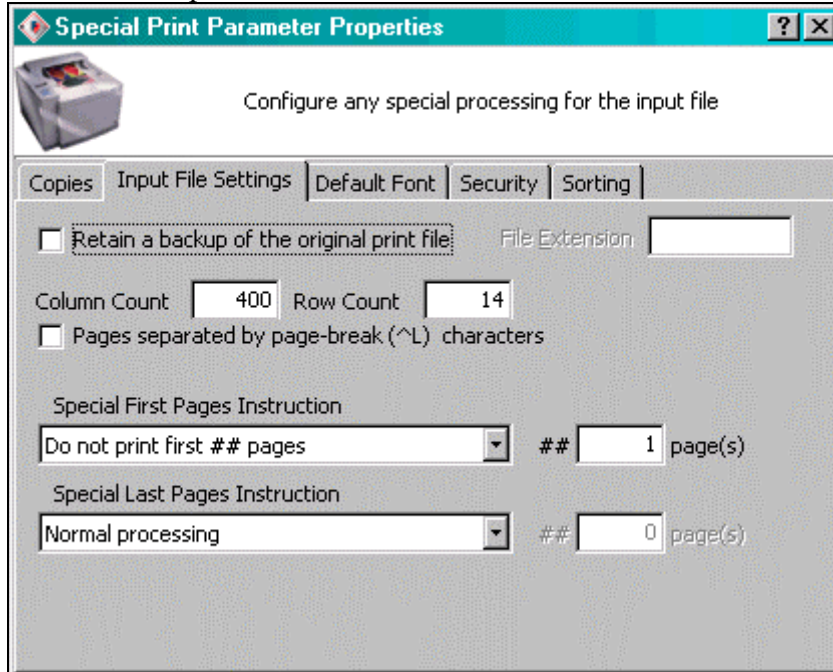
1. **Sorting Order.** Sorting is done in the properties of the SPP. For example it is necessary to reverse the ZIP order when printing because it must pass through the folding machine, which will Reverse the order once more. This will add to the postal savings.



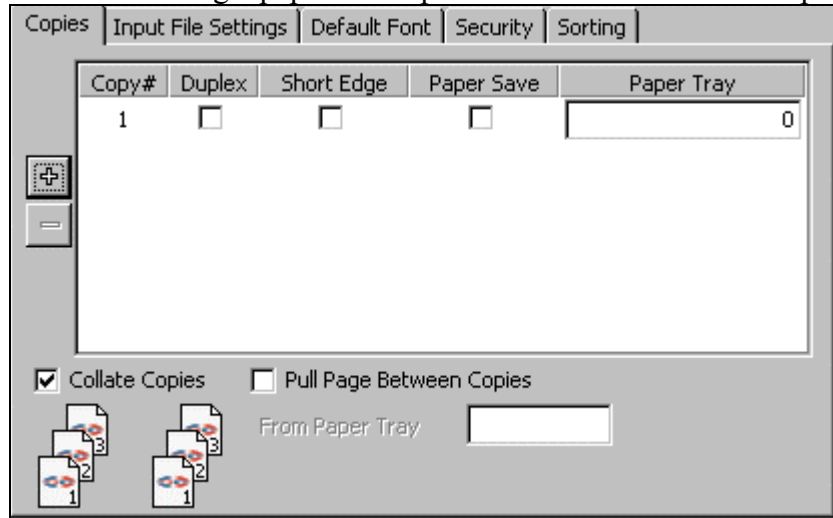
2. **Default Font.** The font may be defaulted in the SPP Properties.



3. **Header Pages.** NOT printing the header page (the first letter with all the headers and not actual data) may also be done in SPP properties. For example see below, where we do NOT print the first letter.



4. **Paper Bins.** Selecting a paper bin to print from is done in the SPP properties.



APPENDICES

Appendix A:

Mail merge letter:

November 13, 2003

ID#: «ID»

«FIRSTNAME» «MIDDLEINTL» «LASTNAME»

«STR1»

«STR2»

«CITY», «STATE» «ZIP»

«INTRO»

Thank you for your interest in financial aid at Savannah College of Art and Design for the ...

The items listed below are required to complete your application for Federal/State financial aid. Some items may be informational only or require you to take action elsewhere. All documents/copies should be completed, signed and returned to our office as soon as possible.

«AWARD_REQUIREMENTS»

If you have any questions about this letter please contact the following:

...

If for some reason you do not plan to attend our school, please notify our office in writing to remove your name from our mailing list. We also recommend that you notify the admissions office of your decision if you have not already done so.

Sincerely,

The Financial Aid Office
Savannah College of Art and Design

Appendix B: Input Data File from GLRLETR

AIDY_YR_DESC,AWARD_REQUIREMENTS,CITY,FIRSTNAME,ID,INTRO,LASTNAME,MIDDLE
INTL,STATE,STR1,STR2,ZIP
"2003-2004 Award Year","Signed copy of parent's 2002 actual U.S. Income
Tax Return
Copy of parent's 2002 W-2 form(s) reported on tax return
SCAD Federal/State Financial Aid Application form
SCAD Request for Federal Direct Parent Loan (PLUS) form
Signed copy of student's 2002 actual U.S. Income Tax Return
Copy of student's 2002 W-2 form(s) reported on tax return
2003-2004 Verification Worksheet for Dependent
Students","Hillsbury","Arthur","000603676","Dear
Arthur","Alcott","Simon","VA","PO Box 1355","","242243"
"2003-2004 Award Year","SCAD Federal/State Financial Aid Application
form
SCAD Request for Federal Direct Parent Loan (PLUS) form","Lady
Lake","Christine","000645789","Dear
Christine","Abarbanel","Jacksonville","FL","840 Walker
Ridge","","23125"