The Latest Word

Marketing edge over Sony

Canon offers still-video camera

Sony had a large booth at ANPA to tout its Mavigraph Plus video imaging system, but Canon beat it to market with a still-video camera. Like the unreleased Sony Mavica, the Canon RC-701 uses a CCD array and a 2" floppy disk to record the video image. The introductory price is $2,600. A playback unit is $2,700 and the color printer comes in just under $7,000.

Although these prices are not low enough to attract consumer attention, we imagine that a reasonable number of news organizations and industrial concerns will pick up systems to experiment with. As Canon ramps up its volume manufacturing, system prices will fall.

For the publishing industry, image quality will be the major concern. We haven't seen actual samples from the RC-701, but its resolution is specified as 780×488 pixels. This is similar to the Sony Mavica's resolution.

Although the images Sony showed at ANPA looked good as 3"×3" prints, they obviously would not hold up well under enlargement. At the present state of the art, newspapers will use video images only when nothing better can be found.

Desktop publishers, on the other hand, will find that the 300-dpi resolution of their laser printers is the factor that limits quality, and the still-video camera's resolution should fit well with that technology. For that market, getting the cost of the camera down will be the primary issue.

OEM customers sought

Superset interfaces Eikonix, PC

Superset, Inc., has a hardware interface between its XP/48 graphics processor and the Eikonix scanner. The combination can scan a full-color image at 4,096×4,096-pel resolution, store and manipulate the image, and output the result on a variety of devices.

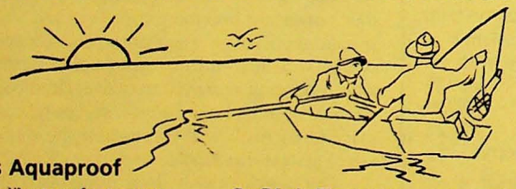
The XP/48, a 48-bit-wide bit-slice processor optimized for vector graphics, is controlled by an ordinary PC through a DMA card plugged into the PC's backplane.

The XP/48 supports various graphics standards, such as GKS, IGES, AutoCAD and Matrix's SCODL. It is able to manipulate the color images at real-time speeds and to merge vector-based image files. It can then output four-color separations to high-resolution imaging devices. Superset has announced it will support the PostScript and Interpress page-description languages, and that it is willing to write customized drivers for typesetters.

Superset, which also has products for the seismic-plotting industry, wants to sell this product to publishing-industry OEMs. The first unit was delivered to Logicon, Inc., for a mapping application. The XP/48 and interface will sell for between $20,000 and $27,500, depending on software options.

For more information, contact Joe Mintz at Superset: 11025 Roselle Street, San Diego, California 92121, or telephone (619) 452-8665.

Comments from Our Readers



GrafMark's Aquaproof

I would just like to refer to your article on GrafMark in your recent DRUPA report issue.

You refer to the inks used in our Aquaproof system as dyes. I would just like to clarify that the inks are indeed the same density, color match and pigmentation you'd expect to find on your press. The only difference being that our inks are developed by water, so it is truly ink on paper.

Secondly, we claim that the only variable is given by the different substrate and that will be a variable on the press also, remembering that by exposure we have control over dot gain.

Thirdly, it was reported in *The Seybold Report on Publishing Systems*, Vol. 15, No. 4, October 21, 1985, that Ian Burns, Ron Johnson and Tina Armold have formed a new company, GrafMark. The company was established, therefore, prior to the reunion of old boyhood chums, Burns and Thompson.

Finally, any system that is automated, gives a controllable proof to match the press run in under 15 minutes, and is half the running cost of systems on the market can be called Mickey Mouse, Donald Duck or whatever; this one is called Aquaproof, and if it is half as successful as the forenamed characters, both ourselves and our exciting industry will be that much happier.

Keep up the good work.

Ian Burns, President
GrafMark Ltd.
International Marketing & Consultants
UK Celadon House
Cumberhills Road, Duffield
Derby, DE6 4AS, England

The Seybold Report on Publishing Systems

Published by Seybold Publications, Inc.

Business and Production Offices:
PO Box 644, Media, PA 19063
Telephone: (215) 565-2480
Telex: 4991493
Facsimile: (215) 565-4659
John W. Seybold, *Publisher*
Stephen E. Edwards, *Managing Editor*
Peter E. Dyson, *Assoc. Editor/Bus. Mgr.*
Martha E. Morrison, *Office Manager*
Jan Bright, *Circulation Manager*
Preecha Edwards, *System Manager*
Patrice S. White, *Editorial Assistant*

West Coast Office:
6922 Wildlife Road, Malibu, CA 90265
Telephone: (213) 457-5850
Telex: 501527
Facsimile: (213) 457-4704
Jonathan W. Seybold, *Editor-in-Chief*
Laurel Brunner
Deborah Griffin
Lena Radstrom
Larry Rott

Associate Editors:
William J. Solimeno
PO Box 83, Harvard, MA 01451
Telephone: (617) 456-8990
Andrew Tribute,
European Correspondent
Hazelgrove Farm
Haslemere Rd., Brook Godalming
Surrey GU8 5LB, England
Telephone: Wormley (042879) 2249

Interleaf: A Fast–Moving Tech–Doc Supplier

LATE in 1983, we invited David Boucher, president of Interleaf, to speak at the up–coming Seybold Seminars '84. He declined on the grounds that Interleaf did not sell a publishing system. Its OPS (Office Publishing System) was essentially a super word processing system intended for office documentation. We argued that we thought Interleaf and its system would be of interest to seminar attendees. Only half convinced, Interleaf decided to test the water by bringing a demonstration system to see what people in the publishing market would think of this office automation product.

The system was *not* a graphic arts product. It could accommodate only a few specially digitized type face/type size combinations. It could only output to a 300–dpi laser printer. It could handle only single–column work. It could not perform hyphenation. Nevertheless, the Seminar attendees loved it. They liked the way Interleaf handled style specifications and text formatting. They liked the way it allowed users to create and edit graphic frames in the middle of a text document. Most of all, they liked the extraordinary speed, the ease of use, and the fact that the software was written to run on standard off–the–shelf engineering workstations rather than specially–built hardware.

Interleaf soon discovered that it is, indeed, in the publishing systems marketplace. It shipped its first system in May 1984, two months after the Seybold Seminars debut. In October it took a hotel suite to demonstrate its system at Graph Expo East because it was too late to get space on the exposition floor. In March 1985, it returned to the Seybold Seminars with more sophisticated software and a cooperative demonstration with Monotype, ImagiTex and Data Recording Systems. This year it put on a tour de force at the Seminars with software running on Sun, Apollo, DEC and IBM workstations, and a preview of release 3.0 software, which will give Interleaf much more sophisticated capabilities in both text and graphics.

Two years ago, Interleaf was not sure it belonged at a publishing conference. This year there were so many Interleaf users among Seybold Seminar attendees that they were able to form an Interleaf user's group at the seminar.

Thus far, Interleaf has been the biggest winner in the "tech–doc" revolution. It has sold fast, easy–to–use and cost–effective systems to a lot of first–time users, most of whom had not typeset documents in the past. But it has also been steadily increasing system functionality. Release 2.5 software used to prepare this article provides excellent working tools for many people engaged in the production of utility documents. Release 3.0 (due for field release this fall) will move the system squarely into the typesetting system market. We have every reason to expect that continuing development will make the system an increasingly attractive alternative for people with more sophisticated composition and graphic requirements.

In this issue, we will concentrate first on the current 2.5 software, then preview the features which will be added with 3.0. We did most of our testing with software running on a Sun Microsystems 2/120 workstation. Not all of the features we discuss are currently available on all other platforms. Interleaf hopes to get everyone except IBM on the common release level by summer.

Interleaf software will run on a Sun workstation, pictured here, as well as VAX, Apollo and the new IBM RT workstations.

The market. Interleaf has focused initially on users who are satisfied with 300–dpi plain–paper output of a few type fonts in a few sizes. The current software will also drive a typesetter for higher–resolution output, but it would not be fair to characterize it as a typesetting system. The typical customer has been a company or a group engaged in product documentation, proposals, engineering change orders and the like.

With the increased sophistication of the next software releases, Interleaf will move into broader markets— including those which require higher quality output and more sophisticated composition and page formatting. The emphasis will remain on documentation applications, but the system should be attractive to some portions of the commercial, magazine, and even newspaper markets.

Products. Interleaf offers two products; a composition and pagination application package which comes in several flavors, and a series of laser printers. There are three versions of the applications software: WPS (Workstation Publishing Software), TPS (Technical Publishing Software), and UPS (University Publishing Software).

- *TPS* is the top–of–the–line version of the software which contains all of the functionality offered by Interleaf. This is the software package described in this article.
- *WPS* is a simpler subset of TPS. WPS does not include the ability to handle multi–column work, the ability to drive typesetters or generate tables of contents automatically. It provides all of the current TPS functions for creating graphics, but does not support scanning of images. Equally important, it does not support communications and translations for input from PCs and word processors. WPS is intended to be used as a workstation word processing program in its own right.
- *UPS* is a new package tailored to the university market. This will follow the progress of the TPS product line, one release behind. Therefore when TPS 3.0 is released, the UPS product line will essentially consist of TPS 2.5 software.

Because WPS is a subset of TPS, the two packages can be mixed on the same network. Users often start with WPS, then upgrade to TPS. Other customers have installed workstations running WPS as input and editing front ends for production workstations running TPS. Assuming consistent release levels, WPS files can always be read on TPS systems. The reverse of this may or may not always be true.

Hardware

Unlike most of its competition, Interleaf is truly hardware independent. It has been careful to write program code to be as transportable as possible. It has also been careful to insure that any new hardware features and functions can be supported across the range of workstations the company expects to use.

Interleaf currently supports engineering workstations from four vendors: Sun Microsystems, Apollo, DEC VAXstation II, and the new IBM RT. The company has traditionally brought new software releases up on Sun workstations first, then carried them over to the other platforms. At the moment, Sun and Apollo workstations support the latest software release 2.5. Interleaf expects to have 2.5 running on DEC workstations later this summer, and will also have the IBM RT up to 2.5 by the summer. There are, as you would expect, slight differences in the way things operate between machines, but the software is completely compatible, and documents created on one brand of workstation can be edited and output on another.

In fact, it is even possible to mix different brands of workstation on the same Ethernet network. As demonstrated at the Seybold Seminars, every workstation on the net can be given access privileges to call up, edit and compose documents stored on another brand of workstation. The only real exception is the Interleaf application software itself. To prevent piracy, Interleaf has decided that its software will run only on workstations which allow application software to check the workstation ID. The software is therefore created for a specific piece of hardware and will not run on any other device.

