

VOL. 3, NO. 12

ISSN: 0887-3054

DECEMBER 1988

I N S I D E

2ND FEATURE

Maturity Beyond Their Years Page 11
Sun's and Apollo's service and support programs show surprising sophistication.

EDITORIAL

Unix and Camelot Page 2
Does AT&T's might make it right? Or is OSF right to try to relieve AT&T of some of that might? In our editorial, we look at the dynamics of the Unix marketplace.

NEWS ANALYSIS

AT&T wins the AFCAC 251 bid • APT workbench from Sybase • Progress Fast Track application builder • Digital licenses Locus's PC-Interface and Xsight products • Motorola supports DOS-to-Unix conversion through XDOS from Hunter Systems Page 16

UNIX IN THE OFFICE

PRODUCTS • TRENDS • ISSUES • ANALYSIS

Document Processing with Interleaf

Is It More Than a Niche Application?

By Ronni T. Marshak

DOCUMENT PROCESSING IS the new industry term for a not-so-new concept. Some consider document processing to be "word processing meets desktop publishing." Well, sort of, but there is much more to consider (see the *Office Computing Report*, Vol. 11, No. 10). Object orientation, compound documents, and structured editors are all part of document processing.

(continued on page 3)

AT THE END of the year, it is good to step back and look at where we stand. Taking a step back may be precisely what is needed in the contentious and confusing Unix marketplace. Why this confusion and fighting? We think it may have something to do with the dynamics of control. While industry watchers are observing the battles between the upstart OSF (Open Software Foundation) and the establishment AT&T and its new found ally, Unix International, Incorporated, something more subtle is going on.

Just a year ago, for example, it appeared that Sun Microsystems was clearly in charge of the future of Unix. At least, Bill Joy believed that he was best suited to bring Unix to its next generation. His logic was fairly widely accepted. And indeed, Joy and Sun have had some remarkable achievements. Some of these have been in the PR arena, others have been technical. Sun technocrats were eager to promote the idea that Sun had taken the mantle away from father AT&T, who had somehow grown old and brittle with time.

It is sometimes easier to explain complex situations with an allegory. So let's look (for a moment only) at these happenings as though they were taking place in feudal England around the fourteenth century. Under these circumstances, we'd see our friends at AT&T transformed into the seasoned baron taking his favorite, newly matured son, a knight, onto the battlefield for the first time. The dynamic created by this duo is met with fear and trepidation by the masses and results in a rapid and radical power shift. Enter the rebellious knights. They, who had long supported the baron with their purses and cheers, are also suddenly fearful. What if this partnership should grow so powerful that their importance is no longer recognized? What if their influence should wane? After all, these knights consider themselves as capable as the upstart son of the baron.

In this emotionally charged atmosphere, the course of

• E D I T O R I A L •

Shifting Dynamics

By Judith S. Hurwitz

action is clear—revolt. The masses are ecstatic. Finally, someone is looking after their interests. Someone understands and is going to do something to make things better. So, for a time, popular support lies with the revolutionaries, who stir the hopes and dreams of the masses.

But, as in any revolution, when the masses realize that all dreams take time to be realized, the luster and the excitement begin to wear thin. Those who had relied on the baron for their protection and livelihood become frightened. Isn't it better to stay in the good graces of the baron than to risk ruin? Maybe that old guy wasn't so bad after all. Anyway, they think, the attempted revolution certainly made him see the error of his ways. Yes, he's a wiser man today. We can trust him to do the right thing.

Now we jump back into the twentieth century. Curiously, the revolutionary knights of OSF represent the best and worst of the Unix marketplace. They are the best because the long-term goals and ambitions of the organization are in the best interest of the industry. The idea of bringing the newest innovations from research laboratories and universities into the mainstream is a lofty goal. It is also ambitious to try to have a group of people create an operating system that meets the needs of the commercial marketplace. But OSF, as a move to seize power and create yet another variation of Unix, also represents the worst. Commercial users are even more confused and worried about the viability of open systems.

So, as we move into another year, what can we expect? Well, the furor over control will probably begin to die down. Both sides understand the stakes; neither wants to kill the goose that lays the golden eggs. AT&T, in the form of Unix International, is clearly staking out the operating system as its only turf, and this clearly leaves room for a compromise somewhere down the road. We hope that the coming year in Unix will be a little less turbulent than the one that is passing us by. ☺

Patricia Seybold's
Office
Computing
Group



Publisher PATRICIA B. SEYBOLD

Managing Editor
RONNI T. MARSHAK

Senior Editors
MICHAEL D. MILLIKIN
JUDITH R. DAVIS
Telephone: (617) 861-3926

Associate Editor
JOHN N. KELLY

News Editor
DAVID S. MARSHAK

Editor-in-Chief JUDITH S. HURWITZ

Contributing Editor
GARY J. NUTT
Telephone: (303) 444-0341

Assistant Editor
LAURE BROWN

Sales Director
RICHARD ALLSBROOK JR.

Circulation Manager
DEBORAH A. HAY

Customer Service Manager
ANNE L. WORKMAN

148 State Street, Suite 612, Boston, Massachusetts 02109 Telephone: (617) 742-5200 FAX: (617) 742-1028

Patricia Seybold's UNIX in the Office (ISSN 0887-3054) is published monthly for \$495 (US), \$507 (Canada), and \$519 (Foreign) per year by Patricia Seybold's Office Computing Group, 148 State Street, Suite 612, Boston, MA 02109. Application to mail at second-class postage rates is pending at Boston, MA and additional mailing offices. POSTMASTER: Send address changes to Patricia Seybold's UNIX in the Office, 148 State Street, Suite 612, Boston, MA 02109.

• INTERLEAF •

(continued from page 1)

Document processing has been available on Unix for some time. Interleaf Incorporated (Cambridge, Massachusetts) is an old-timer in the business—seven years old. The company develops and markets workstation publishing tools running under Unix. But does the product have more universal potential than the publishing niche?

A Bit of Background

Way back in 1981, a full three years before the '84 Olympics-timed introduction of the Macintosh, Interleaf Incorporated was founded by David Boucher. Boucher had been intrigued by the Xerox Star in much the same way as had Steve Jobs (father of the Apple Lisa and Macintosh). But instead of developing a hardware/software environment for applications development, he determined that a WYSIWYG object-oriented environment would be perfect for publishing applications.

Interleaf's first product was called Office Publishing Software (OPS). Intended for the office market, the product stressed ease of use and a visual interface. Unfortunately, the cost of the system was prohibitive for office users. A stand-alone system, running on a Sun 100 with a 30MB disk and 2MB RAM and a laser printer, cost \$90,000. Not your average desktop budget.

So how did the company make any sales? According to Larry Bohn, vice president of Management and Planning, "Interleaf's success is that it has followed publishing economics." In other words, when you have a system that costs \$90,000, you sell it to a market that can cost-justify a \$90,000 standalone publishing system. In 1983, the appropriate market was technical publishing, where the need for good publishing software running in Unix, the most popular environment, was desperate. In 1984, OPS was renamed WPS (Workstation Publishing Software), and, within the next year, the company branched off a new product, Technical Publishing Software (TPS), aimed directly at technical documentation. TPS is currently the main product, though others have been added over the years, including Multi-Language Publishing Software (MLPS) and Interleaf Publisher, co-marketed by IBM for the RT and PS/2 386 line.

Boucher gambled heavily on what he believed to be the future of Unix and 32-bit processing, anticipating a dramatic decline in price. And his foresight has paid off. What used to cost \$90,000 can now be bought for \$10,000, not to mention five years' worth of software improvements.

FINANCIAL CONDITION. 1988 was a good year for Interleaf, its second as a public company. Before going public, the company was in its R&D stage and was steadily losing over \$1,000,000 per year. In 1986, a particularly bad financial year for the company, the net loss was \$2,443,401. In 1987, its first public year, Interleaf lost \$410,121—still no profit, but quite an

improvement. 1988 saw a profit of \$6,979,897. Revenues were up 57 percent to \$58.4 million. However, stock prices have gone down significantly, probably due to overvaluing in previous years.

PRODUCT PHILOSOPHY. The Interleaf product line was designed to do more than simply integrate text and graphics. It is intended to integrate the editing process with the composition process. The traditional time line in publishing used to be: (1) editing, (2) page layout, and (3) graphics insertion. The Interleaf product is a "modeless environment," where all three processes are handled simultaneously on the same screen. In 1988, this isn't unique. In 1983, it was pretty hot stuff.

Multiple Platforms. The products have also been ported to a number of different platforms, from mainframe (IBM) to mini (VAX and Apollo) to Unix workstation (Sun3, 4, and 386i, and the IBM RT) to PC (Macintosh II and any 386-based PC). All files are compatible across platforms.

Interleaf also provides filters to and from popular word processing and graphics formats. The following filters are included in TPS, Interleaf's flagship product:

Standard text filters

- Digital DX
- DCA
- MacWrite
- MultiMate
- nroff
- Stripfilt (strips out Interleaf ASCII markings)
- troff
- Wang
- Xerox 860
- Scribe
- WordStar
- ASCII

Optional text filters

- Microsoft Word RTF, Version 4
- NBI
- Textfilt
- WordPerfect

Standard graphics interfaces

- MacDraw
- Platform-specific raster graphics filters
- MacPaint
- Encapsulated PostScript (eps)

Optional graphics interfaces

- TIFF

CAD interfaces

- Plotter-ready files in Calcomp 925/960, will size and rotate image, and add text and additional graphics

Optional CAD interfaces

- HPGL format, can be sized, edited, and annotated; individual vectors altered
- IGES input filter

Data communication for data-driven charts

- Lotus 1-2-3
- Multiplan
- VisiCalc

The same filters, but with a different mix of standards and options, are available on the other Interleaf products.

