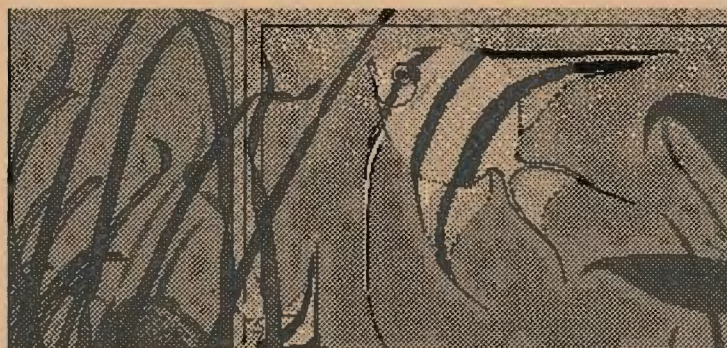


Newsletter of Stanford Palo Alto Users Group for PC

TABLE OF CONTENTS

President's Piece	Brian Christopher, SPAUG Pres.	2
Lossless Data Compression	Software	3
Insufficient Memory	System Resources	6
Windows System Resources	System Resources	7
KanjiWORD for Win	Software	8
MIDISCAN for Win 1.1	Software	9
Install OverDrive Processor	Hardware	11
Avisys 3.5: For the Birds	Software	12
QBasic for the Masses	Book Review	13
Unauthorized Windows 95	Book Review	14
Dreamware-The Sequel	Dick Harding, SPAUG	15
Want to be a Consultant?	Commentary	17
Resource Center	SPAUG	19



GENERAL MEETING & *THREE* PRESENTATIONS
KENSINGTON: The Thinking Mouse
HITACHI: Graphics Tablet
LOS ALTOS COMPUTER: System Integration
VARIAN ASSOCIATES, BLDG. 7 AUDITORIUM
WEDNESDAY, APRIL 26, 7:30 PM

President's Piece



Membership Drive

SPAUG business cards are available now for members to pass along to associates and friends for information about membership. See the card on p.5 of PRinT SCreen and ask Larry Weinberg for several to distribute.

3 Count'em 3 — April Presentation

Kensington will present their Thinking Mouse. Hitachi will demonstrate their graphics tablet. Los Altos Computer will discuss system integration.

Spring Forward

As the days lengthen and clocks move forward, the Program Committee springs forward with another communication breakthrough by initiating the use of name tags at the General Meeting. Should you care to design your own, have one left over from your company meeting, or use a blank one provided at the meeting, I think we will all appreciate the relaxed exchange.

Ouji Board

My Ouji Board has failed to spell the name of a good QuickBooks Consultant who can present this program at a SPAUG meeting. Since Intuit has not returned my calls or faxes, I am still looking. I would appreciate your assistance in this matter.

Nostalgic Return

Do you remember the early SPAUG days when members showed off their hot new programs? Excitement and wonder electrified the Stanford classroom. Now kicked out of the classroom, the excitement of the memory lingers. Here's the problem. SPAUG needs a laptop computer and LCD panel, borrowed, bought, or donated. We need to follow through on this idea. Either that or a Club BBQ at your house, we need a serious injection of fun.

Publishing Contract

In their infinite wisdom the SPAUG directors have decided to stabilize the PRinT SCreen production by paying the Editor. This personnel requisition is open now. Should you desire the excitement and verve of newsletter journalism, then please rush to your telephone and call Larry Weinberg at (415) 969-2292. Larry is logging applicants, qualifications, and remuneration.

See you on April 26 . . . *Brian*

FOURTH ANNUAL PLEDGES DINNER

PotLuck Exclusive Dinner Group Endeavor for SPAUG

Clever or not, the above acronym stands for the need for members to get involved in the functioning of their user group. Elections will be held this month and the pledge dinner follows in two weeks to give members an opportunity to make a commitment to SPAUG. It also stands for a friendly social evening with members. The dinner will be held on Wednesday, May 10. Let Mildred Kohn (415-949-1833) know what you can bring to the potluck.

The following job list is by no means all-inclusive, rather it is pretty much bare bones. SPAUG needs your support and participation for the group to be useful to members and the community. Your participation is an opportunity to practice your already-polished skills or learn new ones, and to have the satisfaction of knowing you're making a needed contribution.

SPAUG Committees/Jobs

SPARC Bulletin Board

Maintain BBS system, including hardware, security, files, messages and bulletins.

PRinT SCreen Newsletter

Publish newsletter. Editing, layout, ad solicitation, printing, and distribution.

Special Interest Groups

SIG Leaders: Organize and lead monthly or occasional special interest groups.