Typical configuration. It is difficult to cover each of the possible configurations from all of the vendors which can run the TPS software. We will therefore use a Sun–based configuration as an example:

Within the Sun product family, the base TPS product would be a stand–alone Sun 3/160 workstation. This has a single–board computer with a 68020 processor, two async interface ports, and a high–speed SCSI interface. The base configuration will include 4 MB of RAM and an 86–MB disk. Larger disks and additional memory are used when the system will be expected to support scanned graphics.

The Sun workstation has a 19″ screen with a resolution of 1152 x 900 pixels. This works out to 75 dpi, exactly one quarter the resolution of the 300–dpi output printer. The Apollo and DEC workstations have similar screen configurations. The IBM RT has a 15″ monitor.

Interleaf uses a single bit plane for display of graphics as well as text. Scanned continuous tone images are displayed as screened halftones.

Networking. An Ethernet link is used to connect several workstations and peripheral file servers together. The average configuration for Interleaf at this time is about four workstations per system. File servers may be other workstations or "headless workstations" (workstations without display screens). Workstations used as file servers are typically configured with larger disks (in the range of 190, 300 or 500 MB).

Workstations also serve as controllers for output printers and typesetters and input scanners. Controllers for output devices can also support terminals. But scanner controllers cannot. The non–stop nature of scanner input requires a dedicated processor which is always available to receive image data as it is sent from the scanner.

Most workstations operate off their own disk while composing and paginating a document. But it is also possible to have workstations which do not have any local disk storage. A diskless workstation will "page" data into and out of its RAM over the Ethernet link. All processing will be done locally. What amazes us most is the speed at which this can occur, even when paginating pages which include graphics.

Pages are composed with a target typesetter or printer in mind. However, since composition produces an intermediary file which is output–unit independent (except for set widths of characters), the paginated document may be sent to a different printer for proof output. The actual conversion of the file into the particular typesetter or printer format occurs as a separate batch routine within the printer server.

Scanners. Oddly enough, even though virtually all Interleaf systems installed to date are used to drive 300–dpi output printers (rather than typesetters) Interleaf does not yet support input from inexpensive tabletop scanners. At the moment, the only scanners for which Interleaf provides an interface are the ImagiTex units. While Interleaf believes its market is more than adequately served by the resolution offered by these devices (475 or 775 samples per inch) for both contone (which it emphasizes is overkill) and line art, it has also agreed to support the ECRM Autokons. This development is expected to be complete within the next few months. When other scanners are added, they will be in the range of 300 to 500 dpi and in a price range closer to $10,000–$15,000 rather than $40,000.

Printers and typesetters. Interleaf is optimized for output to a variety of plain–paper laser printers, most in the 300–dpi range. For best results, Interleaf prefers to drive printers through its own RIP. This gives it control over output type faces and spacing. The base level output printer is the ubiquitous Canon CX. For higher–volume requirements Interleaf has signed a reciprocal OEM agreement with Dataproducts under which Interleaf sells the Dataproducts 26–page/minute printer and Dataproducts sells the Interleaf RIP.

In addition, Interleaf also drives printers with controllers supplied by others. These include Imagen printers (sold by Interleaf under an OEM arrangement), and the Apple LaserWriter.

The Interleaf RIP. In order to get the output print speeds it wants, Interleaf has developed its own raster image processor controller. The RIP–2 is based on a 68000 processor and special high–speed hardware. Since it builds a bit map of the entire page to be printed, it requires a lot of memory. Interleaf uses 3 MB of RAM for storage of bit maps, fonts and code.

Like most other companies who have built RIPs to work with their systems, Interleaf has built the processor to interpret its own page descriptor language. Fonts are stored as separate bit maps for each size of each character in each font. This means that the RIP cannot size or rotate type. Interleaf defends this decision by contending that the only way to achieve quality output on a 300–dpi printer is to use specially tuned bit maps for each size of each font to be printed, and modified for different print engine technologies. We think that Adobe has pretty well disposed of that notion with the LaserWriter and other PostScript printers. However, it is certainly true that the Interleaf RIP is much faster than an equivalent PostScript RIP. If the application only requires a modest number of type faces in a modest number of sizes, then the Interleaf approach makes sense. If you want to output a wide variety of type faces and sizes and/or you want to be able to expand, condense and rotate type, requiring a separate bit map representation for every size of every font could be very limiting.

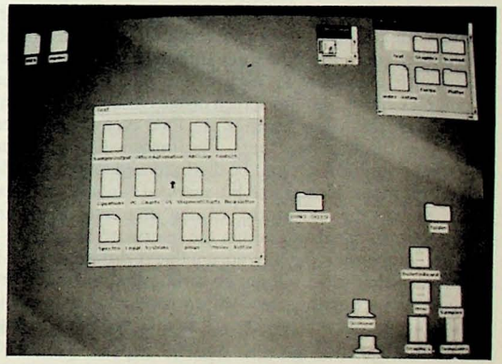
The RIP also does not screen continuous tone photographs to produce halftones. (Interleaf makes the distinction between halftones which are arbitrary sized dots of black or white with what it calls quartertones which are halftones produced with a fixed size dot of black or white.) Therefore all data sent to the RIP for printing has already been converted into halftone format. The RIP doesn't have to convert from contone to halftone on the fly. Since the RIP was specifically designed to control the output of everything composed in an Interleaf document, it supports all of the graphic primitives and fill operations which can be performed on the system.

Software

All of the application code for the TPS and WPS products is written in "C." However, since the software must run on a variety of workstations (VAX, Sun, Apollo, and IBM) it has to run under a variety of different operating systems. For instance, when running in a Sun environment the software runs under the control of Sun Unix 2.2 (essentially Berkeley 4.2). In a DEC environment, the same program will run under VMS.

Operating environment. To run the Interleaf system, the user will boot up the basic computer operating system, log–on with his password, then give the command to load the Interleaf software. From then on (except when performing utilities such as data backup) he will be in the Interleaf "world."

He is presented with a graphic "desktop" which represents the Unix or VMS tree file directory structure as nested icons. (More on this later.) All of the files contained in one icon on his desktop must be physically resident on the same disk. However, if he has access privileges which extend across the network, he may see icons representing collections of files which are actually resident both on his own workstation and on one or more system file servers.



Interleaf's Graphic Desktop

A file which has been opened for editing and/or composition is automatically "check–pointed" (saved to disk) at periodic intervals. Like most non–newspaper systems, Interleaf does not provide any automatic provision for dual writing of data. If you want to back up data you have stored on disk, you will have to run a utility program which copies some or all of the data base to backup media. Most Interleaf configurations include streamer tape drives so that the user can make tape backup copies of the files he is responsible for.

Fonts and screen display

We mentioned earlier that the typical screen can display about 1152 x 900 pixels per inch. On a 19″ screen this resolves to 75 dpi. Although some of the workstations Inter-

leaf uses are available in versions which display gray levels on the screen, the company has elected to stay with the least common denominator and does not use this facility.

Screen fonts are carefully tuned for the resolution to be used. Until now, the company has relied on a specific screen font for each output typeface and size. For 300-dpi printers driven by the Interleaf or Imagen RIPs, Interleaf provides two faces, "Modern" (sans serif) and "Classic" (serif), in 6, 8, 10, 12, 14, 16, 18, and 24 point sizes. These have been digitized in both 75-dpi (screen resolution) and 300-dpi (output printer resolution) versions in roman, italic and bold (but *not* bold italic). All software releases include two typewriter fonts, one math font, one extended math font, a symbol font, and a Greek font.

Interleaf expects to increase this font library in the near future with additional fonts made from outline masters supplied by Bitstream.

For typesetter output on the Monotype Lasercomp Interleaf supplies three screen fonts which correspond to the Monotype Times Roman, Century and Helvetica faces.

Flexifonts. This approach is obviously too limited if Interleaf wants to support a large number of type fonts on a variety of output typesetters. It has therefore come up with a scheme for "generic" screen faces which can represent a wide variety of output fonts. Release 2.5.75 of the software includes support for five generic serif and sans serif characters in each point size. The software will thus have its choice of five different widths for each serif or sans serif character. It will pick the version for screen display and laser proofing which is closest to the width of the final typeset character.

As with other new fonts, Flexifonts will originate as Bitstream outline fonts and be digitized and stored as individual bit maps. Since these bit-mapped fonts will not *exactly* match the set-widths of the character they are emulating, Interleaf will adjust the interword spacing in a line to compensate for any accumulated differences. This does mean that users will not be certain that the interword spacing they see on the screen will correspond to the spacing on the output. We will not know if this is a problem until we have more experience with Flexifonts under a variety of conditions. For most fonts, we do not anticipate too many problems.

For purposes of proof printing only, Flexifonts will also be digitized for a resolution of 300-dpi. However, it does not anticipate that these fonts will be used in applications in which the 300-dpi output is the final output. For these applications, it will continue to use specifically tuned laser printer fonts with matching screen fonts.

User Interface

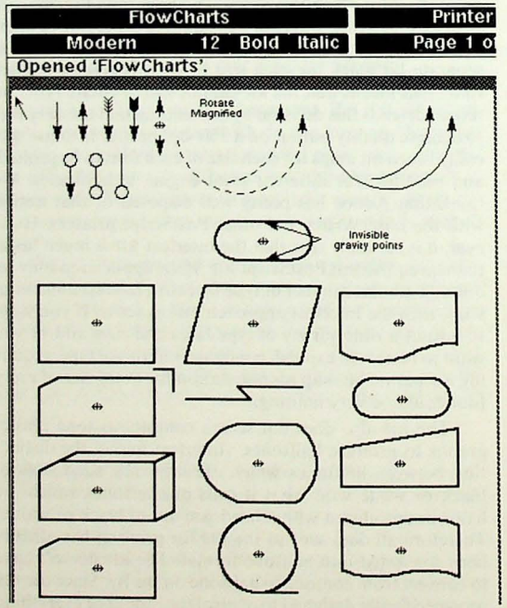
Interleaf has done an excellent job in providing a relatively easy-to-use user interface which doesn't compromise on speed and production performance. The workstation consists of a large (19") high-resolution monitor, a keyboard and an optical three-button mouse. The mouse works in conjunction with a small (about 10") pad which can be positioned anywhere on the user's work area. Although there are some functions which have short-cuts, single-keystroke capabilities programmed directly onto the keyboard, all the application functionality is directed with pop-up menus and icons on the screen and commanded by the mouse.

Desktop display. At the desktop level, the screen presents what amounts to the systems data base, organized within different icons. The screen icons include file cabinets, file drawers, folders, and documents. Theoretically, you can place documents within folders, folders within file drawers, and file drawers within cabinets. But in reality, there are no restrictions above the document level. You can place frequently-used documents or folders directly on the desktop. You can place folders within folders, or even file drawers within folders. As is typical for systems of this sort, when you "open" a cabinet, drawer or folder icon, the system draws a screen window which displays the contents of that icon.