Interleaf Product Line

TPS. Technical Publishing Software (TPS) is the top of the line for Interleaf. The current version is 4.0.

TPS is sold modularly. The core product sells for \$2,500 on the Unix workstations. Optional packages include document management (catalogues, books, and a-pages—see “Compound Documents” below) and advanced graphics (tables, equations, gray-scaling, and image-editing). With these options, the software costs \$12,500.

INTERLEAF PUBLISHER. Publisher is a 386 PC and Mac version of TPS based on Version 3.0 of of the Unix product (Version 3 includes core product, book, and line art editor). Files are upwardly compatible with the latest version of TPS. (Documents created in 4.0 cannot be read by 3.0.) Files are completely compatible between Publisher and TPS 3.0. The product sells for \$2,495.

TEXT EDITOR. For those PC (XT or AT) users who don't need the full publishing functionality of TPS, Interleaf sells a text editor that includes all the basic word processing functionality without the graphics and advanced formatting capabilities. All files are completely compatible with TPS 4.0. The editor sells for \$695.

VIEWSTATION. The Interleaf Viewstation software, which runs on any DOS PC or Interleaf-supported workstations, opens a read-only window into an Interleaf document. While of limited use in traditional document-creation situations (suitable only for review and annotation), the Viewstation functionality supports the concept of the “living document” (see Vol. 3, No. 8, and the *Office Computing Report*, Vol. 10, No. 11), a file that is intended for on-line distribution, not for printout. For example, the Viewstation would be active in the service department of a large organization, where an on-line repair manual can be accessed at any time. Updates to the manual are done through a different department on an as-needed basis. The service personnel, therefore, are always guaranteed the latest and most accurate information.

Within the Viewstation, text can be copied, searched for, zoomed, and annotated. The first time a user makes an annotation, he or she specifies an identifier (usually the user's name). Any subsequent notes during the same session automatically indicate the name of the user. TPS users can read notes, but in

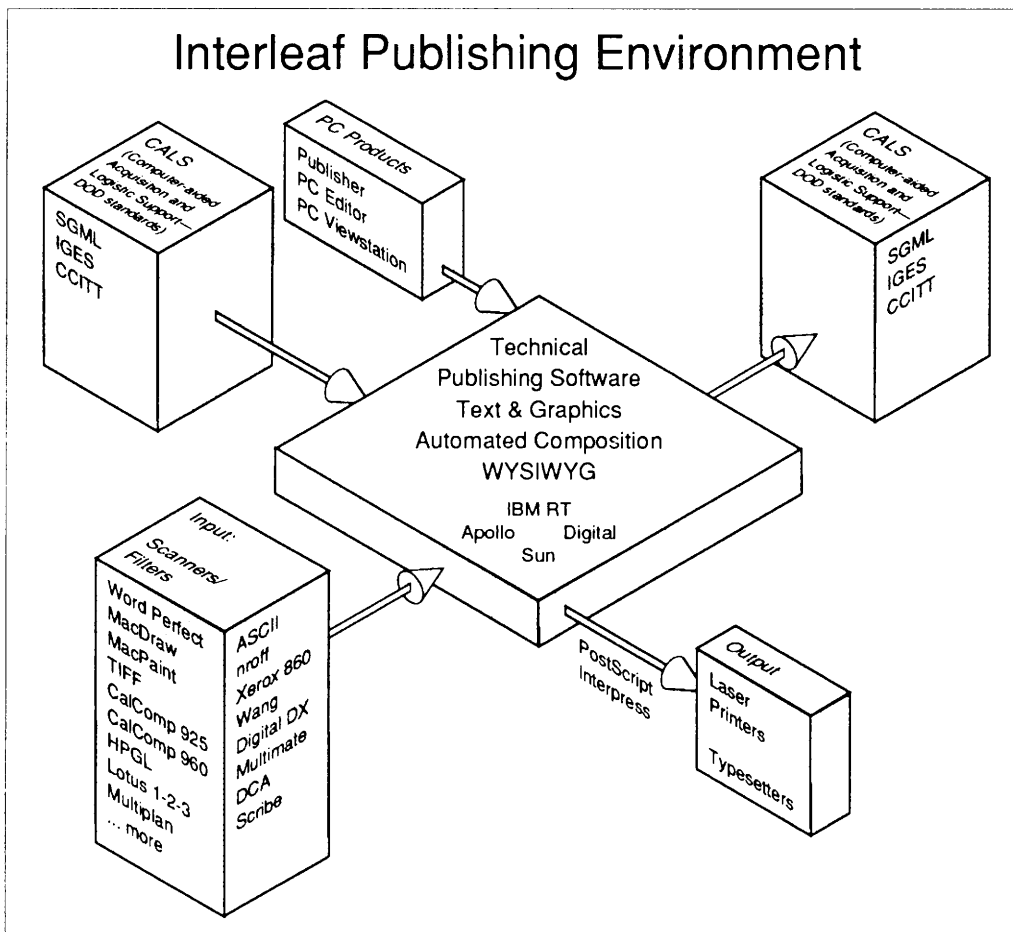
an awkward way. Notes are kept as separate items that include instructions as to where they should be inserted in the document (i.e., what text they apply to). We sincerely hope that Interleaf will make this process automatic.

The Viewstation also offers hypertext capabilities. Actually, the document hypertext links are created within TPS, but they can only be accessed in the Viewstation. While it is a terrific feature for the Viewstation, it seems illogical that TPS users cannot access these links.

Viewstation sells for \$795.

Interleaf's Interface

Interleaf provides its own proprietary, icon-based, windowing environment, which is identical on all platforms, running in the native win-



...dowing environments for those platforms. The Interleaf environment is not currently based on any industry standards, but Interleaf has recognized the strategic importance of the X standard and has stated that the Unix-based products will eventually run under X-Window.

In the Sun environment, Interleaf can run under SunView, thus taking advantage of the Suntools. While mouse usage under Interleaf varies slightly from the SunView implementation, you can easily configure the mouse to use the same model (either Sun or Interleaf) in both environments. A single command executes this very necessary configuration change.

The Interleaf products are very functional. But this can be a problem when you are initially learning the interface. Every section of the screen means something different. Different menus and rules apply, depending on where the mouse is pointing. The mouse pointer displays almost a dozen different shapes, depending on the screen position. While confusing at first, the dynamic functionality of the mouse, once mastered, helps you navigate through the many layers of functionality. Interleaf uses terminology that can be very intimidating for the novice user. By the time we had finished learning about components, inline components, component bars, component caret, text caret, masters, and tokens (to name just a few), we were thoroughly bewildered. After working with the system for a while, we were in better shape.

MENUS. Interleaf provides both pop-up and pull-down menus. There is both good and bad news about Interleaf's menus. Let's start with the bad: There are so many of them! And many menus have multiple submenus (at one point, we counted five!).

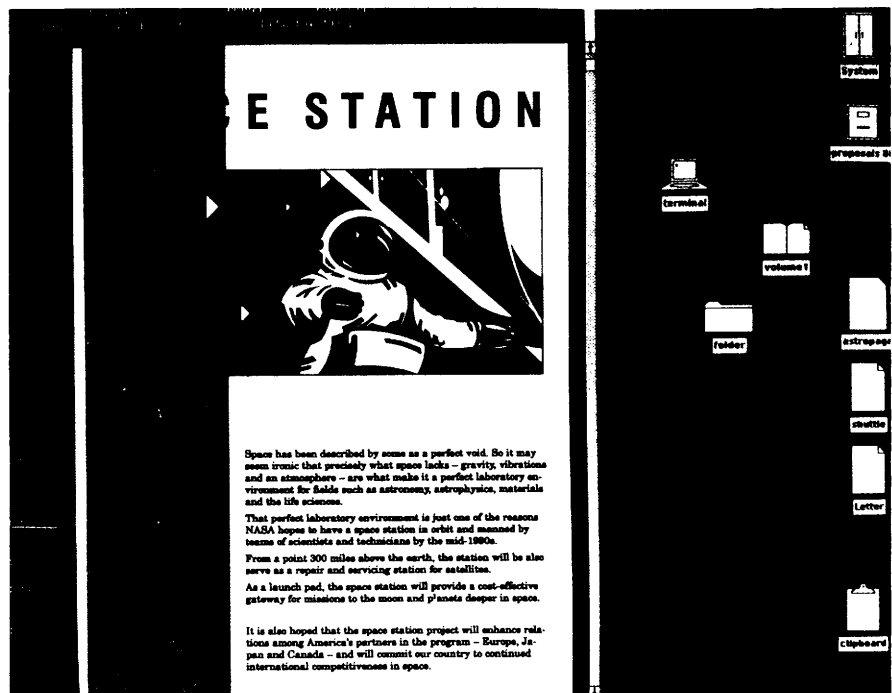
But now we'll look at the good news. Pop-up menus always display the appropriate functions for the current operation. For example, a different menu will pop up depending on where the cursor (selection point) is located and whether anything is selected. Pull-down menus are accessed from headers specific to the active document (including the file name, active font, type size, etc.), rather than the Macintosh/Windows model, where the menu line is part of the applications window rather than the document window.