Product Review

Provide interface between vendors and members for product reviews. Allocate products for review in PRinT SCreen.

Membership

Maintain membership database. Follow up on membership/information requests. Provide mailing list for PRinT SCreen.

Publicity

Disseminate General Meeting information to media for publication in community calendar of events.

Programs

Arrange for speakers and product demonstrations for General Meetings.

Disk of the Month

Select shareware software and duplicate disks for sale at General Meetings.

PLEDGES Dinner

6:00 PM

Wednesday, May 10

RSVP 415-949-1833

Lossless Data Compression

Bruce Behymer, API Product Manager
and Technical Evangelist, Stac Electronics

Reprinted from Boston Computer Society via PCUG of the Redwoods

Stac electronics is a leading supplier of high performance data compression products implemented in software and silicon for personal computers.

Stacker, now available in version 4 for Windows and DOS, is Stac's best-known product. Stacker automatically compresses and uncompresses all data on user-specified disk partitions as required. (By the way, a search of documentation files accompanying the PKZip and ARC data compression software indicates that their creators tend to favor the term uncompress slightly above decompress.)

Bruce Behymer was the author of versions 1.2 and 3 of the Stacker utilities, and ten years ago was the first non-founder employee. Stac was started by a bunch of Cal-Tech students as State of the Art Consulting, hence the abbreviation Stac.

Stac initially provided hardware and software for error correction of tape drives and only later became involved in disk compression for DOS. The two issues are actually directly related, so that it is not surprising that they would be involved in both.

Redundancy—Error Correction vs. Compression

The key to understanding is that information may or may not contain redundancy (that is, extra information that can be created from the information actually present). English text is highly redundant as illustrated by the fact that even with letters removed, a block of text is still readable. We can all read the sentence: Th- qu-ck b-own f-x j-mped -ver the lazy d-g.

In this case the reader is using the remaining information to recreate the missing information. The human reader does error correction by using his or her knowledge of the language to replace the defective characters. On the other hand, if we present a number sequence such as 197-6376-762-276-534 with some digits missing, we cannot determine the missing digits since there is no redundancy.

Error correction technology adds extra information so that some number of defects in the data can be fixed, whereas compression removes as much redundancy as possible within certain limits (such as time available) in order to reduce the size of the data. Error correction should not be confused with error detection (e.g., parity).

As an example, consider a large picture of a green circle which might take thousands of pixels

to represent. A compression technique which understands geometric forms could transform that picture into a few parameters such as radius, center location, and color. This would produce a compression ratio of several thousand, since the new form might only require five bytes to describe.

Another example of compression would be to represent a body of information by providing the Library of Congress the number for a particular book instead of transmitting the actual book. These examples are very simple-minded because the mathematics of both compression and error correction are in fact quite complex.

Compression is measured by the ratio of the size of the data set before versus after compression. Obviously, the higher the ratio, the better the compression. Compression algorithms come in all different kinds, but the best methods are always those which are selected based on the knowledge of the kind of data to be compressed. A compression algorithm optimized for pictures would not necessarily work well for text. Algorithms, which can learn from the data in order to optimize their compression operation, will clearly work better, but they become slow since they require two passes through the data—one to learn and one to compress.

Algorithms can be divided into two gross classes: lossy and lossless. The former is not guaranteed to completely and accurately represent the data, while the latter gives back the original exactly. Lossy algorithms, having better compression ratios, are generally used for human-like signals such as pictures and sound, whereas lossless algorithms are critical for computer applications. There is often more information in a sound or picture than the human can detect, so it is sometimes safe to discard that information.

Compression Techniques—Run Length Coding and Statistical Coding

Run length coding is a very effective technique when the data are in the form of long sequences of identical symbols. A fax machine which is only concerned with black versus white pixels sampled from horizontal lines across the scanned paper takes advantage of this. Regions of white and black on a page tend to be large, with relatively few transitions. A given line may thus look like "wwwwwwbbbwwwwwww bbbbbbwwwwwww," which is easy to code as 6w, 3b, 8w, 6b, 9w pixels.

Cont. on p.4

Lossless cont. . . .

This is many fewer tokens than the original, which has 32 symbols. The reduction from 32 symbols to five tokens gives us a 6:1 compression ratio. This technique is very fast and easy to implement, but does not work well on ASCII text or computer data.

Statistical coding looks through the data and creates a Statistical Dictionary that contains pieces of the message. Those pieces which appear frequently are placed early in the table and are represented by a small number of bits, whereas infrequent sequences are assigned larger numbers of bits. For example, in the sequence ABCDABCBAXYSASABC, the substring ABC appears several times and could be represented by only two bits in the compressed form of the message, whereas the string XY, which appears once, would be left as 16 bits (two bytes). Common examples of this kind of coding are PKZIP, Morse code, Huffman coding, and Netware 4.0 compression.

