XY USERS Group HORIZONS

The Newsletter for and about the Xyvision User's Community May 1993 Volume 4 Issue 2

INCREASING sysmgr 1--> PRODUCTIVITY

By Gary Rozmierski and Wren McMains

I have always felt that we were moderately efficient. I knew there were a few ways we could improve our productivity, but I didn't think that it could amount to thousands of dollars in savings in a single year.

Gary's Story . . .

This past year we were chosen by a well-known publisher to produce custom-selected, short-run books on a demand basis with our Xyvision system. Our equipment, a Model 85 Atlas, runs 2.1Q.5 software. To be able to print custom books involved creating a database with over 300 stories on various topics, with each story being between one and four pages long. Each custom book will consist of an individualized cover, two pages of contents, and up to 94 pages of text. The customer will fill in a form selecting the stories they want, the total page count of which will be near or equal to 94-pages. We will, effectively, be producing a 96-page book. The actual printing and binding of the book will be performed on a Xerox DocuTech high speed printer.

The Xerox DocuTech Production Publisher scans hard copy, converts it to digital masters, then prints up to 135 pages per minute with a resolution of 600 dpi. It can produce saddle-stitched, folded and trimmed booklets, as well as stapled and/or punched booklets. It can also handle PostScript files by way of a standalone Media Server, which is the method we will be using.

After assembling all the stories in our database, and trying several methods to produce sample books, it was apparent that, although the theory was sound, the method was not as productive as we needed. The test runs showed that to be productive with the volume anticipated, we would need:

- to have more than one document assembly (DA) ticket available at all times;
- to be able to verify each DA ticket;
- to be able to queue up for output and/or printing, multiple DA tickets;
- to automatically create a two-page table of contents;
- to be able to "print to disk" the PostScript files on our PC;
- to have software on the PC that would automatically sense the incoming file, open a file by the incoming file's name, write the file to disk, close the file, and wait to receive the next file; and
- to have the PostScript files compressed so they can be "moved" onto a 3-1/2" high-density disk.

To be able to print custom books involved creating a database with over 300 stories on various topics.

Getting Help

With the criteria outlined, I turned to Wren McMains, a productivity consultant, for a solution. Wren and I spoke by telephone a few times, discussing and explaining what I wanted to do and how I thought it could be

(Continued on page 6)

1993 Seminar Series A Winner!

This year's seminar series offered in conjunction with User Conference 93 is hot. With the registration deadline only a few weeks away, the sessions are filling up fast.

Geared toward the intermediate or seasoned user of Xyvision publishing systems, three of these seminars offer an opportunity to advance your knowledge of Xyvision and PostScript macros, and UNIX script writing. The *Advanced Macro Writing Seminar* will be led by Robert Carr of The Psychological Corp. Ed Hougardy of Xyvision will lead the *PostScript Macro Writing Seminar*, and Wren McMains will return for the third time to teach *Intermediate UNIX and Script Writing*.

For management-level publishing system users, we're pleased to offer a free session on *How to Achieve a Return on Investment* led by Doug Bruce of Waverly Press.

For Contex users we're offering an introduction to the world of UNIX in *UNIX for Contex Users* led by Jim Bampos of Xyvision. The session will include demonstrations of the Rel. 4.3, and the new graphical user interface. For Contex owners and managers, we'll host a free session on the *Technical Advisory Panel* led by Dan Clarke of Xyvision Color Systems.

These sessions take place on Wednesday, June 9. The four user sessions start at 8:30 a.m. and will end around 11:45 a.m. Registration costs are $75/person for members and $100/person for non-members. The two free sessions start at 10:00 a.m. and end at 11:45 a.m.

Call the XyUsers Group for more information. Don't wait until it's too late. Register Today! — *C.M.D.*

☒

Productivity

(Continued from page 1)

accomplished. His experience with the system was evident in the way he discussed possible solutions. After we had agreed on the results we wanted (and discussed Wren's fee), Wren dialed into our system from his office in Concord, Massachusetts via modem to check on our environment and to transfer some of his development tools that would allow him to be more productive when he arrived on site. Shortly after exploring our system by modem, Wren came on site for three days and was able to put together almost everything we needed to do on the Xyvision side, but the program on the PC proved to be just a bit more problematic. In just a few days, Wren was able to work out the solution from his office.

The Procedure

We have printed instruction sheets for our operators that detail the steps to make each custom book.

First, we make the cover by copying our master cover division and inputting the cover information from the order sheet. Next we make the DA ticket. The program to do this is accessible through: 1) Mgmt Tools, 2) System Support, 3) Support Tools, 4) More Tools. The program Wren made shows in the list of programs as "order-enty." We then type in the order number and the program creates the DA ticket by that name.