Dynamic Defaulting. But the best news about menus is the default structure. Interleaf recognizes the overwhelming nature of multiple submenus, so the com-

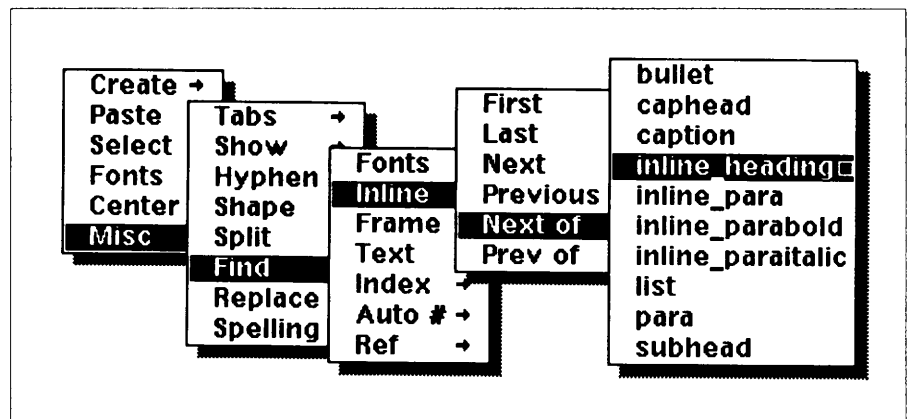
pany included intelligent defaults in every menu choice. The most common action will execute by the user's double-click on the item in the primary menu.

But wait, there's more. Often, after you have executed an operation once, a different option becomes the logical default. For example, when you execute a search, the initial default should be to search in a forward direction. The default should then change to find the next occurrence. Interleaf has built this flexibility, which it calls dynamic defaulting, into the system. It takes a while to learn (and to trust) the dynamic defaults, but, once you do, they are an invaluable timesaving device.

Security. Read and write permissions for owner, group, and all may be assigned by the user at the document level. But a file is not locked during editing. If you call up a document to edit that



The Interleaf desktop with an open document.



Interleaf's walking menus can have many steps.

is already opened, you will receive a warning that the file is in use, but the document is still editable. And the last person to store, wins. While there are sufficient warnings, and, in most cases, users will not be clobbering each other's work, just one mistake or mischief-bound coworker can destroy a lot of work.

The object-oriented nature of the product (see below) seems to lend itself to some sort of component- or section-locking capability, where multiple authors could be working on different sections of the same document. We'd like to see Interleaf emulate the database analogy of record-locking: Only the record currently being edited is locked. The rest of the file is available to others.

Interleaf does offer some facilities to support simultaneous editing of a document. Documents can be broken down into separate documents within the book environment (see "Books" below), so each file can be manipulated separately and then viewed and printed as one document. The limitation to this is that each document starts a new page, so, if sections appear on the same page, they must be within the same file. The a-pages feature (see "A-Pages" below) can divide and rejoin documents even within a paragraph. This solves the problem of new pages for each section. However, neither of these solutions solves the problem of the intentionally malicious user who sets out to clobber someone's work. As corporate culture turns more and more towards workgroups and collaborative projects, this level of security will become increasingly vital.

Links. Within the Interleaf software, you can create links between documents in two ways:

- **Shared contents.** A component or frame within a document is protected from change unless you change the master (see "Masters" below). This is useful, for example, in bullet lists. You can create the format you would like the bullet to take—amount of tab, size and shape of bullet, any associated text (i.e., "item"), etc. Each instance of the bullet component would repeat the specified design but would not be editable. Shared contents are even more valuable within catalogues (see "Catalogues" below), where they can be shared among a group of documents.
- **Copy links.** Instead of making multiple copies to a document, a copy link creates a pointer to the original.

Links to outside applications will be discussed below in the section on compound documents.

TPS Functionality

We will discuss all functionality issues as they apply to TPS, since that is the top-of-the-line product and all others are merely subsets. We will not discuss all the functionality of TPS, because, as we stated, the system has an amazing amount of detail and complexity. (In this case, complexity is a positive attribute. Interleaf has stuffed loads of functionality into the

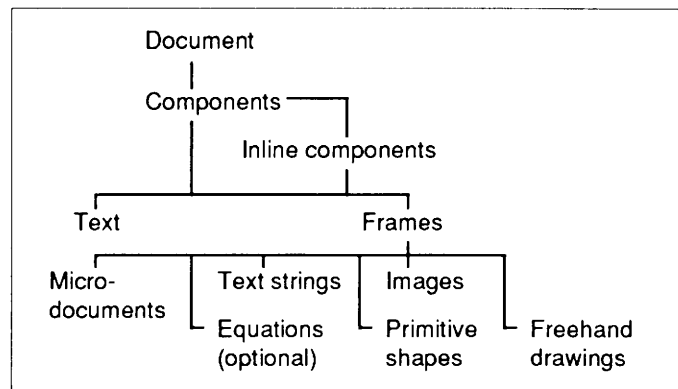
product.) Rather, we will look at the basic structure of an Interleaf document, graphic capabilities, and support for multiple authors on large documents.

TPS OBJECTS. All TPS documents are broken down into components: inline components, tables, and frames. Each can be considered an object with individual properties and characteristics. And, as objects, they can be manipulated by moving (deleting, enhancing, etc.) the name of the object, which is displayed in an area to the left of the document.

This approach offers a lot of flexibility within a structured framework. Admittedly, it takes a while to be comfortable with the idea that every keystroke in your document is part of a structured component. But the freedom this structure gives you is worth the initial discomfort. Documents can be manipulated more easily, updated more easily, duplicated and imitated more easily. (For more information on structured documents, see the discussion of Manuscript in the *Office Computing Report*, Vol. 11, No. 10.)

Masters. All Interleaf objects, components, inline components, frames, and tables are based on a master paradigm. When you create a component (for example), you are actually creating a master version of the component, including the formatting and boilerplate material. Each time you use the component, you are actually creating an "instance" or copy of the master. The process is transparent to the user.

Components. Every text paragraph (or single-line heading, etc.) is actually a separate component, even if the component is entitled "standard paragraph" and has only very basic formatting properties (font, size, line-spacing, etc.) assigned to it. Components are based on master components, which are document specific. If you want to use a specific master in another document, you can either cut and paste it in or store it in a catalogue (see "Catalogues" below). While it would seem ideal to have all components available in all documents, much like Macintosh programs do with style sheets, the potential (unlimited) number of different components would make this extremely hard to manage.



Structure of an Interleaf document.

Inline Components. Inline components are subcomponents affecting a number of characters or graphic objects within a component (for example, a comment that can be hidden or displayed depending on the situation). Like a component, an inline component is created from a named master.

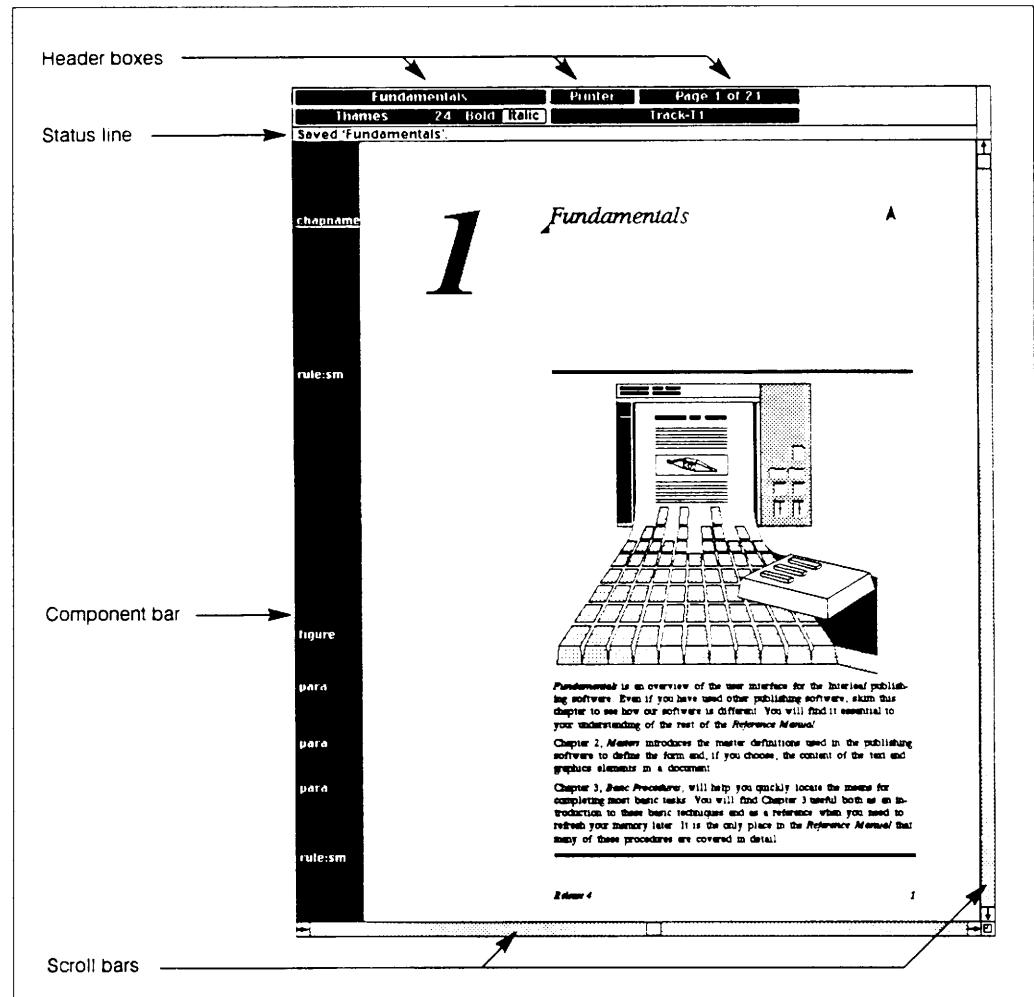
Inline components can be completely independent or can inherit the characteristics of the major component. The inherit option can also be relative. For example, an inline component can inherit the point size of the major component plus two points (i.e., the component is changed from 10 to 12 point, and the inline component changes from 12 to 14 point).