In 1977 a new technique was presented by Lempel-Ziv, which is kind of a mixture of the two previous techniques. It, too, looks for frequently used sequences, but rather than have a dictionary, it simply refers back to a previous part of the message. If a substring appeared 89 characters ago and was five characters long, it is simply encoded as 89,5. The original Lempel-Ziv version was abandoned because it was extremely slow and poorly coded, but since that time there have been improved variations, and Stac uses this approach.

The Stac algorithm looks backwards only over a 2K region and does a hash precoding of each new character added to the region (that is, it stores the locations of character pairs to find them again without searching). This is a compromise which still gives very good compression ratios, but does not require a very large memory nor does it take an unreasonably long time to perform. Other variations of the Lempel-Ziv technique are used in the UNIX compression, LZW, V42.bis, and DCLZ. Stac reports compression ratios that average at least 2:1 and are often greater for computer-like data which includes both text and binary code.

Another dimension to compression technology is the scope over which the compression is applied. With manual utilities such as PKZIP or ARCA, the user selects the object or file to compress and must later explicitly uncompress it before using it. Speed is less of an issue since this task can be controlled by the user or done in the background on a multitasking system such as UNIX. In contrast, the Stac technology does its compression on all objects for the selected driver or disk partition. Because Stac's technology is

wired into the operating system as a device driver, it is completely hidden and transparent to the user.

Once installed, Stac's driver takes over the disk by reserving a very large part of it for its private use and then creates a virtual disk as seen by the operating system. To the user and the operating system, the disk still looks like a disk. For example, if the OS sends down a write request of 32 sectors, Stac takes the data and compresses them to something like 15 sectors. It then finds space in its own private area of the disk and records only those 15 sectors. When the OS issues a read request for the original 32 sectors, Stac translates that request to its location and does an uncompress. Stac is effectively making the disk look larger to the OS. Directories are not compressed, and Stacker keeps explicit track of them.

Stac claims that Stacker now provides more compression (2.5 to 1 in version 4) and even less redundancy than in previous versions. Microsoft claims that its DOS 6 DoubleSpace compresses at 1.8:1 to 2.0:1.

Designers now have the option of placing Stac's compression on network cards or even directly on motherboards in order to greatly speed up compression and uncompression.

The Speed Issue

Ironically, a compressed drive may run either faster or slower than an uncompressed drive. If the drive is relatively slow due to head seek times and if the CPU is fast (such as a 486), then while the head is moving, the CPU can compress or uncompress. This is free time since, at least in DOS, the CPU is idle while the disk is seeking. Because compressed reads and writes have much less data to move, the apparent disk speed is thus increased.

However, if there is a software cache, there is much less dead time since the data are in the cache. For this case, the compression produces a speed penalty.

The Downside

One of the negatives of this kind of compression is that disk errors can corrupt large amounts of data—not just the data stored at the location of the error, but also data which logically follow after and depend upon data stored in the corrupted disk area. Compressed drives are intrinsically more fragile than uncompressed drives. A compressed file can only be uncompressed starting from the beginning, and while the technology can detect an error, it cannot fix it.

Cont. on p.5

Lossless cont. . . .

There are generally three causes of disk corruption:

A physical defect on a disk. The effect is usually localized and will probably cause only limited data loss.

A program that corrupts the operating system and spews garbage onto the disk. This may prove far more damaging than a physical deficit.

An operational problem such as a system crash or losing system power at a critical moment. This may truncate files prematurely or may leave the FAT or some other critical structure in a faulty state.

To create some protection, Stac duplicates critical data such as the FAT (File Allocation Table) on both the uncompressed and compressed drives and does a modest disk integrity check before installation. Generally speaking, any compression technique can only strive to minimize damage caused by hardware problems or software errors.

It has been reported that Microsoft's original release of DoubleSpace had little defense against hardware or software problems, and that such problems could, in fact, corrupt the compressed portion of your disk. It remains to be seen how much of the problem MS has dealt with in the DOS 6.2 release. Caution: MS DOS users who want to use compression should upgrade to DOS 6.2 or to Stacker.

Disk Compression Strategies

Carefully choose if, when, and where you will use compression!

If you have ample disk drive room available, it may be unwise to use disk compression products such as Stacker and DoubleSpace due to the inherent risks of disk compression. However, if your CPU can compress data being written to disk or uncompress data being read from disk faster than the disk's I/O data transfer rate, you might choose to use compression on the basis of a performance benefit.