Other icons include the clipboard for documents being held temporarily, read-only cabinets for graphics components used in drawing routines and templates used for text and graphics. There is also a "terminal" icon which allows the user to open up a screen window which functions as an ASCII terminal to the Unix operating system.

In many ways, the setup is similar to the familiar Macintosh desktop. The major differences are:

- The use of three different icons (cabinets, drawers and folders) rather than simply folders.



Graphics Cabinet. This cabinet displays a list of available pre-drawn flow chart symbols.

- Interleaf does not provide the option of different "views" (e.g., alphabetical listings) of the contents of a folder.
- It is a little more cumbersome on the Interleaf system to do things like moving icons around and re-sizing windows to tailor the screen presentation to your preferences.
- On the other hand, the large screen means that much of the desktop is still visible when you have a document open on the screen. This means that you can move off the document back to the desktop while the document is still open for editing on the screen.

  If you size your windows accordingly, you can open more than one document at one time and flip back and forth between them. (Something we do all the time on systems which support this function.)

Command menus. As with most mouse/icon systems, the basic method of entering commands is by pointing to the command you want in a pop-up menu. As has always been the case with any system which uses more than a single button on the mouse, it takes the user a little while to adjust to the different functions performed by the three different mouse buttons. Basically, the left button selects an object, the right button extends the selection, and the middle button causes a command menu to pop up at the current cursor location on the screen.

Having the command menu pop up at the cursor position (rather than having to move the mouse to the top or side of the screen to pull down a menu) is a common trick on systems with large display screens. (It is not much trouble to move the cursor to the top of the screen to execute a command on the 9" screen of the Macintosh. It is a real nuisance on the 19" Sun or Apollo screen.)

But there are three tricks Interleaf does that make the system much more effective for the moderately experienced user (at the expense of some confusion for the first-time or novice user):

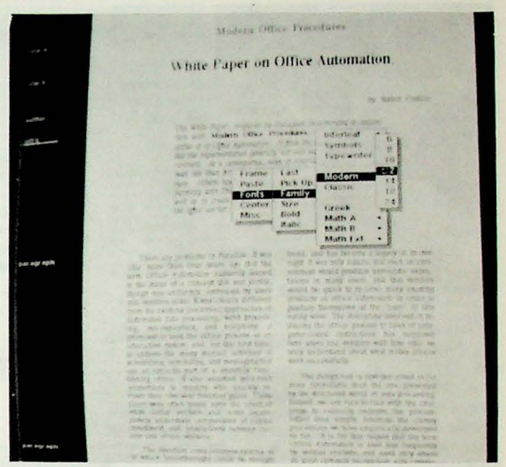
- Different menus will pop up when you point to different parts of the screen before pressing the middle mouse button. In all, there are over a dozen different sets of menus you can get by clicking the mouse button with a document open on the screen. Essentially, you determine what *kind* of function you want to perform by pointing to the appropriate place on the screen, then click the mouse button to execute a command.

  The menus which are displayed at any given time are related to either the object or element which has been selected and is presently "active," or the position of the cursor at the time the menu is requested. For instance, if a particular paragraph has been selected and a menu is requested (by hitting the middle button on the mouse), the displayed menu presents a list of options that apply to selected paragraphs. But if the menu is requested while the cursor is in the area of the Page Box in the header bar (information area about the document), options relating to pages are displayed.
- The command menus are relatively short. Simple, frequently used commands (such as "open," "cut," or "paste") appear at the top level of a menu. Other, more complex commands may require stepping through a series of submenus. But this is relatively quick, since the menus are chained sideways. Sliding the mouse pointer up or down in a command menu highlights the different commands, sliding it sideways opens up the submenus associated with particular commands.

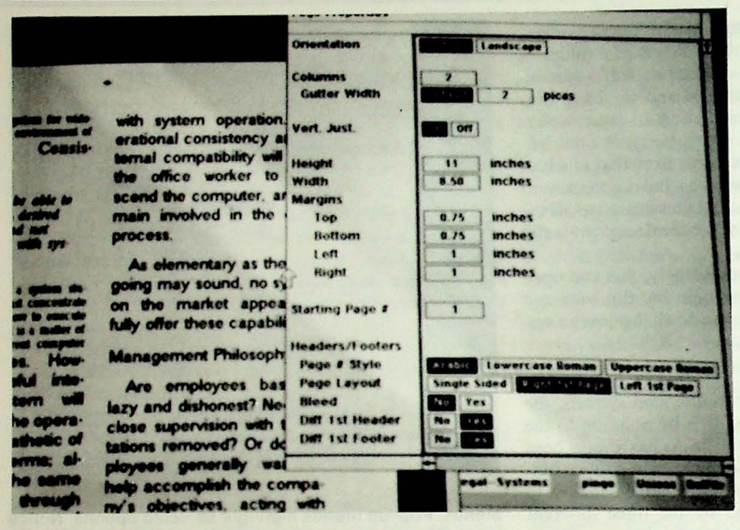
  This works well, except for the fact that the optical mouse pad tends to slip on a smooth surface. Since the mouse reads direction from a grid imbedded in the pad, if the pad is crooked, the mouse pointer will not move exactly as you expect. The solution is to keep the pad on a surface that will hold it firmly in place.
- Most important: the system is very intelligent about assuming defaults. It starts with the most obvious default operation ("open" a document if you are working on the desktop, "close" if you have a document open, etc.). But if you do the same function several times in a row, the system deduces that this is what you want to do and changes the default accordingly. Frequently, you will click on the mouse button and see the system open up a string of three or four chained menus which reflect what the system now assumes should be the default action.

  The result is that the experienced user soon learns he can perform many functions by simply clicking the middle mouse button. The system will not even bother to display the command menu. It will simply execute the current default function in a different direction.

  We do not think that this approach would work for a system which was designed to support a whole variety of application software programs. You would probably never remember the menu and default conventions for each program. However, it is very effective as a set of tools for



Menus. Pop-up menus highlight selected options. Notice the multiple "strung" secondary menus to the right of the primary menu.



Page Property Sheet. These provide the user with easy fill–in–the–blank menus to control the form and style of pages.

day–in, day–out use for a specific application. We have only two real complaints:

1. Changing from one type face combination (e.g., serif roman) to another (e.g., sans serif bold) is more cumbersome than we would like. As Interleaf adds support for more type faces, it may have to give a little more thought to the command procedures involved in this sort of function.
2. There are a couple of functions which should have two levels of protection but do not. It is far too easy to click on the "print" command, for example, and cause the system to print out your entire document when you really meant to click on the "page" command right below it.

In addition to the strings of command menus there are several commands which open up "property sheets" used to specify page and paragraph formats (more on these later).

We have watched several demonstrations of the Interleaf equipment and the multiple "strung" menus always appeared to be confusing and cluttered the screen. However, after using the system to prepare this article, we are very impressed with it. We found it to be an easy–to–use, consistent, and very fast approach to entering commands. It is one of the best menu driven user interfaces we have tried. We also found it easy to use and become accustomed to Interleaf's use of defaults. Since each menu has a designated default associated with it, the user can quickly select items from multiple menus without even displaying the entire sequence on the screen. The number of things you can do with a single mouse click is really amazing.

Property sheets

Document appearance is controlled by two sets of property sheets, one group which controls the form or style of a page and a second set which controls the typographic style or format of individual elements on a page.

Page properties. The properties of a page are controlled via a group of values and selections contained in a Page Properties Sheet. There is one property sheet for the entire document. However, different page styles such as multi–column and single–column pages can be produced via other means. The Page Properties sheet presents choices for the user which includes the following (see photo).

1. *Orientation* – The user can select either a landscape or portrait orientation for the document.
2. *Columns* – The user can specify both the number of columns he wants for the document and the gutter space between columns. The program automatically calculates (based on the page margins and gutter space) the measure of each column. In a multi–column setup, all columns are the same width. Currently, the user cannot mix different numbers of columns. However, within a multi–column document, the user can specify that specific components straddle the entire width of the page. This allows him to mix a multi–column format with a single–column format on the same page.
3. *VJ* – The user can turn vertical justification on or off for the entire document. However, there is no user control over where the justification space is allocated between elements.
4. *Page dimensions* – The user can specify both the size of the final page and the area on the page to be occupied by text and graphics.
5. *Margins* – Within the page boundaries, the user can control the top, bottom, left and right margins collectively for any and all text, graphics, footnotes etc. which are part of the document.
6. *Start page* – Since the document may be assembled as separate sections, the user may specify the starting page number for this part of the eventually grouped document.
7. *Page number and style* – The document can be specified as a single–sided document (all headers, footers, and page numbers the same on all pages), or left or right pages where the beginning of the document and chapters will always begin on even–numbered pages or odd–numbered pages. The page number may be specified as either Arabic, Lowercase Roman or Uppercase Roman characters.
8. *Headers and footers* – Headers and footers can be typed in any font and may be different for first pages versus all others. They can also be specified as displayed on the left, right or centered.

The headers and footers are created as graphic windows. In release 2.5 the header and footer styles are very much restricted by the limitations on text editing and text formatting in graphic windows. With the enhancements promised in Release 3.0 this limitation should pretty much go away.

Component properties. A second set of property sheets specifies the style for the components (paragraphs) of a document. Each type of component is assigned a name (which is displayed in a special area to the left of the page). A set of property sheets which specify paragraph style and paragraph–related pagination parameters is associated with each component type.

The user may edit these sheets at any time. Once he has done so, he may "apply" the changes he has made to that individual component. (In which case that component will have a different appearance than other components with the same name.) Or, he can apply the changes globally to all components with the same name any place in the document. (In which case the entire document may take on a completely changed appearance.)

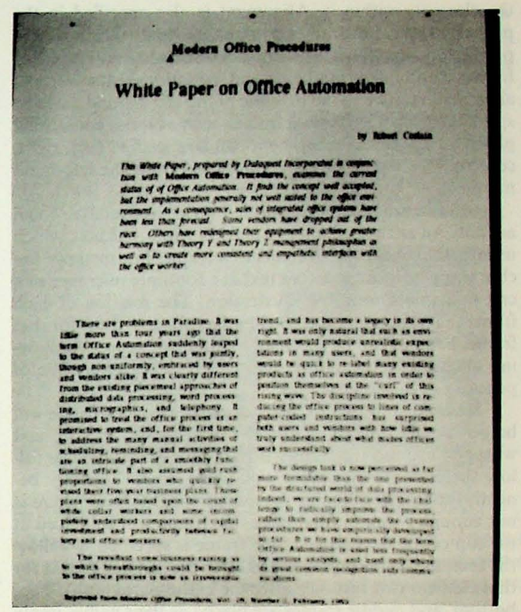
The user may create new component styles at any time by simply editing an existing component property sheet, and giving it a new name.