Frames. Illustrations and graphics are created or positioned in frames. Like components, text-anchored frames (a frame is usually anchored to a specific item in the text rather than to a specific position on the page) are created from masters. Six predefined frame masters, indicating frame position relative to the text, are included with TPS.

Pasting a graphic object into a document will automatically create a frame around it.

To insert text into a graphic frame, you can create an editable microdocument. When working on text, the editor offers the same functionality as the standard TPS editor, including creating and manipulating components. The editor is also used to edit the graphic objects, equations, charts, or images, providing the appropriate functionality based on the type of object being worked on.

Tables. To create a table, you specify the number of rows and columns, and the system automatically creates the table cells. Tables can span pages and include running headers and footers. One nifty function is the ability to convert a table designed in text with tabs into a cellular table. Since cells are basically microdocuments, they, like text, are formatted by components. Cells can also contain graphics. In addition to formatting cells, you can also assign properties to rows and columns.



Parts of an Interleaf document.

Templates. Using templates is a way around copying masters from document to document. Templates are documents containing a set of masters—components, frames, tables, autonumber streams (see “Autonumber and Autoreference Tokens and Streams” below), page properties, and boilerplate text and graphics—that are stored in a special directory (all documents within this directory are accessible as templates through a pop-up menu) and upon which other documents can be based. TPS comes with a set of predefined templates. Users can also define their own using a style-by-example approach. Basically, this involves cutting all the components, frames, tables, and autonumber-stream tokens in a document whose design you particularly like, and pasting them into a blank document. Another method is to copy the document and delete all nonboilerplate contents. You are left with a document containing all the masters and standard text and graphics.

COMPOUND DOCUMENTS. Though there is no documented function to create live links with outside applications, users can establish links through the Lisp Interpreter (see “The Lisp

Interpreter" below) to outside applications. The function can take a specified external file, run it through the appropriate filter, and place it in an Interleaf document.

The process is not transparent, though. At this time, the Lisp link is a desktop function that must be specifically invoked through a pop-up menu.

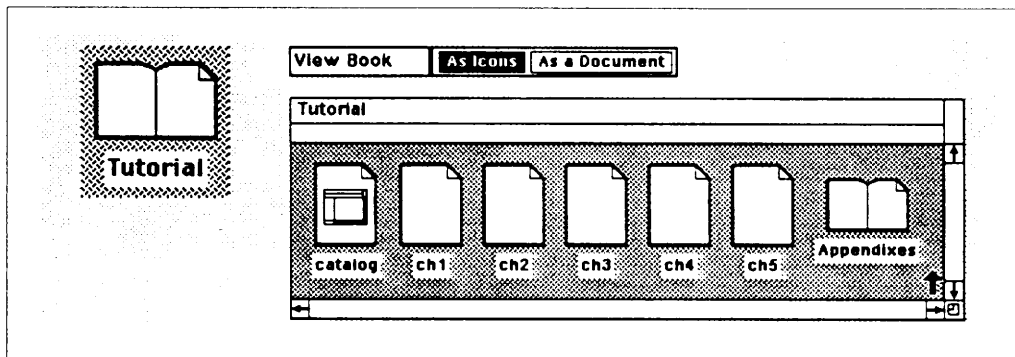
In the future, Interleaf plans to integrate this capability more transparently into the document editing process. No specific time frame was given.

SUPPORT FOR LARGE DOCUMENTS. Interleaf has done an excellent job of supporting large (book length) documents and co-authoring projects. These features are available through the optional Document Management package.

Books. In TPS, a "book" is a special directory that creates a relationship among the documents it contains.

Books offer the following features:

- Consecutive page, series, and reference numbering across documents.
- Concurrent editing. Changes to a book document (usually numbering or referencing changes) immediately display in all other documents in the same book, even if they are being edited.
- The ability to scroll from one document to another.
- The ability to close all documents in a book with a single command.
- The ability to rotate individual pages to accommodate landscape illustrations.
- Viewing or printing the book as if it were a single document.
- Index and table of contents generation for one or more documents.
- Effectivity control (see below).



Iconic view of a book.

- Consistent use of components, frames, and autonumber streams through the use of a catalogue (see "Catalogues" below).

Order of Icons. Within the book icon are folder and document icons that can indicate which pages of the book (if it were printed as one document) each icon contains. The documents are numbered sequentially from left to right. If an icon is moved, the numbering sequences within the book documents automatically update. This is a very interesting feature. If you decide that chapter 3 should actually be chapter 6, all you have to do is move the icon to follow chapter 5. All book-related functions (numbering, table of contents, etc.) will dynamically recalculate.

Catalogues. Basically, a "catalogue" is a book template. In other words, it is a special type of document used to control the master components, frames, and autonumber streams in all the documents in a book. A book may contain more than one catalogue, and multiple catalogues are organized in a binder directory. When a catalogue is modified, it automatically updates the properties of affected documents in the book. If a book document is open, the changes are immediately updated on screen.

Catalogues may also be used to contain boilerplate text for document assembly and to create a library of graphics in a catalogue.

Effectivity Control. Effectivity control enables you to control which elements of a document are visible or effective. This can be very useful, especially in long documents that change continually. For example, the documentation for the Interleaf software is similar in most respects for all product platforms, but there are some differences. Rather than create a separate document for each platform, you write one manual, tag the custom information to indicate under what conditions it should appear, and generate a number of different manuals from one document. The tagged information can range from a single character or word (for example, Sun, Mac II, PS/2, etc.) to entire sections (running under OS/2 versus a chapter on Unix). Entire documents within books can be conditional. And components can be

selected using basic Boolean operators. For example, "choose all components not equal to version5."

Many programs allow you to hide or reveal text, but this control feature does much more. For one thing, you hide or reveal content by component, inline component, table row, or graphic frame name, or by trackable categories (variable fields within a component defini-

tion which can be effected using Boolean logic). This gives you much more flexibility than simply having one or two categories of notes. And any numbering sequence, including page numbers, will readjust to accommodate only the effective sections of the text. This is a very powerful capability. It means, for example, that product specifications can be continually updated and that selected versions can always be printed with correct page and illustration numbering. Actually, the potential is endless. Consider using effectivity control for reports generated from a database. Many different reports can be written from the same query by effecting and hiding selected components.

A-Pages. Often, large documents need to be updated. But, instead of printing and shipping out an entirely new copy of the document, it is more efficient to produce a-pages. These are most commonly used in looseleaf-notebook pages. Let's say, for example, that you need to revise page 247 of a 500-page manual by adding additional text. The a-page feature allows you to create a new page 247 along with a page 247a, 247b, and so on, as needed. The rest of the document does not need to be reworked or reprinted.

A-pages can also be used to split a document into smaller documents for concurrent editing. The separate sections can be viewed as a single document within the book feature.

MORE FUNCTIONALITY. In addition to the document structure and support for large documents, TPS offers some interesting and powerful features.

Diagramming. "Diagramming" is Interleaf's term for graphics creation and editing. TPS offers impressive graphics tools, including standard object-oriented graphics (drawing boxes, curves, arcs, splines, etc.), freehand drawing (using the mouse like a pen), a clip art catalogue, an equation editor (no calculations, just typing in the mathematical symbols), chart creation, and image-editing.

Graphics are not created from a palette. You must select the word command from a submenu to create an oval, arc, line, or box. In such a graphics system, with such a visual interface, we were surprised at this choice. The standard in most graphics packages is to use a palette. Most users are accustomed to seeing the shape they want, not figuring out what word gets them the desired effect. However, users can always create a document to serve as a palette. Palettes are used to determine color fill (see below) and pattern fill.

All the standard graphics operations—size, move, rotate, duplicate, bring to front or back, and group or ungroup—are available, as are some nifty little features like gravity, which causes objects to be attracted to one another when they are close (so you don't have to worry about matching up two objects exactly), and stickiness, which glues intersections together even when the object is moved, sized, rotated, or aligned. Unfortunately, Interleaf does not yet support color on screen, though you can specify color by specifying the percentage of black in the gray tone or using the HSB (hue, saturation,

and brightness) model of color description.

As we mentioned earlier, text within diagrams is created in a microdocument. Single-line text strings can also be created and attached to a point in a frame. These strings are edited differently than microdocuments.

Graphics from external programs, including CAD and plotter files, gray-scale images, and line art, can be included in an Interleaf diagram by importing them through the graphics filters. Encapsulated PostScript (eps) files can be directly inserted by changing the file extension to .eps.

Scanned images can be cut and pasted into a document. The image can be modified with the diagramming tools.