If you have a laptop or notebook computer, you probably suffer from inadequate disk space. Here, disk compression can be very useful—and may be essential if you can't expand disk capacity. By the nature of how we use such portable systems, they generally contain less information that is critical and unique, and new information generated on the portable is probably copied to a desktop system frequently.

On any system an effective strategy for using disk compression is to store commercial software

packages on the compressed drive and to store important original data on the uncompressed drive. A commercial software package can be reloaded if it becomes corrupted—an inconvenience but not a disaster.

A tip: when you install a software package on a compressed drive, inspect its directory for files created after installation (e.g., configuration files). Copy those files to an uncompressed drive or a diskette as a backup.

Applications which produce large volumes of data for temporary storage may lend themselves well to compression. For example, data files containing voice mail (digitized sound) or measurements from devices attached to a PC can occupy considerable amounts of disk space. Of course, programs storing such data to disk may already compress the data.

Any time you are using disk compression it is essential to run CHKDSK periodically (or any program that checks disk data integrity and understands the format of the compressed disk such as Disk Doctor in version 7 of the Norton Utilities).

Suggestion: The last line in your AUTOEXEC.BAT file can run CHKDSK on the compressed disk partition.

Since today's disk drives cost under \$1 per megabyte of storage, the best advice may be to always consider buying a larger disk drive (or adding another disk drive to your system) before considering compression.



Membership Drive

SPAUG
Business Cards

SPAUG Stanford Palo Alto Users Group for PC

DOS-Windows-OS/2
Monthly general meetings // Product demos
Newsletters // Special interest groups
Tech support from members
Electronic bulletin board 415-321-4497, N81
\$35 Annual Dues **For membership info, call**
Beverly 415-329-8252 / Larry 415-969-2292

"Insufficient Memory" says Windows ... Is This True?

Various authors and reprinted/excerpted from
Random Output, East Bay FOG; *Toggle*, Seattle-Tacoma PCUG; and
Orange Bytes, North Orange County Computer Club

Actually, the message "insufficient memory" is ambiguous. At first you'd probably think it refers to the large block of global memory Windows uses to load your application program code and data and that you can expand by installing upgrade memory in your PC. But that's not necessarily the memory this message refers to.

It's just as likely that the message refers to much smaller pools of memory that are deep inside Windows and that your programs never access directly.

What are those memory pools? They're small (64K) blocks that are called heaps and are used as work space by two of the main components of Windows: USER and GDI. Collectively, the USER and GDI heaps are called system resources.

USER controls windows themselves, dialog boxes, and some elements of Windows-like menus. As a program creates and manipulates windows, menus, and dialog boxes, it causes USER to devote more and more heap space (max 128K) to it. If USER's heap space fills, you'll get the "insufficient memory" dialog box.

GDI (which stands for Graphics Device Interface) provides the internal tools that programs use to present information visually. It handles lower level requests that involve bitmaps, fonts, icons, and drawing elements. If the heap space (64K) is filled, you'll also get the "insufficient memory" dialog box.

Windows 3.1 GDI heap can be exhausted by just the icons in one window if you are using a high resolution graphics driver for an SVGA or better board. At 32K colors, some windows will not simultaneously display more than 28 icons and at 16M colors, perhaps less. This is with no programs running, just the window at full screen. The answer to this is to minimize the window when running the program or to create more windows with fewer icons in each. Microsoft says that they will address this problem in future revisions of Windows.

The heap sizes are unaffected by how much additional memory you install in your PC—whether 4, 8, 16 or more megabytes of RAM, the USER and GDI heaps will be the same fixed size. The more programs you run, the more likely you are to fill one of the heaps. And the more demanding these programs are, the more heap space USER and GDI have to use to meet their requests.

One way to identify programs that are "resource hogs" is to use Program Manager. Before you start a program, access the Program Manager's Help menu, then issue the About Program Manager command. Check the System Resources percentage it reports. Next, start your application program, switch back to the Program Manager, and check the percentage again. You'll notice that the percentage dropped, sometimes a little, sometimes a lot.

Another way to monitor memory usage and system resources is to use the SysMeter program that comes with the Windows Resource Kit (a modestly priced manual and diskette published by Microsoft and available from many software retailers). SysMeter displays three graphs that show the percentages of system resources that are free. If a resource is so low that it may affect your system's performance, its graph appears in red. You can minimize SysMeter to an animated icon on your desktop that displays the status of global memory and the USER and GDI heaps as you work.