The component property sheets allow the user to specify paragraph indents (including different indents for first and subsequent lines in a paragraph), spacing above and below components, type face, point size and leading, tab settings, and paragraph–related pagination variables: "begin new page" with this component, "straddle columns," widow and orphan control, OK to break the page above or below the component.

Unfortunately, all of these values (except for line spacing) are specified in inches rather than picas and points, ems or ens. Even given the fact that the original intended market was office applications, we cannot see any reason for this. Granted, the user can call up a conversion chart which tells him what inch values to specify to get the typographic values he wants, but why should the user have to do this? One of the things a computer does best is calculate. It would have been so simple to let the user specify his own preferred units of measure.

Inter–line spacing is specified in terms of decimal fractions of the current point size. If you want eight–point type on ten points of leading you will have to specify an inter–line space of 1.26 lines. If you want ten point type on ten points of leading, you should specify an inter–line space of 1.02 lines.

The user can also specify ragged left, right, center or justified text, along with the level of hyphenation desired.


Modern Office Procedures

White Paper on Office Automation

by Robert Carlson

This White Paper, prepared by Dataquest Incorporated in conjunction with Modern Office Procedures, examines the current status of of Office Automation. It finds the concept well accepted, but the implementation generally not well suited to the office environment. As a consequence, sales of integrated office systems have been less than forecast. Some vendors have dropped out of the race. Others have redesigned their equipment to achieve greater harmony with Theory Y and Theory Z management philosophies as well as to create more consistent and empathetic interfaces with the office worker.

Page Styles. Property Sheets control the number of columns, gutter spacing, and allow selected elements to "straddle" multiple columns.

This can range anywhere from "full on" to "off" with nine levels in between. We will discuss this more fully in the section below on Composition.

A separate set of menus are provided to change the font and size properties of arbitrarily selected text within an element. For instance, to change several words within a paragraph to italic, the user would select the words as a selected text string and use the "text selected properties" menus to specify and apply the appropriate italic font. (Or, strike a dedicated command key on the keyboard.) These are not treated as fill–in–the–blank property sheets such as those above, but instead are individual typographic commands used to markup specific portions of text.

Frames

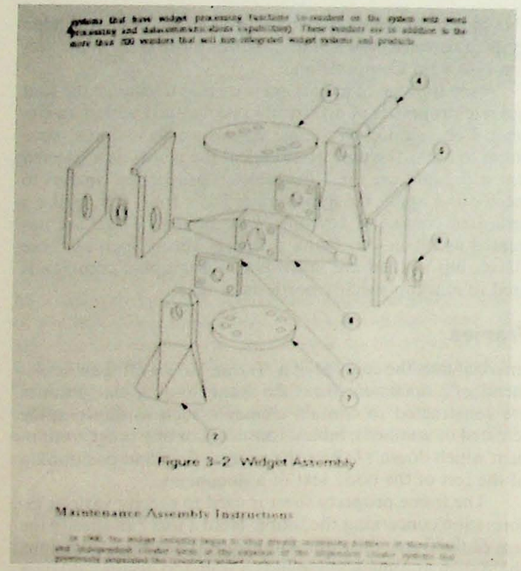
Interleaf uses the concept of a "frame" to set off specific elements of a document from the main running text. Frames are constructed to contain elements such as any graphic (created or scanned), tables, footnotes or any other component which doesn't follow the normal flow and positioning of the rest of the body text of a document.

The frame property sheet is used to specify various information concerning the frame. Here a user can specify the size of the frame in width and height (on certain occasions the system automatically builds a frame to accommodate a pre–drawn graphic and therefore determines the size to fit). The frame must be big enough to contain completely all of

the elements within it. Alignment is also specified in the property sheet. For frames containing elements that will be positioned directly within the text stream itself (such as frames containing mathematical equations), the vertical alignment relative to the baseline of the text around it can be specified. For more typical frames such as those containing photos, a horizontal alignment can be specified (left, right, centered or a specified value in inches from the left-hand margin).

Frames are always positioned on a page relative to an anchor. An anchor is simply a marker within the text, which usually references the contents of a frame. For instance anchors may be located within text at a footnote reference or a call to a photograph or illustration. The relation of each frame to its anchor is specified in the property sheet for that frame. Frames can be specified as being at anchors, following anchors or text, or at the top, bottom, or middle of pages.

If specified at the location of its anchor, the frame will be positioned directly within the text stream, with all text wrapped around the frame. If the frame is specified to follow the anchor, the frame will be positioned directly beneath the line containing the anchor. If, however, there is not enough room on that page, the frame will be placed at the top of the next page. If the frame is specified to follow the text, the frame is positioned after the last line of text for that element (for instance after the entire paragraph). Top, bottom, and middle frames are placed as their names imply on the same pages as their anchors. If there is no room on the same page as their anchors, they are placed in those named locations on the following page.



Frames. Frames can be used to set off specific elements such as this scanned piece of line art and its caption.

Pagination

The combination of document property sheets and the use of anchors and frames provides all of the user control over pagination. Obviously the program itself has its own set of rules and defaults on where and how it places the elements of a document. In general we found the user options coupled with the programs rule set to produce good-looking pages with logical placement of text elements and graphics.

The only problems arise when you decide that you want to do something differently than the program has done it. If, for example, you decide that you would rather place an illustration in the middle of a column rather than at the top of a page, you can change the placement specification for that illustration, but you cannot predict exactly how that change will affect pagination for the entire rest of the document.

Similarly, you have very little recourse if you want to control how space is distributed in a column or if you want to force the program to break at a particular point to solve a particular problem – especially since this program (like most pagination programs) works only one page at a time with no ability to adjust earlier pages to make things work out better on the current page. As with virtually any batch pagination program, there are times when you find yourself trying to resort to tricks to fool the program into producing the results you want. The difference (and it is a vital one) is that pagination takes place virtually instantly so you can see immediately if you have gotten the results you want.

In general, if your pagination needs are relatively straight-forward you will be delighted with the program. Even if there is an occasional thing you want to improve, you will find it quick and easy to do so. If, on the other hand, you want to do complex pages and you are very fussy about getting everything just the way you want it, you should probably look elsewhere.

Editing

Writing and editing copy on this system reminded us of working on the original Xerox Alto systems or (more recently) the Xerox Star. Like the Xerox systems, this is a full-page bit-mapped text editor. The screen display and the fact that documents are always maintained in paged form means that this is very much a page-oriented system. In normal operation, the user will view a page of copy at a time on the screen. Sometimes, when he is working with something which spills over from one page to the next, he will page down a part of a page so that the system displays the bottom half of one page and the top half of the next.

In general, you do not scroll through copy on this system, you page through it, a page or a partial page at a time. We found even ten-point type legible enough for comfortable editing on the screen, and we always like to work on a display which shows a page of copy at a time. It is much easier to view text in context on a large display, and it is quick and easy to zip around the page to make changes with the mouse. In general, we found that we made far less use of hard copy for review purposes than we do when working on system with partial-page display. Single keystrokes on the terminal keyboard will page you backwards or forwards a page at a time, and the paging is almost as quick as turning a page in a book.

In general, the editing functions work as you would expect on a system of this sort: you "define" a block of text by pointing to it with the mouse and clicking, then you delete, cut, copy, or change the style of the text you have defined. Since the system is always in insert mode, you insert new copy by simply clicking on the location where the copy is to appear, and typing in the new copy.

This is not to say, however, that the program could not be improved. We think some of the basic editing functions in the Macintosh MacWrite program are easier and faster to use. To replace a word in the Interleaf system, for example, you have to position the cursor at one end of the word, click the left mouse key, move the cursor to the right end of the word, click the right mouse key, strike "delete," then key the new word. To perform the same function on a Macintosh you simply point anywhere in the word (a quicker and less precise task), double click the mouse and type the new word.

We would also like to see Atex/Xywrite-like use of multiple save/gets, and single-key mapping for frequently-used components (paragraph styles), type font/size combinations and the like.

All editing functions work within a document as well as across documents. Therefore you could easily "cut" a section (any defined amount up to the size of the entire document) and "paste" it anywhere else within the same document or any other document. Depending on how this is done the pasted element can retain its own properties or take on the properties of the element into which it was pasted. The system also supports simple splitting and merging of text components. This makes it very convenient to modify or correct the structure of text that was improperly structured upon input.

These interactive editing tools as well as the ease at which one could quickly change the properties of individual elements, groups of elements or frames provide a very fast and flexible scheme for editorial intervention to resolve or change the pagination decisions of the software. In this regard the Interleaf system is very powerful.

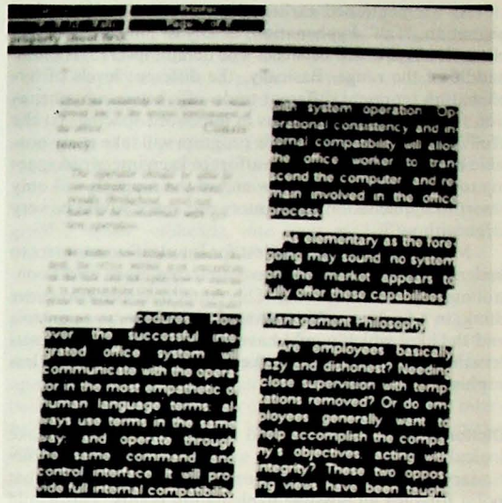
Composition

If Interleaf's strength is viewed as its user interface and interactive page assembly facilities, its weakness is still fairly unsophisticated hyphenation/justification.

The system provides the ability to specify the basics on a global or individual element basis: font, point size, leading, left and right indents, and quadding: ragged right, left, center or justify can all be specified. It does not yet provide support for more than a few type faces and point sizes, and it does not provide as precise control over interword spacing that commercial users have come to expect.

Nor does it provide the second level of composition commands such as kerning, tracking, inter-letter space control, and set size modifiers. Also missing are formatting capabilities and automatic features such as handling dropped initials or setting to fit. There aren't any true formatting capabilities which allow the user to build standard typesetting structures using macros and conditionals.

These are clearly less important in highly structured documents in the tech-doc field as long as there is a good pagination program to handle the logical placement of all elements. But they are important to some users who are more familiar with true composition and demand this level of sophistication. Most current Interleaf users are quite happy without higher quality composition, but it is clear that the company can expand its market if it adds the features that more sophisticated users want. (We have found that even "less sophisticated" users soon demand more typographic sophistication after they have gotten more experience with systems of this sort.) As of now it is not yet a system for quality typesetting.



Editing. In a typical "define-operate" mode, Interleaf highlights (reverses) text that is defined and subject to the next command such as delete, cut and paste (move) etc.