Charts are created in frames and edited within the diagramming system via property or data sheets rather than through the graphics commands. Spreadsheet data can be imported into a data sheet to create a chart.

Revision Control. At present, Interleaf does not offer extensive revision control. Outside of revision bars—black stripes that indicate new material—there are no specific features for redlining or co-editing. However, using inline components for notes, effectivity control, and overlay frames (basically, post-it notes), you can create your own redlining procedures. Not terrific, but definitely usable.

Interleaf is planning to automate the revision procedure, including version management and tracking of editing changes. Though no date has been specified, our hunch is that this capability will appear in about a year.

Autonumber and Autoreference Tokens and Streams. Within a single document, many different lists (tables, footnotes, etc.) can be numbered, each represented by a different stream of numbers. Each stream can have a different numbering sequence. A token is inserted for each numbered item. Tokens look like any number, but they will dynamically change as the list is renumbered.

Autoreferences are tokens whose value is linked either to an autonumber token or to a page number. A tag (yet another marking device) is used to match up the autonumber token with the autoreference token.

WHAT IT DOESN'T DO. TPS is not a page layout product. Unlike FrameMaker, PageMaker, and Ventura Publisher, the Interleaf product is document oriented. In many publishing scenarios, this is desirable, but there are certain applications where page orientation is vital. For example, our own newsletter format requires complex, revisable page layout, where text must flow discontinuously from a column on one page to another several pages away. While Interleaf can create these pages easily enough, it is not simple to revise them.

Also, TPS does not officially provide any keystroke macros. There is a prototype keystroke capture facility which is available to customers, though it is undocumented and unsupported. There is also a command-language capability with the new Lisp Interpreter (see below), but it is definitely not an end-

user tool. In order to create macros (and more complex programs), you must feel comfortable working in Lisp. Not many end users have that ability.

A major deficiency is the lack of live links between external applications—true compound documents. The next generation of Lisp accessibility will help considerably in that direction.

The Lisp Interpreter

An Interleaf-developed Lisp Interpreter, a subset of Common Lisp, is incorporated into TPS 4.0. It is not intended as an end-user tool. Rather, its audience is Interleaf's and customers' internal engineering departments and third-party application developers.

WHY LISP? We aren't too keen on the choice of Lisp as the means to integrate outside applications with a product. The majority of applications based on Lisp, and, thus, the majority of Lisp programmers, are in the artificial intelligence arena, another niche market like publishing. Unix office applications are simply not being developed in Lisp. They are being developed in C, Objective C, and C++.

According to Stephen Pelletier, chief technical officer and one of the architects of TPS, Interleaf saw no need to reinvent the wheel by developing its own proprietary programming language. Lisp was chosen because it is an interpretive (as opposed to compiled) language that understands objects. A Lisp function doesn't need to be told what type of object is being affected (text, graphic, etc.). It can figure it out on its own.

And, Pelletier was quick to explain, outside applications do not need to be written in Lisp to integrate into the Interleaf environment. The integration can be done through a Lisp layer. This layer can be "thin" to provide, for example, external links to applications running on the same platform or network. A more complex layer can be written to integrate applications into the Interleaf environment, incorporating the Interleaf interface with the application's engine.

And the opposite is also true. For example, in a CASE environment, the Lisp layer can drive the Interleaf engine in a batch control mode.

Applications that are written in Interleaf's version of Lisp do have an advantage. Because all of Interleaf's software is ported to each of the supported platforms (Unix workstations, 386 PCs, and Macs), applications written in Lisp are automatically ported to all Interleaf environments. The third-party developer does not need to do the port.

In the future, Interleaf plans to provide Lisp tools to facilitate application programming for the Interleaf environment—a sort of toolkit. This is important to the Interleaf strategy. According to Larry Bohn, one of Interleaf's directions is to become an integrating environment, working with VARs and OEMs to develop more commercial applications, including office functionality, that run with and under Interleaf.

Futures

Besides the enhanced revision control, improved external links, and Lisp toolkit, Interleaf is also planning to offer a full macro language, including both keystroke capture and editable scripting. Security is being addressed, as is a customizable keyboard facility.

Interleaf in Summary

Interleaf has done a good job of providing products that span all the major platforms used in the industry and that offer a variety of functional levels suitable for different categories of users. The software, however, is expensive—well worth the price if you have a definite need for the extensive functionality, but not cost-justifiable for many desktop publishing applications.

The TPS software is impressive. There is so much there. But it is difficult to learn and, therefore, can be rather intimidating. The documentation is not much help in clarifying concepts. It tells you how to do certain operations, but not necessarily why you would want to do a particular operation.

Interleaf obviously knows the publishing industry very well and has targeted its product to this vertical application. But is there a place for Interleaf in a more horizontal office environment?

GETTING OUT OF ITS NICHE. Interleaf has designed a nice graphics environment for its publishing products. Though there are certain drawbacks to the interface—number of submenus, etc.—most of these problems are specific to the size and complexity of the Interleaf application, not the environment. With the Lisp programming language, Interleaf is opening up its world to outside application developers.

Developing an integrated office system based on Interleaf could be very interesting. The product has a great deal of functionality not specific to publishing (word processing, image-editing, table generation, etc.), and the object-oriented, structured nature of the software would be an excellent basis for many applications (project management and database front ends come to mind). But in order to achieve this, Interleaf must make its product extensible, and the current implementation of Lisp is not sufficient.

Also, in order for an integrating environment to succeed, it must make users instantly comfortable. Interleaf should rethink some of its terminology (i.e., why not use "subcomponent" instead of "inline component"?) to make it more approachable for users who migrate from other environments, especially the Macintosh world.

Once Interleaf provides a toolkit including programming APIs to software developers, the Interleaf desktop, with its built-in structure and object orientation, could be an attractive Unix office environment that could rival many systems already out there. ●

Apollo and Sun Service and Support

By Alexia Martin

MAYBE A SOFTWARE ENGINEER at Carnegie-Mellon is willing to hold on the telephone for minutes on end to get technical support from Apollo or Sun, and maybe a design engineer at Intel is willing to box up his workstation for return-to-factory service. But a broker or financial manager would balk and go back to his green visor and quill pen.

These nontechnical end users and their colleagues want on-site service to install, configure, or maintain networks of complex workstations. They want help setting up system administration procedures for their first installation. They want immediate access to technical support personnel knowledgeable about their installation and their business applications and able to speak non-nerdese. They want courses to develop system administrators and end users. Their programmers need courses on employing new workstation tools in their application development efforts. Their technical support personnel eventually clamor for self-maintenance options as they become comfortable and expert with high-end workstations. And their purchasing departments demand volume maintenance plans as the numbers of workstations grow.

Responding to customer demands and competitive pressures, Apollo and Sun are offering service, technical support, and training programs showing a maturity of customer service marketing and program implementation well beyond their years. The matrix at the end of this article shows the resources and breadth of offerings of these two workstation manufacturers. We use a similar matrix to describe the service, support,

and educational offerings of Data General, Digital, Hewlett-Packard, IBM, and Wang in the *Office Computing Report* (see Vol. 2, No. 6). Sun and Apollo offer programs similar to those of these larger vendors. Both, however, are weak on people power for delivering some services, such as installation and technical support. And so far, neither has remote diagnostic or predictive support capabilities as sophisticated as those of these other five vendors.

Integrated Packages, Start-up Packages, and Network Services

Where Sun and Apollo have been able to surpass the big vendors is in the provision of integrated maintenance and support. Without the traditional infrastructures of separate field and software engineering organizations, they are finding it easier to offer integrated packages of hardware maintenance and technical support. With a single contract, customers can cover hardware and software service needs. Sun offers the most complete integrated program under SunClass Services.

SunClass Basic Service includes software update materials, same-day, on-site server service and next-day standalone workstation service, and AnswerLine software support. Customers requiring more comprehensive service and support can buy SunClass System Service, which includes all of the above plus initial system and software installation assistance and customer education certificates. Extended service (24 hours, 7 days a week) and priority response (2 hours) are available as options.

Apollo offers equal service and support. Its Standard Sup-

	Apollo	Sun
Strengths	<p>Comprehensive service and support portfolio</p> <p>Implementation services good for new users</p> <p>Consistent service features and pricing for international customers</p>	<p>Comprehensive service and support portfolio</p> <p>Integrated service, support, and training packages</p> <p>Management attention on improving customer service and customer satisfaction</p>
Weaknesses	<p>Multivendor maintenance arrangements</p> <p>Limited vertical industry orientation</p> <p>No predictive support</p> <p>Customers must purchase separate hardware and software contracts</p>	<p>Multivendor maintenance arrangements</p> <p>No predictive support</p> <p>Inconsistent service features and pricing for international customers</p> <p>People resources spread too thin</p>

port Agreement includes on-site service when needed, priority response, installation of engineering improvements as appropriate, and operating system support through Apollo's Software Response Center. But a Software Subscription Service must be purchased separately to obtain operating system updates, documentation, and the right to copy software onto multiple nodes.

As both vendors begin to sell into the nontechnical commercial market, they have found that customers require "start-up services" to get them up and productive in a timely fashion. Apollo's Quick Start program is designed to help users quickly integrate Domain into their own working environments. The services range from system orientation to implementation to application-planning, and include specific services such as setting up registries, defining system security, tailoring start-up scripts, using naming services, and distributing application files across a network. Customers purchase Startup Services at a single, fixed price and then choose the combination of services they need from a program menu.