Under Windows 3.1, the only sure way to avoid running out of USER and GDI heap space is to keep the number of programs you have active at one time under control. Close programs you don't need to have open, especially if one that you do need is a resource hog.

Admittedly it's a frustration not to be able to use all the global memory on your PC because Windows internals are constrained by small heap sizes. But that's the way it is.



April Mergermania

Vatican City (XP). In a joint press conference in St. Peter's Square today, Microsoft and the Vatican announced that Microsoft will acquire the Catholic Church for an unspecified amount. Pope John Paul II will become Microsoft's Senior VP in charge of Religious Software. The Julian calendar is part of the acquisition, allowing Microsoft to issue Windows 95 on an appropriate date.

The move could spark a wave of mergers as other churches scramble to strengthen their position in an increasingly competitive religious market. Rumored (appropriately enough) is an Intel takeover of the Church of Scientology.

Windows System Resources

Sid Bala

Reprinted from Chicago Computer Society via Buffalo IBM PCUG

System Resources: The key to understanding Windows memory errors

Any Windows user can tell you that certain memory errors simply stop you from completing a particular operation, and others can cause Windows to crash. Next time you are faced with a memory error, if possible, click on the ABOUT PROGRAM MANAGER... command from Program Manager's HELP menu to check your system's available memory. This command opens the About Program Manager dialog box. Often, you'll find that the system reports plenty of free memory. In such cases, despite what the error message indicates, the problem is very likely a depletion of system resources, not a shortage of memory to run applications.

System resources explained

The bulk of system resources is made up of two 64KB blocks of RAM (random access memory): the "GDI heap" and the "User heap". The GDI (Graphical Device Interface) heap stores handles and pointers to the graphic objects (such as bitmaps and buttons) that constitute Windows' graphical user interface. The User heap stores actual components of the user interface, such as dialog boxes and windows.

How do system resources become depleted?

DOS, not Windows, limits the size of the system resource heaps to 64KB each, regardless of how much RAM your system comes with. This limit can constrain the programs you run, especially if your programs contain many graphics or device interface components.

Applications that use complex toolbars or frequently manipulate graphics place a high demand on the GDI heap and can consume it very quickly. The User heap, on the other hand, decreases when you open a window or a dialog box.

Unfortunately, some applications, upon exit, don't completely free up the RAM they used in the system resource heaps. Running such programs several times can deplete the system resources and generate a memory error. To prevent Windows from crashing, you should prevent your system resources from falling below 30%.

Solution: Minimize the depletion of system resources

1) Maximize the amount of free system resources at startup by minimizing the number of icons displayed and windows open when Windows starts. To do so, you start by closing all applications

except Program Manager, removing all applications from the StartUp group, and minimizing all program group windows. Next, you select SAVE SETTINGS ON EXIT from Program Manager's OPTIONS menu. Then, you exit Windows and restart.

2) Run a freeware utility such as COMPACT.EXE to compact memory before starting an application. COMPACT.EXE is available on many bulletin boards.

3) Check for applications that might not release system resources when you shut down. Run each application individually, checking the status of your system resources before opening and after closing the application. If the system resources are below 70 percent after closing the application, chances are good that the program isn't releasing the resources. Look for newer releases (that will not hang on to system resources) of such greedy or retentive software.

3

Creative Computer Workshop Classes

on Power Macintoshes & Pentium PCs

Introduction to the Macintosh

(One Sunday afternoon session; Macintosh only; \$75)

Sundays, 1-4 P.M. April 9; May 7;
June 18

What's New in Adobe Photoshop 3.0

(One weekday evening session,
6:30-9:30 P.M. \$95)

Wed., April 12

Fractal Design Painter 3.0

(One Sunday afternoon session;
\$120)

Introductory: June 4

Advanced: April 2, April 23, May 21,
June 11

Intro to Adobe Photoshop 3.0

(Four weekday evening sessions; \$195 plus \$50 for
Adobe Classroom in a Book with CD-ROM)

Sessions beginning on Thurs., Apr. 20 & Mon., Apr. 24



Creative
Computer
Workshop, Inc.

145 Hamilton Ave.
Palo Alto
Tel. 415-328-5048



KanjiWORD for Windows

Bob Schneider

Reprinted from *Redwood Chips*, PCUG of the Redwoods

KanjiWORD for Win

Pacific Software Publishing, Inc.

Mercer Island, WA 98040

(206)232-3989. List: \$199

System Requirements: Win 3.1, 386 min, 3.5" floppy drive, 5MB for installation.

Pro: Easy-to-use Japanese/English word-processor. Excellent front-end processor.

Good English-Japanese dictionary. Exports nicely into other Win programs.