Hyphenation. Hyphenation and justification can be controlled at two levels. At the document level the user has the option of turning h&j on or off entirely. If turned on, he has the option to specify the number of consecutive lines that are allowed to be hyphenated. The user can select either one, two, three, four or "any." If he selects "any" the program allows hyphenation on any line of the document.

At the component level he can specify more precisely what hyphenation will be allowed in that component. Since component properties can be applied globally, the user may employ this capability to prevent hyphenation for all heads, subheads, intros and chapter headings and allow normal hyphenation for all body copy.

As we mentioned earlier, the user can specify no hyphenation, "full" hyphenation, or any of nine levels of hyphenation frequency between. The default level (5) is in the middle of the range. Basically, the different levels of hyphenation represent different trade-offs between hyphenation frequency and variations in interword spacing. At the "full" end of the spectrum, the program will take every possible hyphenation point in an effort to keep interword spacing to a minimum. At the low end of the range, it will only resort to hyphenation if the interword space would be very large without it.

Maybe this is what is easiest for Interleaf's customers to understand, but we would have preferred more direct control over interword spacing. This could have been done, we think, in a fairly graphic manner (with slider type controls and the like) which would have given experienced typesetters the control they would like without mystifying the less sophisticated customer.

Dictionary. Years ago, we had the idea that it would make logical sense to look up every word in a document in a dictionary *as it is entered* and carry with the word from that point forward all possible hyphenation points. This would permit you to do the dictionary look-up once when the computer had lots of time to spare, rather than doing it over and over again (during h&j) when computer resources are at a premium. It further occurred to us that the same look-up could also serve as a spell check.

It has taken a long time, but we are finally seeing this concept put into practice. Interleaf is one of the first companies to take this approach, a number of others (e.g., Itek, Ventura) are following suit.

The dictionary in this case is the 80,000-word American Heritage Dictionary provided by Houghton-Mifflin. It is supplemented (for words not in the dictionary) by Donald E. Knuth's TEX hyphenation algorithm. This algorithm is very conservative and usually does not hyphenate at all if there is any question as to the best location for hyphenation to take place. In addition to the software generated hyphens, the system allows the user to intervene manually and insert his own hyphenation points or remove software-generated hyphens.

The same dictionary is used for a spelling check program. Although common dictionaries for h&j and spelling checks is not our favorite approach, the spelling check program itself is very well implemented. Like everything else in the system, spelling check is essentially an interactive process. When invoked the program highlights the first occurrence of a misspelled word. You can quickly ignore and proceed, correct the word or temporarily add the word to the dictionary. as well

Other composed elements. In addition to straight body copy and heads there are three special types of data which a technical documentation system needs to handle: tables, footnotes, and math. Here again we will leave it up to the individual reader to determine how important each of these are. But for most customers there usually is some need for all three.

The current system treats all three types as elements within a frame. As we mentioned above, frames can be used to position text as well as graphics. However, text that is entered into a frame is limited in both the editing and composition features that may be applied. The text editor used for the normal body copy of a document does not apply to frame text. For instance, to edit a word in the middle of a footnote the user would have to delete all the text from the end of the line to that word before he could correct it. A similar problem occurs for specifying typographic changes to text within a frame. These can be done but the process is not nearly as easy, user friendly or complete as the excellent capabilities provided in normal text.

We would assume that the need to treat footnotes, tables and math as graphic data within a frame would make it more difficult to conveniently input this type of data embedded with the normal document copy. The lack of a simple translate or correlation table on input also makes it difficult to properly input tables with user-defined rows and cells. There is a tabular property sheet which will aid in the setting of tabular data in much the same way as a word processor would do. But data set in this manner cannot be ruled since it cannot be set within a frame. This is a far cry from the very sophisticated table-building capabilities such as Texet's. Here tables can be built purely from encoded gentags and dynamically grow and are ruled as a function of the contents of the table. These more sophisticated table-building functions also provide the ability to cross reference and number tables as a function of the section of the document the table is in. But if simple word-processor-style tables are sufficient, Interleaf can provide the tools for building them.

The most difficult of all may be math. There isn't any formatting language which allows the setting of math equations. Therefore, it is all but impossible to input the data of mathematical equations in a form that the system would understand and turn into something the writer had in mind. We found the only real way to handle math on the Interleaf system is to treat the math equation as a graphic. With the aid of a background grid the user can place the math symbols, one at a time into position as if building a complex graphic from graphic primitives. The tools are there to provide an accurate manual method for building these equations, but what the system clearly needs is an automatic approach based on treating the symbols as text, not graphics.

Table of contents. Although Interleaf hasn't yet implemented a totally automatic table of contents (TOC) function, it has a fairly workable semi-automatic approach. The system provides an easy method of selecting (globally) all elements of a particular type(s). Therefore, by selecting all major element types which are to be included in the table of contents and asking the system to build the TOC, the system will extract each item matching the selection criteria and list them in order in what amounts to a TOC format. The system picks up the page number for each item and adds it to the item listed in the TOC. It also automatically adds leader dots between the item and the page number. When the TOC program selects each item, it retains its point size and font as it appears in the document. Therefore, the only job that the user must perform is to add a consistent typographic style to the text.

Character sets. One final note that relates to the subject of composition: like most word-processing-based systems, Interleaf does not provide the full complement of characters which people in the typesetting industry are accustomed to. In particular, we could find no way of generating fractions. Foreign language accents are treated as supplemental characters on the screen and are essentially treated as floating accents for typesetting. These may not be important in some applications, but they are vital for us.

Input from PCs

Although Interleaf has focused on the tools required to create, edit and compose documents directly on engineering workstations, there are a lot of users who would like to be able to bring generic-coded text into the system from other sources – and especially from PCs.

Interleaf provides a program that runs against the incoming text and through global search-and-replace operations has the ability to convert the text strings to user-definable data. The user can also develop his own programs for dealing with this input translation problem. As we will discuss later, this general input processing can also be used to merge graphics into the appropriate locations within a document.

The system also provides another helpful data analysis function on data input from a PC or word processing environment. Interleaf uses an AI-type program as a generic input filter which analyzes incoming text based on characteristics such as margins, indents, font changes, carriage returns and spacing between elements. It will then make educated judgments on this criteria and group and tag elements with common characteristics. The program actually makes several passes on the data first to analyze the entire document for common characteristics and then to label common elements.

In effect this is automatic gencoding. We played with the program with both straight text and tabular data and found it to be quite helpful in sorting out common elements from raw text input streams. Paragraphs entirely typed in a separate font (bold) were labeled differently from paragraphs typed in the normal font. All elements with common indents were labeled as another common element type.

As nice as these facilities are, we still wish Interleaf would add a more conventional user-definable translate table which could be used as well. Interleaf does not, for example, provide a straight-forward input coding scheme for defining tabular data in specified row and cell format.

Gen code. Interleaf doesn't view its system as having a strict separation of form and content, but the system does allow the user the full capability of dealing with each topic separately. For instance, many pagination systems will create the form (style or envelope) first and pour the text into this mold. Although Interleaf doesn't work exactly in this fashion, the results are much the same.

When a text file is opened, it automatically has a default set of page and element characteristics. It is therefore treated as a document, complete with form, right from the start. To apply a standard, predetermined document format or style, the user simply merges a selected empty document with the now open text file. At any time the user can modify all or part of the properties of the page descriptions which regulate the style of the entire document. At this time the user can also rename, regroup, replace and assign properties to each of the major elements of the documents (paragraphs, heads, subheads, into lines etc.). The tools for quickly and easily accomplishing this are excellent.

While applying and re-applying page properties and element properties to the document, the batch pagination process is occurring, in real-time. There is no "paginate key." One of the things we are continually impressed with is the speed with which this real time, continually paginated operation occurs. As a result, there really isn't a separate batch vs. interactive process or operational mode. All interactive operations on the system invoke an automatic repagination process.

Graphics

One of the real strengths of the Interleaf system is its graphics handling capability. It has some of the most complete and powerful graphics tools of any text and graphics pagination system we have tested. The TPS software deals with four different types of graphics: data driven graphics and charts, vector art, CAD art, and scanned line art and contones. Some originate or are computer generated directly on the terminal while others are scanned or fed from external sources.

Scanned graphics. Interleaf currently uses the ImagiTex CCD scanner to input all scanned line art and contone images. Since all image processing takes place on the Interleaf system itself, it only interfaces with ImagiTex's Level 1 scanner (Level 1 is a simple scanner which generates an unprocessed stream of digitized data). A time-dedicated Interleaf "scanner server" or headless terminal acts as the controlling, buffering and storage device for all incoming data from the scanner. Once started, the scanner continually digitizes and outputs data at a steady rate. There is no start/stop capability so the scanner controller cannot be used for any other purpose while the scanner is running.

Although the scanner has the ability to scan at resolutions as high a 775 samples per inch, Interleaf encourages its customers to scan at much lower resolutions. Images which are to be output as 85 or 100-line-screen halftones are typically scanned at about 100 to 150 samples per inch. This greatly reduces storage for contone images to about 10K bytes per square inch; much less than the 1/4 MB-per-square-inch requirement if scanned at 475 samples per inch. This reduced sampling rate produces adequate results for both image manipulation and output (samples we viewed were produced on a Monotype Lasercomp typesetter).

For line art scanning, Interleaf recommends scanning at the resolution of the output device. In most cases this

means 300 samples per inch for laser printing and the ImagiTex maximum of 775 samples per inch for typesetting.

To further reduce storage and transmission of data between processors, Interleaf stores both line art and contone data in a compressed format. It claims compression ratios for line art up to 20:1 range. This is almost double what most vendors are getting from standard run–length encoding of data. Interleaf has a proprietary scheme which we have not been able to evaluate. It also compresses contone data and claims anywhere from a 2:1 to 3:1 ratio.

Scanned graphics processing. All scanned graphics are first buffered and stored on the scanner file server. The graphic can then be cut or copied from this data base and pasted into a text document. When this is done, the actual high–res format of the graphic is moved onto the disk which contains the rest of the document (including all text). Therefore graphic images are maintained (in all resolution formats) as an integral part of the document itself.

Processing of scanned images can occur on any workstation. Usually 4 MB of RAM are provided on workstations which are likely to perform graphics image enhancement and manipulation. This is sufficient space to deal with full–page photos in full true high–res formats. There are two schools of thought on image manipulation of contone images: 1) manipulate lower screen resolution images in real–time—then operate on the actual high–res file in a batch mode with a set of transaction instructions, or 2) operate directly on the high–res file in real–time. Interleaf has chosen the latter. Without any specialized high–speed bit–slice processors to aid in the high–volume number crunching, Interleaf has been able to accomplish a fair amount of image manipulation in real–time on the high–res files. This includes cropping, sizing, rotating and tonal adjustments.