Sun provides three levels of start-up services. SunStart One hardware and software installation and configuration service is a fixed-fee package priced depending on the system. SunStart Two, which builds on SunStart One with implementation assistance, and SunStart Three, which builds on SunStart Two with the addition of network design and installation management, are offered as fixed-fee packages with a custom quote component. Custom quotes are based on network service rates and discounts.

Apollo makes it easy for customers to understand what they are getting and what they can expect with the following network service packages:

- **Network Planning and Design Services.** On a project basis, Apollo will perform a requirements analysis, site survey, and network design and plan, and will provide documentation.
- **Network Installation.** Apollo offers network installation through Apollo-approved vendors. Verification and certification are included.
- **Verification and Certification.** Apollo provides these free with any installation service for Domain, Ethernet IEEE 802.3, or IEEE 802.5 networks.
- **Network Management Services.** On-going support is provided such as: diagnostics, network administrative procedures, network security, network topology enhancement, backup procedures, specialized data communications consulting, and problem isolation, escalation, and resolution.
- **Network Maintenance Management.** Apollo will offer on-site network support in the future, which will cover physical layer components of the network.

Training

Both Apollo and Sun have expanded their training offerings substantively in the past year. Their Unix courses have achieved a level of renown that entices Digital, IBM, and HP customers to attend.

Apollo offers three types of training: regularly scheduled, on-site, and video-based. Some 31 courses are available in eight categories: operations, networking, programming, graph-

ics, communications, data management, systems maintenance, and applications software.

The growth and success of Sun made it inevitable that its Educational Services would have to expand its course offerings. Its current curriculum is organized into five major areas of concentration: end-user curriculum for nontechnical students to develop an understanding of Sun systems as office tools; system administration to prepare students to install and maintain a local area network of Sun workstations; programmer curriculum for technical users to learn to employ programming tools in their applications; installation and networking for students to learn and practice SunOS installation procedures and create a complete network environment of Sun products; and maintenance for students to learn the fundamentals of Sun workstation hardware architectures along with their respective diagnostic techniques and troubleshooting methods.

Strengths and Weaknesses

With the increased reliability of both vendors' systems, neither Sun nor Apollo should need hordes of field engineers to provide maintenance services. But Sun's success is breeding discontent among its growing user population. It is sometimes hard pressed to provide timely maintenance and technical support. Sun temporarily needs more engineers capable of installing hardware and software and setting up networks. If Apollo sales pick up again, it, too, may suffer comparably, because its people resources appear even slimmer than Sun's. Relationships with third parties may be appropriate to provide these services, but both Sun and Apollo will need to carefully manage the installation and configuration assistance these third-party vendors provide to ensure a level of quality commensurate with their customers' expectations.

Customers have complained about knowing more than the technical support staff of both vendors. "We feel we have to teach them," says a technical support engineer at GTE. "They don't understand how we use our systems," says another at Intel. While both companies have increased their technical support ranks substantially in the past year, they still need more staff and more sophisticated customer support facilities. Beginning to address these criticisms, both have installed call management systems to reduce customer wait times. Both are building or enhancing their customer information systems so that callers need not describe their system environment each time they call. Apollo is beginning to provide two-tier support:

The front line addresses the most common requests and problems, and the back line, consisting of high-level R&D types, addresses the most complex. Customers who buy Sun's Personal AnswerLine Service are assigned a support engineer who becomes knowledgeable about their accounts. Sun is also beta testing a "Technology Center" that will incorporate remote system and network diagnostic services. It will use artificial intelligence applications with several symptom-cure databases for problem resolution.

While customers recognize the need for training on new systems, they are often unwilling to pay the price of sending employees to off-site training. Both Sun and Apollo offer on-site courses, but both need to offer low-

cost training options. Sun needs to come out with Easy Unix on its larger systems and desktop servers, instead of just on the 386i, as soon as possible.

The matrix on the preceding page summarizes our assessment of service and support offerings of Apollo and Sun. Both have complete service and support portfolios of start-up services, software support, hardware maintenance, network services, and training to satisfy just about any customer. Apollo's implementation services appear slightly more comprehensive than those of Sun's and more attuned to nontechnical users. On the other hand, Sun's integrated packages and its marketing of them make it easy for customers to understand their value, making it easy for Sun to sell them. Sun's attention to service and support appears slightly stronger than that at Apollo.

Both companies show similar weaknesses: insufficient attention to multivendor maintenance arrangements—should they manage such arrangements, cooperate with a major manufacturer, or partner with a major third-party vendor? Both Apollo and Sun have sufficiently robust portfolios to configure service and support programs for any company in any industry. However, neither they (nor any other vendor for that matter) seem to be attuned to the *personal* needs for service and support of the individuals in various vertical industry sectors, (i.e., a bank secretary needs different support than does a shop floor operator). Both lack predictive support programs. While Apollo has consistent service program features and pricing for international customers, Sun does not. Finally, Sun's people resources are spread too thin and, while it has the basic structure to provide service and support, it must continue to strive to maintain customer satisfaction as its customer base grows. ●

Alexia Martin is a writer and consultant for the Institute for the Future, Menlo Park, California.

While both companies have increased their technical support ranks substantially in the past year, they still need more staff and more sophisticated customer support facilities.

	Apollo Computer, Inc.	Sun Microsystems
	(All figures worldwide)	(All figures worldwide)
RESOURCES:		
Sales locations	70	110
Direct sales staff	Not available	523
Marketing support representatives	Not available	267
Telephone response centers	1 U.S., 22 WW	1 software, 2 hardware (U.S.), Software Response Centers in each country outside U.S.
Software support representatives	100	175
Training centers	4 U.S., 22 WW	3 U.S., 9 international
Trainers	60	65
Field service locations	70	125
Field service engineers	Not available	325
Workstation repair centers	3	2 U.S., 9 international
PRE-SALES SUPPORT: (feasibility studies, cost justification, application analysis, etc.)	Part of sales process or available on time and materials basis	Part of sales process or available on time and materials basis
IMPLEMENTATION SERVICES:		
No charge support	90-day warranty; site planning for power requirements	Some pre-sales only
Packages	Apollo Quick Start; Network Services	SunStart Services
Porting and conversion services	Consulting service	Time and materials basis
SOFTWARE SERVICES:		
Contracts for:		
On-site support	Account Software Management (ASM)	Software Update Installation Service, SunAdvisor
Telephone support	Response Center Support (RCS) 8:30-5:00 local time	Personal AnswerLine Service immediate response; AnswerLine Service provides 2 hr. response. Both 7:00-5:00 PST
Software/doc. subscriptions	Software Subscription Service (SSS)	Software Update Service
Remote "repairs" and patches	Under RCS coverage	Under AnswerLine Services
Problem/solution database	Online Bulletin Board under ASM or RCS	Online Bugs Database under any software support service
Other software services:	System development and office applications available as consulting services	System development and office application tailoring available as T&M

	Apollo Computer, Inc.	Sun Microsystems
HARDWARE SERVICES:		(U.S.)*
Warranty period	90-day on-site, plus RCS for operating system	90 days (U.S.)**
On-site services:		
Basic contract	Standard Support Agreement (SSA)	SunClass Basic Service (includes Answer-Line software support)
Hours covered	8:30-5:00 local time	8:00-5:00 local time
Response time	24 hours	Ranges from next day to 4 hours/same day
Premium contract	Apollo Maximum Service (AMAX)	SunClass System Service
Hours covered	8:30-5:00 local time, or 8:30-midnight, or 24 hours	Up to 24 hours/day
Response time	4 hours	Up to 2 hours/same day
Remote diagnostics	Through dial-in modem	In beta testing stage
Predictive support	None	None
Self-maintenance/cooperative support	Shared Maintenance Program	Cooperative Maintenance Service, SunPartners (OEM)
Maintenance, other vendors' systems	Selected peripherals	Considered on individual basis
Other:	Return to Apollo, Board Return-to-Apollo, Site and Volume plans	On-site Service Center, Customer-assisted Return Service, System Return Service, SunSpares
NETWORK SERVICES:	Complete range of network planning and design, installation, verification and certification, network management, and on-site network support (future program)	Range of network services from design to installation management
TRAINING:	Operations, networking, programming, graphics, communications, data management, systems maintenance, applications software	End-user and system administrator, programmer, installation and networking, hardware maintenance
HOURLY RATES:		
System engineer hourly rates	\$100 to \$150	\$110 telephone support, \$160 for consultants (discounts for multiple days)
Field engineer hourly rates	\$100 to \$125	\$110
Average cost of one day's training	\$125 to \$320	\$200 to \$333
		* Varies outside U.S.
		** Varies up to 12 months outside U.S.

NEWS

PRODUCTS • TRENDS • ISSUES • ANALYSIS

ANALYSIS

• AFCAC 251 •

The Aftermath

AT&T could not have been happier: It won AFCAC 251, the largest government bid ever for computer systems. The telecommunications giant saw its victory as a way to legitimize its position as a computer vendor. We thought it might be interesting to look at how this bid, which could be worth as much as \$4 billion, turned out.

AT&T was not the only winner in this megabid. The company estimates that about 40 percent of the money will go to subcontractors such as TRW (networking) and CDC (storage technology). Database vendor Unify will have a piece, too.

