

PrintDuplex

COLLABORATORS

	<i>TITLE :</i> PrintDuplex		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 3, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	PrintDuplex	1
1.1	PrintDuplex	1
1.2	Mumbo-Jumbo	1
1.3	Introduction	1
1.4	Using PrintDuplex	2
1.5	PostScript Printing	3
1.6	ASCII Printing	4
1.7	Summary	4
1.8	Miscellaneous	5

Chapter 1

PrintDuplex

1.1 PrintDuplex

Program: PrintDuplex V 1.0.0
Copyright © 1993 MRsoftware, All Rights Reserved
Last Change: 01/16/93
Author: Mark R. Rinfret, MRsoftware

Mumbo-Jumbo

Introduction

Using PrintDuplex

PostScript Printing

ASCII Printing

Summary

Miscellaneous

1.2 Mumbo-Jumbo

PrintDuplex is freely distributable but is Copyright © MRsoftware. It is not in the public domain and may not be distributed with commercial software without prior written permission nor may any fee be charged for its use, other than reasonable duplication and distribution fees. PrintDuplex is provided as-is and MRsoftware makes no warranties, either explicit or implicit with regard to its use.

1.3 Introduction

Introduction

PrintDuplex is an Amiga utility which will print PostScript or ASCII text files to a laser printer or printer with single-sheet capability, two pages per sheet. It will not work with printers which rely on continuous forms. PrintDuplex has a fairly naive notion of a document: a stream of pages numbered 1-N where N is an even number. You can print documents having an odd number of pages. PrintDuplex will supply the "missing" (blank) page.

A late addition to PrintDuplex is the ability to handle "n-up" PostScript documents created with such programs as PSPrint, A2PS, etc. Though this worked with the few samples I tried, there may be problems with more complex documents. Please let me know if you have a problem and, if possible, send me a copy of the document which revealed the problem.

I wrote PrintDuplex because I had to output a large Professional Page 3.0 document to my laser printer in two-sided format and doing so from within Professional Page (via an ARExx script) proved to be quite unfriendly to the rest of my system. By directing the PostScript output to a file, the Pro Page printing operation must only be performed once (per revision of the document) and PrintDuplex can generate as many hardcopies as are needed while placing much less of a burden on my system. I've successfully used PrintDuplex with PostScript files generated by Professional Page and New Horizons' ProWrite and quite a few plain (non-PostScript ASCII) documents.

1.4 Using PrintDuplex

Using PrintDuplex

The command synopsis for PrintDuplex is:

```
PrintDuplex [options] From <input_file> To <output_file>
```

and the AmigaDOS command template is:

```
First/K/N, Last/K/N, Front/S, Back/S, Layout/K/N, ASCII/S, Lines/K/N,  
Copies/K/N, From/A, To/A
```

where <input_file> is the name of the text file containing PostScript or plain ASCII text to be printed and <output_file> is the name of the file or device (e.g. PRT:, PAR:) to receive the output. When PrintDuplex is started, it first scans the entire <input_file>, determining page boundaries. It then prompts you for confirmation to proceed with the printing of first the front side, then the back side. Be sure that the printer is in the ready state BEFORE confirming that you are ready to print either page group. In particular, after the front side is printed, you'll have to remove the printed pages from the output tray, juggle them a bit to get an even stack, then place them back in the input tray, usually top-in, unprinted side up.

The [options] may be none or any of the following, in any order:

ASCII

The file contains only ASCII text. Do not attempt to interpret it as a PostScript file (default format is PostScript). Pagination is accomplished by a combination of form-feed detection and line-counting (see Lines).

Copies <count>

Specifies the number of copies to be printed.

First <page_number>

Specifies the first <page_number> to be printed. The <page_number> is currently restricted to an odd value (1, 3, 5, etc.).

Last <page_number>

Specifies the last <page_number> to be printed. The <page_number> is currently restricted to an even value (2, 4, 6, etc.).