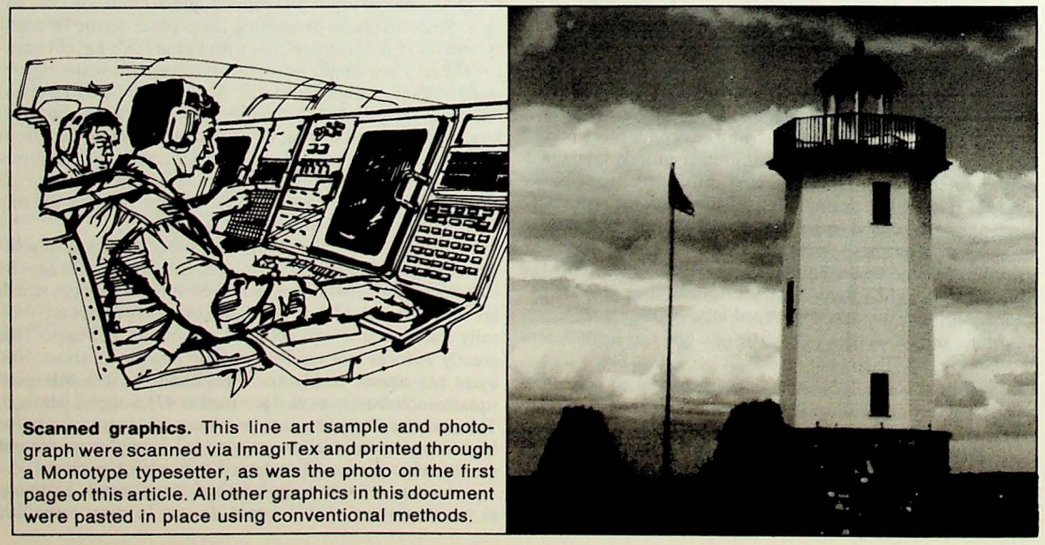
Currently only rectangular cropping is provided. A window is used to select the portion of the photo which is to be cropped. Sizing can be specified in inches or the image can be sized directly to the size of a frame (which can be built as a multiple of the column size). Rotation is very fast and very flexible. Rotation can take place about any point of the graphic or any other selected graphic in the frame. This makes it very easy to accomplish the desired results.

Tonal changes can be made to contone images via a contrast tonal curve. Multi–break points can be set individually on the curve so that different tonal portions of the photo can be adjusted separately. The system is not truly interactive. The user does not see the picture image change as he changes the tonal value curve. Rather, he makes a change to the curve, then "applies" that change to the picture image to view the results.

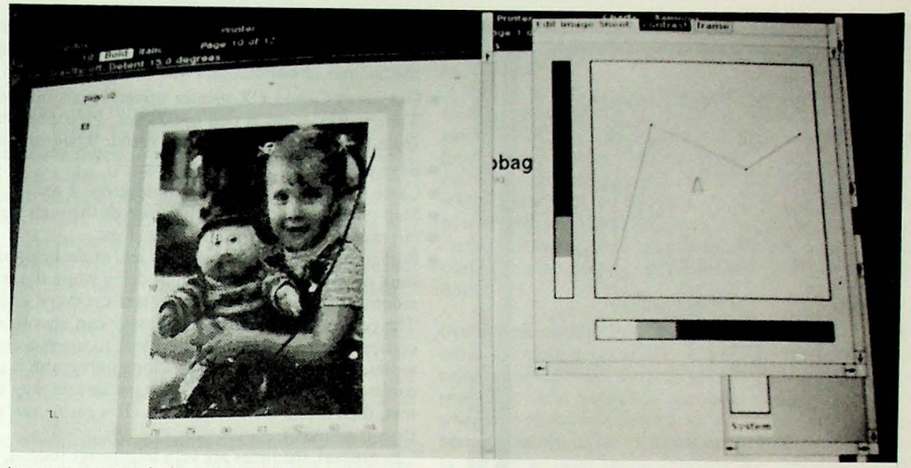
Pixel editing is now supported for line art only. Different brush sizes and shapes can be selected and used as paint brushes to paint either black or white onto a piece of scanned line art. The system also has a nice feature of painting with what it calls a null color which acts as an eraser in removing previously edited (black or white) information.

Vector art. Interleaf's set of drawing tools for creating drawings and diagrams are better than others we have tried. The program is somewhat MacDraw–like in that it creates resolution–independent drawings made up of layered geometric shapes filled with patterns.

The nicest features of the program as far as we are concerned are a "clip art" library of arrowheads, curved lines, flow–chart symbols, geometric shapes and other handy objects, and powerful ability to stretch and shrink these objects (or any other) into almost any imaginable size and shape. Flow charts, line drawings etc. can be created directly on the screen by connecting, distorting and combining several basic primitives. With the aid of the mouse, boxes, lines, ovals, splines, polygons and charts can all be sized, rotated, filled and positioned in order to create the desired line drawing.



Scanned graphics. This line art sample and photograph were scanned via ImagiTex and printed through a Monotype typesetter, as was the photo on the first page of this article. All other graphics in this document were pasted in place using conventional methods.



Tonal changes are made by altering the tonal contrast curve and then applying the altered parameters.

Another nice feature is the concept of "gravity." This means that lines which connect two objects will automatically attach to logical points (the center of an edge or a corner, for example) on each of the objects without overlapping or failing to butt.

There is an alignment grid and a variety of other aids which can be invoked. The functionality of their ruler and guidelines are not as complete as on other systems we have tried.

CAD graphics. Graphics created on other systems can be converted into the Interleaf graphic language for further processing and inclusion in text documents. Once "foreign" graphics have been converted into Interleaf language, the full set of graphic manipulations such as cutting, copying, sizing, rotating, grouping and editing (as an object) can be employed.

CAD art can also be input directly into the system. Data in 960 or 925 Calcomp plotter formats as well as IGES and HPGL format can be loaded directly into the system. However, Interleaf does not convert these into its own graphic language and thus can perform only limited manipulation of these graphics on the screen. The Interleaf system, for example would describe an ellipse as a single object which could be stretched, sized or otherwise manipulated. The same shape output from a CAD system in standard output format would consist of a series of short connected vectors. The notion that this collection of lines is an ellipse has been lost. A growing number of CAD vendors are writing filters to put their drawings into Interleaf vector format.

Charts and graphs. The fourth type of graphics which can be easily created on the system is data–driven charts. These are charts which are automatically created by the system based on data provided by the user. The data itself determines the size of each element in the chart. Charts can be created in one of three styles: bar charts, pie charts and line charts. In addition, bar charts can be oriented vertically, horizontally, or in a surface format.

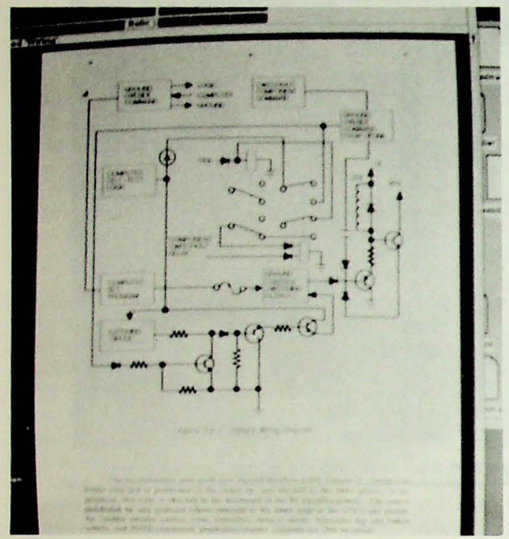
However, all of the Interleaf graphics can be combined within the same frame. The various types may be grouped for positioning as a single element, rotated as a group, or have certain elements of a group "locked" or frozen relative to others within the group. The concept of locking elements and treating locked and unlocked elements as a group provides an infinitely complex array of capabilities for the graphic designer. About the only function that could use some enhancement would be layering. Although any object can be placed behind or in front of any other, the concept of transparency and reversing text and images which overlap (*a la* Camex) isn't supported.

With all four types of graphics, Interleaf uses the "clipboard" to move graphics into or out of a document. A graphic is first cut from its original location, file or disk. It is then "pasted" into the final location from the clipboard.

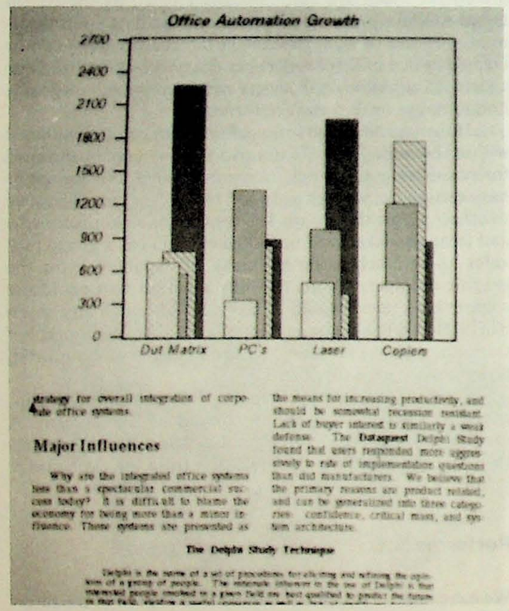
Performance

We have made frequent references to the speed of the Interleaf system. How fast is it? Here are some concrete figures. (All times were measured on a dedicated Sun 2/120 workstation using the document you are reading as a test document.)

- Copy file (to create a new version on disk): 10 seconds.



Created artwork. Schematics and data driven charts are both supported by the Interleaf system.



- Open document from disk (read from disk, compose into pages, and display on the screen): 26 seconds. This appears to be a fairly linear time. Doubling the size of the document almost doubles the time required to open and compose the document.
- Recompose the entire document after making a major format change: 4 seconds to 12 seconds, depending upon the change. (Four seconds was the most common time.)
- Output to Canon CX printer driven by an Interleaf RIP. Time workstation is tied up: 25 sec. First page emerges from the printer: 55 sec. Print speed: 8 pages/min.
- Save edited file to disk: 5 1/2 sec.
- Close edited file and return to desktop: 1 sec.
- Step one page backwards or forwards through file: 1 sec.
- Go directly to any specific page: 1 sec.
- Edit composed text: If you insert text at the beginning of a long paragraph there is a slight delay while the program recomposes the balance of the text on every keystroke. The constant rippling of the display can also be disconcerting. We found that the best way to insert a couple of sentences at the beginning of a long paragraph is to key a carriage return first to break the paragraph, key the new text, then delete the extra return. (It is easier and quicker than it sounds.)