One winner that has received the least publicity is Cambridge, Massachusetts-based VenturCom. VenturCom's little-known office package, Prelude, was, in fact, the office system that AT&T bid. The other four finalists for the contract all bid Uniplex, so it is safe to say that many in the industry believed that Uniplex would be the Air Force standard. Ironically, AT&T has chosen Uniplex as its strategic office platform. Why, then, didn't AT&T bid Uniplex (like everyone else)? Because AT&T decided that it wanted an exclusive with whichever company it bid.

Therefore, when Uniplex refused to give up its other commitments, AT&T turned to VenturCom. Most of VenturCom's expertise is in the operating system arena. In recent years, however, it has poured resources into its Prelude product with varying degrees of success. In essence, VenturCom built a custom product for AT&T that could meet all the technical requirements of AFCAC. All is not smooth, however, at VenturCom. There are rumors that the company is in financial trouble.

IN PERSPECTIVE. Of course, AFCAC's turbulent life is not about to settle down. Honeywell Incorporated, one of the five finalists, has protested the AT&T award, stating that its price was 19 percent lower than AT&T's. (There is something about the potential for more than \$4 billion worth of business that brings out the fight in these vendors. Remember Digital's lawsuit against the Air Force when it required System V as the operating system?)

Last January, we took an in-depth look at AFCAC 251 both from the perspective of what it asked for and of what the ramifications of the bid could be. In the most optimistic terms, the specifications could have created a sophisticated, highly integrated system. However, because the RFP was so vaguely worded, all vendors, without exception, took the low road, taking ex-

• I N S I D E •

AFCAC Results: And the Winner Is AT&T. **Page 16**

Sybase Provides an Object-Oriented Development Environment. **Page 17**

Progress Is on the Fast Track. **Page 17**

Digital Adopts a Promising PC Strategy. **Page 19**

Motorola Bets on Unix. **Page 19**

isting office offerings like Uniplex and adding the required software such as project management. Therefore, being able to cut and paste a part of a spreadsheet into a document was defined as integration by many vendors. No one offered object-oriented, icon-based software, which will become commonplace over the next two to three years. In addition, rather than selecting a distributed network computing technology, the vendors offered traditional minicomputer networking.

But, despite our disappointment in that area, AFCAC 251 was the most important single event in the history of Unix. For the first time, vendors and users began to believe that the movement to nonproprietary operating systems was more than lip service. Here was a large agency willing to pay billions to get Unix. If the Air Force wanted Unix, maybe others were serious, too. Wouldn't all those public and private organizations that did business with the Air Force also purchase Unix systems? Even if AFCAC never grows to its projected net worth of \$4.5 billion, the seed has been planted: Unix is where the money is.

Before AT&T won the AFCAC bid, it was an embattled company. Its empire seemed to be crumbling. Its traditional Unix licensees were in revolt, and its role as protector of Unix was fading badly. The award of

AFCAC 251 changed the way many in the industry perceived AT&T. The company lost no time making it clear to anyone willing to listen that AFCAC was the most important bid in the history of the computer industry, and AT&T had won it. The king was back on the throne, and all was well in the kingdom. At last, the keeper of Unix would start making some money. It was time for the wayward licensees to get back in line.

THE FUTURE. How will we look back at AFCAC 251 three years from now? Well, our guess is that it probably will not be the panacea that AT&T anticipates. The company will still have to work hard to distinguish itself from the other Unix vendors. While some vendors will be reassured of AT&T's leadership ability, OSF will continue to build its operating system and challenge AT&T's dominance.

We expect that the Air Force RFP will be looked upon as a symbolic point in the long history of open systems. It will be the point when Unix was first viewed as an alternative to a proprietary commercial operating system. ●

— J. Hurwitz

• SYBASE •

Tools for Applications Development

Sybase Incorporated has announced its first 4GL (fourth-generation language) development environment, the APT (application productivity tools) Workbench, for the Sybase RDBMS (relational database management system). APT Workbench is a new addition to the Sybase SQL Toolset, which comprises the client (or front-end) component of Sybase.

Without APT Workbench, developers must use a 3GL such as C or Cobol to develop Sybase applications. APT Workbench, on the other hand, will provide an object-oriented development environment, including a 4GL,

with a windows-based graphical interface. This will significantly improve developer productivity by reducing development time, and will allow prototypes to be extended into production applications. The objective is to let the developer generate a prototype application visually, enhance it with the 4GL, and then add 3GL procedures if necessary. APT Workbench will also allow less experienced programmers and end users to develop applications.

We will provide a detailed review of APT Workbench when we review the Sybase RDBMS in a future issue.

APT Workbench includes four components:

APT-BUILD. In APT-Build, the developer selects the tables to be included on a form, generates a default form (which contains all of the fields in the tables selected), and selects the appropriate application options to be associated with the form (such as add, modify, delete). APT-Build then generates the application. This includes setting up any relationships between the tables and generating a default menu. The form can then be customized with APT-Edit, and the application itself can be customized with APT-SQL.

APT-EDIT. APT-Edit is a visual forms editor for defining the layout and characteristics of form objects, including single fields, groups of fields, pop-up menus, and windows. (Forms are treated as windows containing a collection of objects.) This tool is used to customize and enhance a default form generated by APT-Build. Among other things, APT-Edit allows the developer to define the dimension, location, and point size of the form; field-level help; multilevel grouped fields (e.g., master-detail relationships); and navigation and scrolling options.

APT-SQL. With the Sybase 4GL, the developer controls objects (forms, menu items, fields, groups, etc.) and their properties. The developer can add processing to fields or other objects on a form, define totals, etc. While APT-SQL includes a full complement of

flow-control logic, it also provides advanced group processing. This feature can eliminate much of the looping and control programming required when processing computations against groups of records. Group processing is common in many applications dealing with master-detail relationships.

APT-SQL also provides access to Transact-SQL, the Sybase transaction processing and database query language that accesses the SQL Server, and to 3GL routines. Another important feature of the 4GL is its support of the client/server model. The language is designed to provide multiserver communications support, which is important in a distributed database environment.

APT-EXECUTE. APT-Execute is a standalone, run-time module designed to provide cost-effective deployment of applications developed using the APT Workbench.

AVAILABILITY. All components will be commercially available on Sun workstations under Unix and on Digital VAX under VMS in the first quarter of 1989. APT Workbench will be available for other Sybase platforms in the second quarter. Prices for APT Workbench range from \$9,600 to \$14,400 on Sun workstations, and \$19,200 to \$76,800 on Digital VAX. The run-time component costs \$3,200 to \$4,800 on Sun workstations, and \$6,400 to \$25,600 on the VAX. ● — J. Davis

• PROGRESS SOFTWARE •

A Fast Track for Building Applications

Progress Software Corporation now offers Progress Fast Track, a comprehensive, menu-driven, applications builder for its Progress RDBMS (relational database management system) and fourth-

generation language (4GL). It is designed as a productivity tool, speeding the development of applications for both developers and end users. Fast Track, itself a Progress application written in the Progress 4GL, includes a menu editor, screen painter, report writer, and QBF (query-by-forms) generator. Once a database is created (using the menu-driven Progress Data Dictionary), the developer can design menus, screen forms, reports, and QBF access to the data without having to use the 4GL. The result can be a complete application, or can provide modules of a more complex application.

HIGHLIGHTS. Fast Track enhances the developer's productivity and effectiveness in several ways.

Code Generation. Fast Track automatically generates standardized and optimized Progress 4GL code. The application can be modified within Fast Track, or the code can be edited directly using the Progress 4GL editor. Fast Track is designed to be self-documenting, including comments in the code it generates.

Visual Approach. Fast Track uses a WYSIWYG (what you see is what you get) approach to enable the developer to easily see during the design phase exactly what the end user will see on the screen when the application is done.

Security. Fast Track offers security options at the menu level and the object level. You define which users can see each menu item. For a user not included on the list, Progress automatically deletes the item and renumbers the remainder for that user. You can also define access for each QBF, report, procedure, etc.

Flexibility. One of the most attractive aspects of Fast Track is its flexibility. You can design an application from the top down, creating a menu structure and then defining the underlying processes (reports, data entry forms, etc.) invoked when a menu choice is made.

Or you can first design the individual pieces and then integrate these into the menu structure. Fast Track has been designed to easily incorporate procedures written in the Progress 4GL. Thus, an application initially generated with Fast Track and customized further with 4GL procedures can be modified within Fast Track without "clobbering" the embedded procedures.

USING FAST TRACK. The Fast Track main menu gives direct access to each of the four application modules, as well as to the Progress Help screens, the Progress 4GL editor, and the operating system. Function keys are used consistently in both Fast Track and the Progress 4GL/RDBMS, and the commands used within Fast Track are consistent among the modules. Fast Track employs a combination of ring menus and pop-up windows to lead you through the applications development process. Experienced users can bypass the menus.

Menu-Building. Setting up a menu is quite easy. When you enter the menu editor, you are asked for a menu name, a menu title (which will appear at the top of the actual menu screen itself), and a description. (This description is included in the source code comments for reference.)

You are then popped onto a blank menu window with the title you entered at the top. You type in the choices as you want them to appear on each line of the menu. Fast Track automatically numbers the items, and, if you insert or delete a line, the remaining items are renumbered for you.

At this point, you can define the process that will be invoked for each menu choice. To do this, you merely highlight the desired choice and press F1, which is exactly what the end user would do to invoke the choice. Fast Track now opens a window containing a form for defining the menu item.

The options here are to call another menu, a QBF screen, a procedure, or a report; leave the application (quit); or return (to the calling program). (You

can request a list of your valid choices any time while you are completing a form in Fast Track. The choices will be displayed in a pop-up window, and you can select one by highlighting it.)