Con: Significant bugs. Manual could be better.

Whenever I tell people that I use my computer for writing Japanese, the first response I usually get is, "Isn't that really hard to do since Japanese has thousands of characters?" The answer to that question—and the essence of a good Japanese word-processing program like KanjiWORD—lies in the software's front-end processor (FEP). That's the part that takes the out-loud reading of the text you want (which is easy to tap out on an ordinary keyboard—lots of *ka's*, *ko's*, *ju's*, etc.) and puts it into written Japanese. The problem is that for each sound there are many kanji, or characters. For example, *ki* can be written with more than 80 kanji. So, in a worst-case scenario, you input *ki* and hit your spacebar 80 times or so, cycling through all the characters that can be read *ki* as they pop up on your screen until you can select the right one.

Let's say, however, you want to write a word that has two or more sounds—say, *kisha*, comprising a character read *ki* and a character read *sha*. Well, that makes life a lot easier. There may be only five words that are read that way (which would still be a lot of "homonyms" in English, but not at all uncommon in Japanese). After inputting *kisha* in Roman letters, you would want to select the word you need from among the five two-character sets that the FEP supplies. However, what if your FEP can come up with only four of the possible combinations, leaving out the one character set you want? Then you have to look up *ki* and *sha* independently, which, as we have seen, can be a time-consuming operation. A good FEP should thus be able to supply all of the possible words for a given reading of a compound when you ask it to.

Rather than input-and-convert word by word, however, it is much faster and easier if your program can handle whole phrases and sentences in gulp—i.e., you enter the entire out-loud reading in Roman letters, hit a key, and there's the Japanese you want. This is where a top FEP really gets to show its stuff. One word processor used in Japan is famous for being able to come up with

"*kisha no kisha ga kisha de kisha shita*" in a single shot. The four *kisha* mean (a) your company, (b) reporter, (c) train, and (d) return. The full sentence means, "Your company's reporter returned by train." Here the FEP has dissected the syntax and meaning of the sentence and put into Japanese the four *kishas* written with characters as well as the "word-relationship" and "tense" components of the sentence, written not in kanji but in the Japanese syllabary known as *hiragana*. If there happened to be any Western-language-derived words in the sentence, the FEP would have to be able to detect those as well, and put them in the syllabary called *katakana*. (I hope this entire discussion has been useful, but it's undeniably simplistic, and thus my apologies to Japanese scholars.)

KanjiWORD's FEP can't quite handle the four-*kisha* sentence, but it is far better than that of the Japanese WP program I previously used, and it suits my own needs fine. (Frankly, my own Japanese remains sadly ungrammatical, so there would be something desperately wrong if it could figure out everything I was writing.) Reviews of the program that I have read indicate that even native speakers find KanjiWORD's FEP relatively robust. Remember, though, that KanjiWORD is not a program for learning Japanese—those fluent in the language will use it best although it's certainly accessible to intermediate students.

I found the documentation so-so—I didn't have much problem learning KanjiWORD, but important information required for importing files from other Japanese WP programs was left out. As you might expect, KanjiWORD's word-processing features are not as strong as those of, say, WordPerfect. For example, there is no *Undo* command. On the other hand, I am using a full range of TrueType fonts. (Inputting English, of course, is no problem.) Important formatting features like borders and shading are available. The program's 50,000-word English-Japanese dictionary is very good, even containing some fairly obscure words like *eleemosynary*.

Cont. on p.10

MIDISCAN for Windows 1.1

Gwenda Davies

Reprinted from *Redwood Chips*, PCUG of the Redwoods

MIDISCAN for Windows 1.1

Musitek Music Recognition Technologies

Ojai, CA

805- 646-8051

List \$299

System requirements: Windows 3.x, 4Mb RAM, mouse, flatbed, edge-feed or handheld scanner (black and white, gray scale or color with minimum output resolution of 300 dots per inch), imaging software capable of outputting TIFF files. **Recommended:** 16 MHz 386 or better, 4Mb RAM, a black and white hand-scanner, and imaging software capable of "auto-stitching."

Pro: A unique means of converting a printed musical score to a MIDI sound file.

Con: The MIDISCAN editing procedure is time consuming and requires knowledge of music notation.

For purposes of testing the program, I used an IBM clone 486/33, 8Mb RAM, and a Logitech ScanMan Plus hand-scanner set at 300dpi. I also used an HP ScanJet IIP flatbed scanner set at 350dpi.

For my sample data, I scanned several pages of printed music from different sources, including popular song sheets, laser-printed copies produced from a music notation program, and photocopies of music with varying notation sizes and print quality.