In general, the times are very impressive indeed. We are not entirely certain why Interleaf does not take greater advantage of the multi-tasking nature of the Unix operating system. (Why, for example, should it have to tie up the workstation while it composes a job to be printed?) We suspect that this may be because the compose function requires lots of memory and does not leave many computer resources available to perform other tasks.

But this is a minor complaint. In general, the system is so fast that you cease to think about composition and page formatting at all. You simply work on the content of your document, and whatever you do, the document is always magically kept in pages.

Working on standard hardware

We have been saying for a long time that systems based on standard, off-the-shelf hardware were going to prevail in most applications. What does the user give up in comparison with a system based on hardware specially built for the application?

Although we have seen the Interleaf system running on a variety of workstations, most of our in-depth hands-on experience has been on the Sun 2/120 which Interleaf lent us to prepare this article. Based on this experience, we would have the following observations:

A. The keyboard is OK. The feel is similar to the clacky feel of an IBM PC keyboard. The layout is not outstanding, but usable. None of the function keys are labeled, of course, but you could add your own. In general, the layout is not as good as a layout designed for the purpose, but still better than the layouts on some of the dedicated systems we encounter.

B. The screen display is large and reasonably crisp, but as is typical with bit-mapped display screens, there is no anti-glare treatment. Almost no one has done as good a job as Apple (on the Macintosh) in providing a screen which combines a sharp image with some anti-glare properties.

C. The cabinet which contains the processor, memory and disk is moderately noisy—about on a par with most other similar-sized systems we have used. We think that the newer hardware (Sun 3s and Apollo 3500s) is quieter, but we have not yet had any of these in the same quiet environment.

D. Dealing with Unix for utility functions is a little bit of a nuisance. It would be nicer if these could be executed from the Interleaf desktop. On the other hand, we are impressed with some of the Unix facilities now available.

Although Berkeley 4.2 and Unix System V are greatly improved in this respect, we are still a little gun-shy about the vulnerability of Unix systems to data base corruption. We are very careful not to do things like inadvertently powering down the system while application software is running.

But given this caution (which, we will admit, is based on traumatic experiences with earlier versions of Unix), for most day-to-day operator functions, dealing with Unix is really not much worse than dealing with MS-DOS on an IBM PC system.

On the whole, our complaints are fairly minor. From the user's standpoint, about the only thing you really give up is a keyboard optimized for the application. We would willingly trade this to get the advantages of standard hardware and operating system software. (Besides, if a customized keyboard was the only reason for buying a specially-built system, it would be a lot cheaper for a vendor to supply a customized keyboard than it would for him to build an entire dedicated system.)

Release 3.0 features

We have dedicated the majority of this article to coverage of what really exists in the field, the software you could purchase and install today. However, Interleaf has shown us a number of significant improvements which address many of the short-comings of the current 2.5 release. We have therefore separated these features from the primary review of the product to minimize confusion of what is ready now and what is planned for release 3.0 in the fall.

All of the new features (which were shown publicly for the first time at the Seybold Seminars in March) deal with items which might be construed as short-comings in release 2.5 and earlier software. In general Interleaf has done its homework in identifying the areas needing attention and has developed an excellent set of new functionality to address these needs.

Autonumbering, cross referencing, and indexing. Until now the only autonumbering in the system was page numbering. There wasn't any way of numbering sections and subsections of text in the document, footnotes, tables and charts or figures and illustrations. The new functions handle all of these, consistent with the rest of the Interleaf system.

With the addition of newly created property sheets labeled by name (Interleaf calls these numbering sequences "streams") the user can quickly and easily setup the sequencing structure and the style of the numbering tags (numbers, alphas etc.). A user would establish a numbering stream for each category of elements he wishes to be numbered. Therefore he can set up a numbering stream for all heads and paragraphs, a separate one for all figures, and another for tables.

Once the properties of each stream are established, the system automatically numbers the elements and sub-elements of the entire document. As the user inserts numbered paragraphs, deletes them or moves them, the system dynamically renumbers them based on their relative location within the numbered sections. The user also has the ability to intervene and manually restart the numbering scheme for certain elements such as numbered lists where the same numbering scheme may be repeated several times within the document.

Cross references. For charts, photos and footnotes the cross reference scheme is excellent. The system automatically numbers items such as photos and inserts the corresponding cross reference numbers directly into the text where the photo was referenced. As with all other autonumbering streams, as the elements are moved to other sections or within sections the system resolves the numbering sequence automatically.

Since the numbering scheme is based on a property sheet which can define the style of the numbers themselves, we found it very easy to use the numbering scheme to set up outlines. The system automatically numbers in the user-specified style (Roman followed by dashes etc.) and indents the number for each element of the outline automatically.

Indexing. Another attractive new feature is indexing. In a matter of seconds, the system will automatically build indexes (one at the moment) based on markers (and corresponding property sheets) placed within the text. Interleaf indexes the ideas behind the words rather than the actual text which appears in the document. Markers placed in text indicate the point at which the reference occurs. The reference itself is entered into a property sheet associated with each reference marker.

In essence, the user enters the index terms into the property sheet. In this menu the user would type what is being indexed at this point. In other words, he is typing the text that will actually appear in the index when it is built. He also specifies a "range" which indicates how many pages are part of this indexed item. Ranges are not restricted to number of pages but can be expressed in more meaningful terms such as "for the rest of this section or chapter." In this way the index program will figure out the actual pages to be included in the index once the document is complete and truly ready to be indexed.

The indexing handles all of the special conditions common to most indexing such as multi-level sub-elements, "see" and "see also" references. It provides a reasonable sort algorithm which will cope with text as well as numbers. Since the index is built as another set of pages within the document, it can be edited and formatted like any other portion of the document.

The book. Along with the autonumbering, cross-referencing and indexing functions comes the concept of a "book." Until 3.0 the largest structure contained in the system was

the document. The entire document was one piece. Its format was consistent for the entire document.

With the concept of a book, the structure of the system has been elevated one level. In essence, a book is a collection of documents, each of which could have been built separately with radically different document formats. Therefore, part of the book may come from one document which was built in a two–column format, while another may have been built as a three–column etc. However, for the purposes of page numbering, cross referencing, indexing, and autonumbering, all of the documents must be treated as a single entity.

Math. Another weak link in 2.5 was the handling of mathematical equations. With the new release, math will be handled more as text than as a graphic. Equations will still be placed in frames and positioned as any other framed item. But they will be built from input text with imbedded commands modeled after the Unix EQN program.

The Interleaf software uses essentially the same commands as EQN. Text is input as a coded file, then the system composes the equation and displays it in graphic form on the workstation screen. It he/she does not like the results, the operator can correct the input text and redisplay it on the screen.

Each expression will have its own property sheet. Within this the user would type the equation including these special mark–up commands which add form to the equation. At the present time the special Greek and math symbols must be entered as words (such as lambda)—which is what EQN had to do because it was written to accept input prepared on ASCII terminals. Interleaf intends also to make these special characters available directly from its keyboard.

Once the property sheet is filled out and stored, the equation is automatically built and displayed within its frame. At any time the user can return to the property sheet to correct any text or formatting errors, or can graphically manipulate the math expression within the frame. At this time the user cannot edit equations directly within the document. He does this through manipulation of values in the equation properties sheet. However, he can size it there, move it into position, and do any other function allowed for graphic elements within a frame.

For the convenient inputting of mathematical data, Interleaf will set up special filters or translate tables to convert incoming data into a format which is consistent with this approach.

Text manipulation within frames. One of the most powerful additions to the system, and one which generated a lot of excitement at the Seminar, is the ability to completely and freely manipulate text within frames. The easiest way to state this is to say that any function that could be performed on the text of a document, outside of a frame, can now be performed within a frame as well, but this understates the true capability.

Not only can the user perform all of the standard Interleaf text functions within a frame, he can also perform much more free–form composition which will allow him to cope with all kinds of pages which are currently difficult to do on the Interleaf system. The demo version of this, at least, was very impressive. The user can dynamically stretch or shrink a column measure, for example, and the text is recomposed interactively to fit the changing boundaries.

Further, in the future the new text manipulation software will allow users to link arbitrarily–sized copy block areas (*a la* Aldus PageMaker) and flow text over several pages. This should give Interleaf the ability to cope with magazine pages and other free–form documents which the current system does not address at all.

Bitstream type. For composition of display type and headlines within a graphic window, Interleaf will support Bitstream font masters in outline form. The outline font masters can be manipulated like any other graphic element. Then, the system will fill the character outlines.

Enhanced graphic manipulation. Release 3.0 will also include significant enhancements to the already impressive capabilities for manipulating graphics.

Spell check. The next version of the spell check program will include a pop–up menu of system–generated suggestions on the spelling of words which are not in the dictionary.

Pagination. Finally, the release 3.0 page formatting property sheets will give the user more flexibility and control over the vertical justification process. The user will be able to control the percent of feathering as well as define when (by percentage of a page with non–breaking elements) VJ is allowed to be used.

Not included. As we mentioned above, the new feature and functions of release 3.0 go a long way in addressing the weak points of an already strong release 2.5 product. There are a few areas, however, that have not been completely addressed in this next release:

A. The system will still lack a full complement of quality composition features. It does not provide any direct user control over interword spacing, any control over letterspacing, and no kerning, tracking or modifications to set size.

   Interleaf has not added these features as yet because its existing customers have not demanded them. We would expect that this will change since Interleaf is planning to extend its market focus to include users with more commercial–quality typesetting concerns.

B. Another area which has been lightly addressed is user control over the pagination rules. The user is still not given some of the control more sophisticated customers often want over this process.

C. Finally, the area which must be considered the weakest link in the Interleaf product is its ability easily to handle tables. There is the possibility that some automatic table handling will be ready for 3.0, but we haven't been able to see any as yet. However, other added features in the system, most notably the ability to freely manipulate text in

A look at Interleaf

Product line. Document processing and pagination system which is designed to run on a variety of off–the–shelf hardware workstations. Currently, the system is configured to run on Apollo, Sun, DEC VAX, and the new IBM RT PC.

Interleaf sells both software and turn–key hardware /software sysems. Interleaf OEM's (except for Kodak) sell Interleaf software to run on their systems.

Workstation Publishing Software (WPS). This has most of the functionality of the more complete TPS software but differs primarily in the fact that it is not configured to handle scanners or typesetters. Other than Kodak, this is the only product which is being sold by Interleaf's OEMs.

Technical Publishing Software (TPS). This is a super set of the functionality included in the WPS system. In addition to all WPS software features, it handles scanned line art and contone data from an ImagiTex scanner and has the support of Interleaf's communications group for customizing input filters (translation programs) to allow easy link–up to foreign systems.