If you choose Menu, QBF, or Report and the name you enter does not exist, you have the option to create it now (which moves you into the appropriate Fast Track module) or to come back later.

Designing Forms. As an example of designing forms, let's say you choose QBF and give a form name that doesn't exist. Fast Track asks if you want to create it. If you answer yes, the screen painter is invoked, presenting you with a blank screen and a list of options. You select the fields you want on the form from a pop-up list. The list includes the base table you entered on the form defining the QBF process, and, if a row in the base table uniquely identifies a row in any other table, that table will also be listed. For example, if the customer table is the base table and identifying a customer identifies a specific sales rep, the sales rep table will also be listed. If the list of fields contains more than one table, it changes as you highlight different tables.

You can insert and delete fields, define variables, and enter text; move elements around on the form; automatically generate a default form containing all the fields in the table; define rules for displaying the form on the screen, such as its location and whether other forms can overlay it; and copy the content from one form to another. You can also easily view the form on the screen exactly as it will be displayed in the application.

Designing Reports. The report writer also uses a WYSIWYG approach. Reports can include data from multiple tables (up to 10 in one section) and user-defined variables. Records can be grouped and sorted. The format and appearance of the report can be refined with a full-screen editor. You can define a header and summary at the report and page level, headers and detail for

any number of parallel or hierarchical sections within the report, and control breaks. Reports can be up to 255 characters wide. As in the screen painter and QBF modules, the report writer has a "view" function so the report can be run at any time from within the report writer. This is a helpful tool in the design and debugging process.

PACKAGING. Fast Track is designed to appeal to both professional developers and to experienced end users. The product comes in two forms, both marketed as additional options to the Progress 4GL RDBMS. Progress Fast Track is the complete Fast Track development system, which can be used in conjunction with either the Progress Full Development System (the complete 4GL and RDBMS) or the Progress Query/Run-Time System (which does not allow the end user to modify applications to change the database). Progress Query and Report includes the report writer module of Fast Track, and only runs in conjunction with the Progress Query/Run-Time System. Thus, the developer can give the user an easy-to-use tool for developing additional reports, but not allow changes to the database.

Fast Track is available for any platform that currently supports Progress Version 4 or higher. This includes many versions of Unix, Xenix, Digital's VAX/VMS, and DOS. The cost ranges from \$600 to \$39,000.

SUMMARY. While Fast Track may not completely handle all aspects of a complex application, it can greatly improve productivity in designing menus, forms, and reports while allowing the developer to incorporate procedures written in the 4GL. Fast Track is also a good tool for prototyping applications. The addition of Fast Track enhances the

ability of Progress to compete more aggressively, and it is an important step in the company's strategy to eventually split the 4GL from the RDBMS. The objective here is to market the 4GL Fast Track as the front-end applications development environment of choice, regardless of the underlying DBMS. ●

—J. Davis

• PC INTEGRATION •

Digital Fills the PC Gap

Digital has a lot riding on the success of Unix in the commercial arena and has invested a lot in its Unix strategy (see Vol. 3, No. 10). But if Unix is to become an office heavyweight, PC integration is essential. The two operating systems can certainly coexist. What with its multitasking capabilities, Unix makes an ideal server for DOS. Digital knows that, and its PC strategy (although much maligned in the past) looks promising.

Digital's goal is to provide a common interface from top to bottom, using DECwindows as the integration vehicle. To that end, the company has licensed two PC integration products from Locus: PC-Interface and Xsight. PC-Interface is simply a VT100 terminal emulation software program that will link PCs to Ultrix. Xsight is a DOS-based X server that turns the entire PC into an X display terminal. You can divide your screen into multiple windows with each window running a different X client application.

PC-Interface for Ultrix Version 2.3 is available immediately. The price varies from \$500 to \$6,000 according to

the number of users. (A \$500 server supports up to 16 users.) The Xsight portion retails for \$423 and will be available at the end of the month. ●

—L. Brown

Motorola: DOS Applications on Unix

Motorola is also taking bets on the commercial viability of Unix (in fact, the company is moving its departmental solutions to Unix) and thus inked two deals aimed at giving Unix a smoother ride to the office.

Motorola announced support for a DOS-to-Unix conversion software package developed by a young, 20-employee software house, Hunter Systems (Mountain View, California). XDOS directly converts DOS programs into Unix programs. The process, called binary compiling, runs as a utility, without DOS or any coprocessor boards. The program uses a "key file" to perform the recompilation process. The XDOS Analyzer codifies each DOS application and sends it to the XDOS Converter with the key file, which provides a binary set of instructions for the DOS application. From there, the converter creates a Unix application that replicates the original. Converted programs run at PC speeds or better.

Motorola also announced support for SoftPC, a software emulator from Insignia Solutions. SoftPC resides on Unix platforms and emulates a 286-based PC—kind of the reverse of Locus's developments, but that's okay; both technologies are advantageous to the future of a commercially viable Unix. ●

—L. Brown

Patricia Seybold's Computer Industry Reports

ORDER FORM

Please start my subscription to:

		U.S.A.	Canada	Foreign	
<input type="checkbox"/>	Patricia Seybold's Office Computing Report	12 issues per year	\$385	\$397	\$409
<input type="checkbox"/>	Patricia Seybold's UNIX in the Office	12 issues per year	\$495	\$507	\$519
<input type="checkbox"/>	Patricia Seybold's Network Monitor	12 issues per year	\$595	\$607	\$619

Please send me a sample issue of:

Office Computing Report UNIX in the Office Network Monitor

My check for \$_____ is enclosed. Please bill me. Please charge my subscription to:
Mastercard/Visa/American Express
 (circle one)

Name: _____ Title: _____
 Company Name: _____ Dept.: _____ Card #: _____
 Address: _____ Exp. Date: _____
 City, State, Zip code: _____ Signature: _____
 Country: _____ Bus. Tel. No.: _____

Checks from Canada and elsewhere outside the United States should be made payable in U.S. dollars. You may transfer funds directly to our bank: Shawmut Bank of Boston, State Street Branch, Boston, MA 02109, into the account of Patricia Seybold's Office Computing Group, account number 20-093-118-6. Please be sure to identify the name of the subscriber and nature of the order if funds are transferred bank-to-bank.

IU-1288

Send to: Patricia Seybold's Office Computing Group: 148 State Street, Boston MA 02109; FAX: 1-617-742-1028; MCI Mail: P SOCG
 To order by phone: call (617) 742-5200

Topics covered in Patricia Seybold's Computer Industry Reports in 1987/1988:

Back issues are available, call (617) 742-5200 for more information.

Office Computing Report

1987—Volume 10

- # Date Title
 11 Nov. User Developed Applications—
 They Know What They Need
 12 Dec. Hewlett-Packard's New Wave

1988—Volume 11

- # Date Title
 1 Jan. Office Computing: 1988
 2 Feb. Mac Attack
 3 Mar. Electronic Data Interchange—
 The New Frontier of Electronic
 Mail
 4 Apr. PC LANs Revisited
 5 May Remission or Recovery—Will
 NBI's New Direction Bring
 Renewed Life?
 6 June Service And Support *Second*
Feature: LAN Support
 7 July AI Update—Expert Systems
 Flourish While the AI Industry
 Disappears
 8 Aug. Metaphor Computer Systems—A
 Quiet Revolution
 9 Sep. Technology That Supports
 Meetings
 10 Oct. Document Processing
 11 Nov. Upper CASE Tools—Getting
 Users to Help Design Systems

UNIX in the Office

1987—Volume 2

- # Date Title
 11 Nov. Expres Initiative—Is there an
 Office Use for University
 Communications?
 12 Dec. Unix Roundup—Unix-Based
 Integrated Office Systems

1988—Volume 3

- # Date Title
 1 Jan. Beyond Unix—Creating the
 Standard Operating Environment
Second Feature: X/Open
 Champions Open Market
 2 Feb. Progress
 3 Mar. Pyramid Technology
 4 Apr. IBM's Unix
 5 May AI and Unix
 6 June SmartWare from Informix Soft-
 ware—The Quiet Contender
 7 July Olivetti—Global Bridges on a
 Standard Foundation
 8 Aug. BBN's Slate System—Delivering
 Compound Documents
 9 Sep. Distributed Databases—Where
 Are We?
 10 Oct. DECwindows
 11 Nov. Data General's Unix—A New
 Beginning

Network Monitor

1987—Volume 2

- # Date Title
 11 Nov. IBM and SNA—Meeting the
 Challenge?
 12 Dec. Micom Interlan Inc.—Learning
 From Past Mistakes

1988—Volume 3

- # Date Title
 1 Jan. Taking Stock, Looking Ahead
Second Feature: Mac to PC:
Making the Connection
 2 Feb. Excelan—Riding the Crest with
 TCP/IP *Second Feature: Update*
on Banyan
 3 Mar. 3Com—Opening a New Window
 4 Apr. OS/2 Networking
 5 May Hewlett-Packard—Riding with
 the Wind
 6 June Exploring Distributed Network
 Computing
 7 July Transparent File Systems—The
 Locus Model for DNC
 8 Aug. Novell—The Juggernaut Rolls
 On
 9 Sep. OSI Standards—A Promise
 Fulfilled?
 10 Oct. ISDN: We Still Don't Know
 11 Nov. Data General's PC Networking
 Strategy