When the DA ticket comes up, we fill in the file comment field with the invoice number, job number and date. In the table comment field we put the customer name and city. When talking with the customer, they may refer to a particular job by its cover title, or the purchaser's name and/or city. It's our responsibility to know what particular job they are referring to.

We create a rule for the cover and the contents in the DA ticket, (i.e., cov-1323 and cont-1323, where the four-digit number corresponds to the assigned job number). It's important to consistently use the same naming convention, especially when you have more than one person doing this work. And, of course, the program is set up this way and won't work properly unless this naming convention is followed. We then add new rules, one for each division name to match each entry on the order sheet. Then we print the DA ticket to get a hard copy to attach to the order sheet. Next we store the DA ticket which also processes it, verifies each division's linking, creates the contents pages, and prints the contents pages to our 600dpi laser printer. We then send the cover for output to the laser printer.

Checking And Processing

We have our proofroom check the cover and contents pages against the order sheet. It's not necessary to read the contents because it is computer generated from a corrected master list; we just check that each entry is the correct one.

When the proofroom OKs the cover and contents, we process the order(s). Processing is accomplished through 1) Mgmt Tools; 2) System Support; 3) Support Tools; 4) More Tools; and 5) choosing the program "orderproc." This process sends the job to the PC through the IBM port which we use for inputting the jobs we have keyed by our cottage labor force. On the PC side we enter "capture" at the DOS prompt which starts the program that will 1) sense the incoming file, 2) create a file by that name, and 3) compress the file. We leave the system up over night while it processes the orders to the PC that we have queued up. By processing at night we do not need operator attendance and the processing does not interfere with other production work. The next day we'll use the 4DOS command "move" which copies the compressed files to a 3-1/2″ disk and erases them from the hard drive in one command.

> ***By processing at night we do not need operator attendance and the processing does not interfere with other production work.***

We also have the option to print the job to our laser printer. We can print the job itself, or print to disk and to the laser printer simultaneously.

Automated Maintenance

While Wren was here I asked him if he could automate our maintenance procedure. In the past we would run maintenance on Monday mornings, the procedure taking an average of 1-1/2 hours every week. Wren set it up so that we run our maintenance procedure at the end-of-shift on Monday evening, unattended, with just a few keystrokes. At the sysmgr 1--> prompt we enter "maint-1" and go home.

The maintenance program includes removing unprocessed files in /comm/Ribm (lost files), removing old communications logs, shortening log and error files for: composeq, lineprinter1q, typesetter1q, typesetter3q, out1q, out3q, commq, postfmtq and postoutq, spool maintenance, remove save buffers, remove raw files (in defined directories), and style and session maintenance. It still takes 1-1/2 hours to run, but it's done outside production hours. The savings in increased production hours realized from this procedure alone was more than enough to cover the cost of hiring Wren for the other tasks he performed.

Another thing I asked Wren to do for us (as long as you have him on the clock you should use his abilities to the fullest), was to set up a program that runs "from XSF" on selected divisions and transmits the files to the PC, also through the IBM port. We access this program through 1) Job Tools, 2) More Tools, and choose the program "div-2-pc." We run the capture program on the PC and when it senses the incoming file, it places it in a different subdirectory than the PostScript files and does not compress them. We have a number of jobs that are composed on our system but the updating of the information is done by our cottage labor force, and then re-input to the system, composed and proofs sent to the customer. Using the same keystrokes to ship to another company to be used

Trick Used For Writing Maintenance Scripts

Maintenance scripts and other scripts that either take a long time to execute, or delete files and directories as they execute, can be painful and time consuming to debug. One trick is to create a test variable:

```
set tst = 'echo'
```

and then precede all commands which do anything for real with $tst, for example:

```
$tst rm -r /usr/spool/SPOOL1
$tst mkdir /usr/spool/SPOOL1
```

You can then run your script and see what commands it is going to issue and find any errors quickly. Once all the bugs are out, edit the script so that the tst variable is set to nothing and the script is ready to run for real:

```
set tst = ″
```

in a CD-ROM is another reason we send the ASCII file back to the PC.

Wren's automated method saves us time and time is money. That's what a productivity consultant like Wren is all about—saving time and effort through automating procedures—and Wren is good at his job.

Wren's Job . . .

The first problem I encountered was how an operator could specify the contents of each book with minimal effort, while still providing enough information to control all the automated processing that had to take place. Because Xyvision already performs a lot of automated processing based on the document assembly (DA) ticket, we decided to use it as our base.