Input and output hardware. ImagiTex scanners are used for input of graphics. Output is generally to 300–dpi electronic printers driven by an Inerleaf RIP. Interleaf sells the 8–ppm Canon LBP CX and the 26–ppm Dataproducts laser printer. It also sells Imagenn printers driven by an Imagen controller. Other output devices supported include the Monotype Lasercomp typesetter and III Comp 80.

The Principals. Three members of the Interleaf board who are also officers of the company are co–founders Dave Boucher and Harry George, and George Potter. Boucher who is president of Interleaf, was a co–founder of Kurzwiel where he served as Vice president of Manufacturing. George, also a co–founder of Kurzweil was both Controller and Vice President of Finance and a member of the Board. He also served as Chief Financial Officer of Micon Industries. George Potter presents a familiar face in the publishing world having served as National Sales Manager for Publishing Systems for Atex and later as District Sales Manager for Wang. He also held management positions in marketing and sales for both IBM and Inforex.

Financial History. For the its second fiscal year (ended March 31), Interleaf expects to report about $18.6 million in sales, more than double the $8 million it booked in its first fiscal year.

Sales and Support. Interleaf sells both direct as well as through a group of domestic and internation OEMs. It currently has offices in 16 US cities and 2 more in Canada (which are partially owned by a third party.) There are a total of 25 salesmen working out of these offices. Most offices also have application and services engineers to support equipment in the field. In Europe it is represented by:

Scandinavia: Nokia Corporation, Electronics (Findland). *France:* Institut de Genie Logiciel. *Belgium:* Belgian Institute of Management. *United Kingdom:* Amazon Computer Ltd. *The Netherlands:* Betagraphics, and Koning en Hartman.

All of these are distributors of either Sun and/or Apollo Workstations. An active project is underway to provide true language support for systems sold in these countries and is expected by late '86 and early '87.

Currently, over 70% of Interleaf's sales are complete systems including both hardware and software. Only TPS software is supplied with system sales. The remaining are purely software sales to customers with existing Sun, Apollo or DEC workstation and may be either WPS or TPS software.

Of its OEMs, only Kodak currently sells TPS (and only TPS) software. Others such as DEC, Computervision, and the CAE Division of Textronix sell only WPS software. For now, IBM sells WPS. This summer Interleaf expects to have its TPS software running on the IBM RT PC. It also hopes to become a full IBM VAD (value added dealer) by then. This will allow Interleaf to sell the IBM hardware along with its Sun, Apollo and DEC offerings.

There are other less formal arrangement which Interleaf has (mainly with CAD manufacturers) and other OEM deals which appear very close to fruition.

Interleaf
Ten Canal Park
Cambridge, MA 02141
(617) 577–9800

frames, will, by itself, greatly improve the table-building process.

User comments

The Interleaf customers we have talked with are generally very pleased with the system. Most of the comments are along the lines of "best thing since sliced bread." In general,those who have purchased the system for the purpose it was intended (technical documentation reproducible on high-quality plain-paper output devices) are exceptionally happy. They all spoke of significant improvements in the quality of their documents (many had been outputting on dot-matrix or daisywheel printers) and were surprised in actually achieving the "order of magnitude" improvement is speed and turn-around.

When asked if the system could be improved the answers were also very similar. The most common reply was the need for more fonts. Everyone agreed that the quality of the existing fonts was equal to or better than they had anticipated from a plain-paper, 300-dpi device. Many use the system for documents which used to be set outside by commercial typesetters. But everyone wanted a larger number of fonts to choose from.

It almost sounded as if users have been teased with quality, almost-typographic quality fonts, and now want the larger libraries of fonts normally associated with typesetters. However, most of the users we talked to were not typesetting documents before and were not trying to compare the quality or features of the Interleaf system with commercial typesetting systems. As you would expect, those users that had come from a more typical typesetting background did ask for more composition features.

In addition to wanting more fonts, the users requested parallel improvements in system functions. Most were very pleased with the general functionality of the page make-up, editing, and graphic features. The most frequent comments were directed at the need to improve footnotes. The current system requires non-automatic work-around procedures to properly insert, number, edit and position footnotes. The biggest concern seemed to be positioning footnotes on the same pages with running headers and footers.

The rest of the concerns seemed to be in line with what Interleaf itself recognizes as its areas of needing improvement. The same topics of footnotes, tables and numbering are clearly the targets in Interleaf's next release of software. In this regard, Interleaf appears to be listening to its customer's needs.

Larger systems

As we noted in the introduction, most Interleaf systems sold to date have been relatively small configurations. However, some users are moving towards relatively large networked configurations which include Interleaf software. Note, for example, the Apollo installation described in our Seybold Seminars report. We do not think that Interleaf itself is likely to be the sole turn-key supplier of systems of this sort. Rather, they will be good-sized networked systems which support a variety of application software, including Interleaf WPS and TPS software for composition, graphics and page output.

Such a system will rely on the basic Apollo, Sun, or DEC file management capabilities. It does not have the more elaborate database management facilities described by Caddex, Context *et al.* in the Seybold Seminar presentations. However, it is a direction that Interleaf says it plans to address.

Pricing

Standard Interleaf pricing (as of May 1, 1986):
WPS software for other workstations: $3,000.
Volume discounts available.
WPS software for IBM PC RT: $1,995.
TPS software for any configuration:
$20,000/terminal with disk.
$10,000/diskless terminal
A typical four-workstation system with a 515-MB file server with 1/4" tape back-up and a Dataproducts 26-ppm printer: $140,000.
ImagiTex scanner (completely interfaced but not including controller): $40,000.
Installation and support: Interleaf offers a three-day training secession with all sales: $1,000.
Hardware and software maintenance: Generally 1% of total system price. This includes on-site hardware maintenance, toll free telephone support (application and hardware), and all future product enhancements (2.5 customers would receive, free of charge, all standard [non-optional] features of release 3.0 when these are made available.)
Software maintenance only: 1 1/2 % of software charge. This includes toll free support and future software enhancements as above.
A complete single-user starter configuration with either a Sun or Apollo desktop workstation, 86 MB hard disc, 1/4" tape back-up, Interleaf RIP, Canon CX printer driven via a single serial port, and full TPS software: $29,995. (This is a single-user system without any network interface.)
Individual pricing: Sun or Apollo desktop workstation with TPS software: $25,000.
Interleaf RIP and Canon printer with a serial port interface: $8,000.
Interleaf RIP and Canon printer with a parallel port interface: $10,000.
Dataproducts 26-page/min. printer with Interleaf RIP: $27,500.
An alternate laser printer to complement a configuration to provide full function final publishing (Imagen with Interleaf fonts): $11,500.

Conclusions

There is little doubt in our minds that the Interleaf system is powerful, fast, easy to use, and produces an excellent product on laser printers. For pagination of structured documents it is a pleasure to work on and the quality is excellent.

It is also clear that the current system has some limitations: footnotes, tables, automatic numbering and math. Most important, it really only supports a few type fonts in a few sizes on a few output devices.

Release 2.5.75 with Flexifonts should go a long way towards giving Interleaf the ability to drive a wider variety of output devices with larger font libraries. The additional capabilities of release 3.0 will address many of the remaining short-comings of the current system. You might argue that release 2.5 software is still essentially a super word processor with exceptionally good facilities for handling graphics. We do not think that you will be able to make the same argument about 3.0.

This does not mean, however, that the Interleaf system will mature into one suitable for highest-quality book composition. Interleaf is concentrating its development efforts on the types of functions its customers say they want. Truly high-quality composition is not a high priority in this market. For the most part, customers simply want to produce good-looking, effective documents quickly and easily. This Interleaf is certainly able to do.

In all, this is a very impressive product. There are very few systems at any price which offer as complete a set of tools for producing completely paginated documents with text and a full range of graphics. If Interleaf's capabilities meet your needs, the system is hard to beat.

Bill Solimeno
Jonathan Seybold

Interleaf and a Data Base Application

During the Seybold Seminar, we looked at an interesting intersection of two technologies: the Interleaf publishing system and the *Isocrates* CD-ROM data base system. It so happens that the Classics Department of Brown University was one of the beta sites for the IBM RT computer and the WPS software that is being offered on it. Of course, the classicists were also familiar with the *Thesaurus Linguae Graecae*, a compilation of the full text of nearly all known Greek literature written before 700 A.D. It includes over 170 authors—poetry, drama, philosophy, history, and religious writings such as the New Testament.

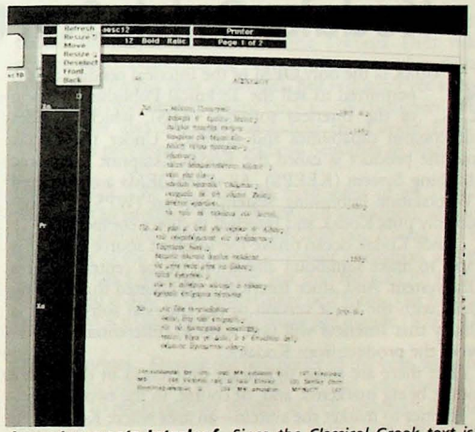
The TLG data base, which takes about 250 megabytes of storage, had previously been indexed by the Harvard Classics Computer Project. The indexing data takes an additional 150 megabytes. Harvard had also written search software designed to run on a variety of Unix systems and to support any display or printer capable of imaging the full Classical Greek character set. The folks at Brown University decided to convert the data base to a CD-ROM, to run the data base retrieval software on the RT, and to use the Interleaf software in publishing treatises on the ancient texts.

It is, so to speak, a classical application of data storage and retrieval techniques. A user can search the data base by author or keyword, the only complicating factor being the use of the standard English alphabet to transliterate classical Greek, since the retrieval software runs under Unix. Even that complication can be smoothed out: Apple Macintoshes, using a Greek font and MacTerminal, have been used as Greek terminals for the RT.

Having searched the data base and found something quotable, the user simply marks the beginning and ending of the text selection. The software then copies the text to a Unix file in Interleaf-readable format. When the user exits Isocrates and starts the Interleaf software, the file appears on the user's desktop and may be treated like any document.

The Brown scholars concocted 10- and 12-point Classical Greek fonts out of the Math Greek symbols provided by Interleaf. The principal issue was how to treat the accented characters; Classical Greek vowels may take seven accent marks appearing in eleven positions and in multiple combinations. The scholars elected to make each combination a separate character. This implied that there would be about 220 total characters, so each Classical font was handled as two Interleaf fonts. With that problem solved, the rest of the implementation was straightforward.

The result: the scholar can freely intermingle Greek and English text in a document, with a considerable degree of assurance as to the authenticity and correctness of the Greek. And shall we doubt that this can only improve the quality of academic publishing? Nay, perish the thought.



Isocrates meets Interleaf. *Since the Classical Greek text is extracted intact from an authoritative data base, the odds of introducing printer's errors are much reduced. The proofreading of this text is left as an exercise for the student.*