MIDISCAN processes scanned uncompressed TIFF images of a musical score at a rate of 3-5 minutes a page to create a MIDI file. However, before the MIDI file is created, one must use the graphic editor to make corrections to the recognized score stored as an internal file called MNOD (Music Notation Object Description). The fun part of using MIDISCAN is to watch the reconstructed MNOD version appear line by line in the lower part of the screen, mirroring the scanned TIFF image above.

The not-so-fun part is editing the mis-recognized or omitted notation. The notation toolbox contains all the necessary musical symbols for editing. In addition, selected "quick keys" provide useful editing shortcuts. It is important to understand a little music theory when identifying errors, adding omissions, and correcting misrecognized notes. Once the MNOD file is edited, it is ready for conversion to a MIDI Type 1 file. Each staff is assigned a single MIDI track. The MIDI file will accommodate up to 16 tracks. For playback software, I used Sound Impression version 3.4C with a Pro Audio Studio sound card.

The MIDISCAN manual is compact and well designed. Musical terms, explanations, and instructions are all written in very clear, concise, easy-to-read English. Graphics and charts are well placed and effective. The introduction includes a

helpful glossary of musical and computer terminology. The tutorial is basically an editing manual, and the reference section is an essential troubleshooting guide. If one follows the tutorial, it takes very little time to learn to use the program.

My original purpose for use of the MIDISCAN was to scan a large quantity (380) of poor-quality photocopied single-staff-line musical scores to form a database of MIDI files. I was attracted by the advertisement that "MIDISCAN will even handle poorly printed sheet music and music printed with smaller fonts."

I abandoned my mission when I discovered that I would have to rephotocopy each score to enlarge and darken it before scanning. I would then have to clean up each TIFF file, removing all the unwanted symbols before processing in MIDISCAN. I would then have to edit each MNOD file before creating the MIDI file since many of the copies contain print irregularities, and some pieces are written with nonstandard measures.

I soon learned that the major role of the MIDISCAN user is to edit...Edit...EDIT! Here are some reasons that errors occur in processing, and suggestions to improve the accuracy of recognition:

a) Poor print quality of original score.

Photocopy on dark setting or scan on dark setting to enhance fine staff lines and washed out notation.

b) Use of hand-scanner.

Guide with a T-square to keep the scanner steady. Photocopy to enlarge and darken small or poorly printed scores before scanning. "Auto-stitching" works well, but expect to edit at the

Cont. on p.10

"seams." The scanned image from a hand-scanner will not be clear enough for processing without editing the MNOD file. For best results use a flatbed scanner.

c) Extra symbols on or near the staff lines interfere with the recognition of notes.

Clean up the TIFF image file in a graphics program such as Paint Shop Pro or Ansel to erase smudges, guitar symbols, trills, dynamics, abbreviated expressions, and fingering markings before processing in MIDISCAN to create the MNOD file.

The product is unique and innovative. The concept of creating sound from the printed page without having to play an instrument is exciting and will open new doors for many musical projects.

To avoid the frustration of time taken to edit MNOD files, I would consider purchasing MIDISCAN only for use with a high-quality flatbed scanner. I would not stretch the limits of MIDISCAN by attempting to scan nonstandard scores or poor quality reproductions of scores.

I hope I have identified for the reader the margins within which MIDISCAN is happiest in perfecting its unique task of converting sight to sound.



It would have been nice to end my review here, but unfortunately I'm obligated to add that, in the six weeks I've used KanjiWORD, I've discovered several significant bugs. Most egregious is an absolutely devilish maneuver where it goes into your general Windows printer settings and changes the default for number of copies to whatever number you've last asked KanjiWORD to print. Perhaps I'm somewhat inexperienced, but I've never seen any program do something that dastardly. A company representative on CompuServe confirmed that, yes, the same thing has happened to him—and the vendor is having difficulty fixing it.

Scurrilous as this bug is, it's easy to work around. However, I've found other problems. Once, I opened a KanjiWORD file in WinWord, and then was unable to open it in the program itself. Another time, while switching between two open files in KanjiWORD, I got hung up and was unable to input. I didn't lose any data, but it was still disturbing. In short, I'm slightly edgy about using the program, so I save my work regularly. I love using KanjiWORD and it will remain on my machine, but I hope version 3.0 eliminates most of these problems.



Microsoft Clarifies Licensing of Bob™

Redmond, Washington (XP). In response to customer inquiries, Microsoft today clarified the naming policy for *Bob™*, its new software product designed for computer beginners. Contrary to rumors, Microsoft will not demand that all persons formerly named "Bob" immediately select new first names.