Efficient Job Specification

Because Xyvision only allows one DA file per job, the first step was writing a script to manage DA tickets. We kept them in a single directory with names based on the customer order number. To edit one, the script renames it to the name Xyvision expects (:da_job.p). After editing is complete, a translation is run to extract an ASCII list of divisions that can be used in other scripts; the script than starts a process that automatically produces the contents division using this list, and finally the DA file is renamed back to the name it is saved under.

Sample Code:

```
mkdir /tmp/$ty-$num
if ( -e /usr/boyd/orders/$ty-$num.p )
then
   dp /usr/boyd/orders/$ty-$num.p
      /tmp/$ty-$num/:da_job.p
   dp /usr/boyd/orders/$ty- $num.t
      /tmp/$ty- $num/:da_job.t
   endif
# Style data editor (sdedit) creates
#    DA file if none exists
sdedit -cd/tmp/$ty-$num da job
# Extraction translation and enque
#    command to produce contents
#    pages goes here
mv /tmp/$ty-$num/:da_job.p /usr
   /boyd/orders/$ty-$num.p
mv /tmp/$ty-$num/:da_job.t sdit -cd
   /tmp/$ty-$num da job /usr/boyd
   /orders/$ty-$num.t
rm -r /tmp/$ty-$num
```

where $ty contains a product name (three characters) and $num an order number (four digits). Editing in a unique /tmp directory allows multiple operators to enter orders at the same time. By checking for the existence of the /tmp/$ty-$num directory first, you can make sure someone else is not editing the same DA file.

Minimize Processing By Proofing First

Because the contents pages are produced automatically based on the contents of the DA file, proofing the contents guarantees that the requested stories will be added to the book in the right order and that folios will be numbered correctly. It is, therefore, important to produce the contents pages during the day while people are still around to check them. The rest of the processing can happen at night when the system is unattended. Doing most of the processing at night also keeps the autoprocessing load from slowing down the system when operators are working. We even produce the contents pages in the background, allowing an operator to specify many orders without having to wait for the order entry script to do additional processing.

Using Your System At Night

The bulk of the work is done at night. Each night the last one out, before turning out the lights, queues all proofed books. Our database for each product (type of book) consists of composed story divisions sitting under a single job. To produce each book, the appropriate DA file must be copied into place, the folios in each division included in the book recomposed, the job verified, PostScript pages for each division written to files, optionally hard-copy proofs printed, and all the PostScript pages combined into a single file for each book which is then transmitted to the PC.

One major problem we encountered revolved around the way Xyvision names the files it produces when using the PostScript output driver. The file names are based on the job/division names (three characters from the job name, six characters from the division name, followed by a page number.) Because you always tend to name divisions based on existing names which have more than six characters and have meaning to operators, this PostScript naming convention presented a problem even in one book, to say nothing of the problems in producing many books that reuse the same division but with different folios. And because the names had already been published to our customers and were used in the incoming orders to specify a book, it was too late to change them. To make things more difficult, the processing is asynchronous — Xyvision produces the files in an output queue over which you have no control. We finally

(Continued on page 12)

Productivity

(Continued from page 7)

solved this problem by having the processing script monitor the output queue, waiting until the pages for each division were completed before starting on the next division. In this way it is possible to add these pages to the composite output file before they were overwritten by pages from another division with a similar name-(which in our application always seemed to follow quickly.)

Controlling A PC From A Xyvision System

The most interesting outcome of this project was using the Xyvision system to control processing on PCs. File transfer to the PC is slow over a serial line. If we had to have a person sitting at the PC to capture each individual file it would have eaten up all the potential profit from this venture (the file transfer takes about 30 minutes for each file). Because many books a day may be involved, we wanted this to happen during the night so that we would not be tying up PCs that could be used for other things during the day. Another problem we had was that the files were too big to fit on a diskette. The people with the DocuTech system told us which compression program they wanted us to use, but it only ran on a PC, not on the Xyvision system. Therefore, we needed a way to capture each file (book) separately on the PC, compress it, and name it appropriately. Optimally, we would have also transferred the compressed file to diskette, but after the first file, the diskette was almost full and we couldn't find a DOS command to switch diskettes without a human around to assist.

We're really excited about the potential for sharing processing resources between Xyvision systems and other machines like PCs. This project couldn't fund the development of all the software necessary to handle the communication in both directions and issue commands on the foreign machine under control of a script running on the local machine. We are looking for other users who might benefit from this kind of power to share some of the development costs in return for a license to use this software. If you're interested in exploring this project, contact me at (508) 371-1319.

Gary Rozmierski is System Manager/ Assistant Forman at Boyd Printing Co., Inc. Wren McMains is an independent productivity consultant based in Concord, Massachusetts.

☒

XyUsers Group Newsletter
P.O. Box 3022
Kirkland, WA 98083-3022