Lines <number>

This option sets the number of lines per page to <number> and is only relevant when the ASCII option is enabled. If the file is known to be paginated with form-feeds, make sure that <number> is at least as large as the lines-per-page setting used when the file was formatted (otherwise, you'll end up with a series of alternating long and short pages).

Front

Print only the front-side pages in the selected range.

Back

Print only the back-side pages in the selected range.

Layout <pages_per_page>

This option is useful for printing the output of certain programs which have "n-up" capability (printing multiple logical pages per physical page), such as PSPrint, A2PS, etc. The <pages_per_page> value MUST match the value that was used by the originating program or you're guaranteed to get wierd results. This feature was added very late in the game. I hope it works consistently.

1.5 PostScript Printing

PostScript Printing

A little empirical analysis of a PostScript text file output by Pro Page (I'm not all that knowledgeable with regard to PostScript) revealed that it consisted of the following structure:

```
prologue
%%Page: <lpn> <ppn> (<lpn> <ppn> = logical, physical page number)
...page "program"
.
.
.
%%Page: <lpn_final> <ppn_final>
```

```
...page "program"  
epilogue
```

where <lpn> is the logical page number and <ppn> is the physical page number. By partitioning the file into its components, the text can be sent to the printer (or a file) in the correct order. This order is:

```
prologue, 1, 3, ..., N-1 epilogue  
(reload the paper tray)  
prologue, N, N-2, ..., 2, epilogue
```

I decided to send the prologue and epilogue before/after each side after experiencing some strange problems with my printer. This may not have been the best fix, but it solved the immediate problem.

The determination of the start and end of each PostScript page isn't quite as straightforward as it should be since (I think) the available tools don't all output the "same" PostScript :-). PrintDuplex doesn't have any real PostScript parsing prowess. The file is scanned primarily for %%Page comments. All text preceding the first %%Page comment is considered to be the prologue. The file is then scanned for 'showpage' commands (currently, they must appear in column 1, though that is not a PostScript requirement). The showpage command tentatively marks the end of a page unless a %%Page comment is found beyond it. Thus, the epilogue is comprised of all the text that follows the last showpage command.

1.6 ASCII Printing

ASCII Text Printing

ASCII text files are split into pages through the use of an arbitrary line count (the default is 60 but can be overridden with the Lines option) or through the detection of form feed characters (decimal 12) embedded within the text. These work together to determine the page boundaries.

1.7 Summary

Summary

Duplex printing obviously saves paper and reduces the volume of your printed documents. On the down side, if you have a finicky printer, you may find that it jams the paper when it is reloaded. This is largely dependent upon the quality of your paper, how much your printer distorts the paper as it prints to it, humidity, temperature, phase of the moon, etc. Readability may be a problem if you're using a lightweight paper. The print may show through to the opposite side, which can be quite distracting. Experiment with your setup and a small document to determine if duplex printing is feasible for you. Use the First/Last options to print a subset of the document until you get it right. Your printer manufacturer may have provided suggestions

for two-sided printing in your owner's manual. For example, Panasonic KXP-4455 owners are advised to use the bottom paper tray.

1.8 Miscellaneous

Miscellaneous

PrintDuplex is my first complete C++ programming exercise and was programmed with Comeau C++ 3.0b with Templates in conjunction with SAS/C 6.1. The experience was enjoyable, frustrating, illuminating, and convinced me that C++ offers a better way to craft programs. I was actually quite surprised at how easily it integrates with the AmigaDOS/Intuition environment. I hadn't intended to add any GUI features for the first release. However, rolling a class "around" EasyRequest was so, ah, easy, that I couldn't resist.

Feedback?

Please send bug reports and enhancement requests to:

BIX: markr
Usenet: mrr@mrsoft.network23.com

PrintDuplex is freeware. However, if you feel strongly motivated to send a small donation, I'll happily accept it. Send all correspondence to:

MRsoftware
348 Indian Avenue
Portsmouth, RI 02871