"I don't know where these rumors come from," commented Steve Balmer, Microsoft Executive VP. "It's ridiculous to think Microsoft would force people outside the computer industry to change their names. We won't, and our licensing policies for people within the industry will be so reasonable that the Justice Department could never question them."

The new licensing program, called Microsoft *TrueName™*, offers persons who want to continue being known by the name Bob the option of doing so, with the payment of a small monthly licensing fee and upon signing a release form. As an added bonus, *Bob™* name licensees will also be authorized to display the Windows 95 logo on their bodies.

Persons choosing not to license the *Bob™* name will be given a 60-day grace period during which they can select another related name. "We're being very lenient in our enforcement of the *Bob™* trademark," said Balmer. "People are still free to call themselves Robert, Robby or even Rob. Bobby, however, is derivative of Microsoft's trademark and obviously can't be allowed. He added that *Bob™* Harbold, Exec VP and COO, has become the first Microsoft *TrueName™* licensee and will have the Windows 95 logo tattooed to his forehead.



Installing an Intel OverDrive Processor

Peter M. Kendall

Reprinted from *Bibmug.txt*, Buffalo IBM PCUG

In August of 1994 the mail-order prices of Intel DX2 OverDrive processors dropped dramatically, probably due to the long overdue appearance of the DX4 chip on the market; a DX2/50 chip can now be obtained for about \$250, and a DX2/66 for less than \$300. This seemed to me to be a very cost-effective way of "pumping up" my 486DX/33, which has a respectable amount of memory and reasonably fast video and hopefully ought to have several more years of use left in it.

For anyone contemplating the purchase of an OverDrive chip, the big question is: which one? You are probably already aware that the DX2/50 chip replaces a 25-MHz processor and the DX2/66 chip replaces a 33-MHz processor. However, both of these chips come in two varieties, one having 168 pins and the other 169, and although there are a few exceptions, in general it is best to get the one which matches the socket in which you intend to put it.

If your PC already has an empty OverDrive socket OR an empty 487 coprocessor socket (they are actually exactly the same), lucky you; you know which chip you need, and installation will be a snap. Just be careful not to mistake the Weitek coprocessor socket for an OverDrive socket: the Weitek socket has 15 pins on a side, and the OverDrive socket has 17 (or 19). If you do not have a suitable empty socket, you will have to remove either your 486DX CPU or the 487 math coprocessor in your 486SX system to install the OverDrive chip. Here again, you might just be lucky enough to have the chip in a "zero-insertion force" (ZIF) socket, in which case you just need to raise the little lever or loosen the little set screw, and you can easily remove the processor and check the socket. I believe the 486DX chip has 168 pins and the 486SX and 487 have 169, but I strongly recommend not taking my word for it. Check the documentation that came with your system, or call the system manufacturer, if you want to be sure. You could even call Intel: it is certainly in their best interests to be able to tell you what will work in your particular system.

Let's assume that you have the correct OverDrive chip, and that you were not lucky, and you need to install it the old-fashioned hard way (like I did), by forcibly prying the old CPU out of its socket. Before you do, check to see which corner of the chip has the dot, or the rounded corner, or other identifying mark, because the new chip HAS to be inserted in the same orientation.

Along with your new OverDrive processor you will get a chip removal tool, which looks like a miniature garden rake. Its use is (hopefully) self-evident: it is meant to go between the bottom of the chip and the top of the socket, and between the pins of the chip. This assumes that you can even reach all four sides of the CPU socket: good luck, most motherboards do not make life so easy. If your CPU socket is particularly difficult to get at, you might have to go so far as to remove your whole motherboard; it is a pain in the hind end, but you can do it. Just be very careful to note the correct locations of all wires which you disconnect from the motherboard. As far as the chip itself is concerned, the big trick is to pry on opposite sides (insofar as possible), a little at a time, keeping the chip more-or-less straight and edging it up until it is loose enough to remove by hand. Be sure that you have grounded yourself on the PC's case before handling the chip, since these things are sensitive to static electricity.

Okay, you're halfway home! You've removed your old CPU without bending its pins too badly; you should put it in a safe place, you never know when you might need such a thing. Unpack the OverDrive chip, ensure that its pins are straight, gently insert it into the socket, making sure the pins slip into the proper holes in the socket, and push down to seat it. Sounds easy, does it not? Here's what happened to me when I did all of the above and then turned the system on: nothing. No disk activity, no video activity, just a cold blank monitor screen. Stay calm; take a couple of Maalox; Read The Fine Manual. For this particular symptom, the manual gave some really intelligent suggestions, like "be sure the power cord is plugged into the wall outlet." Ha, ha, Intel, very funny. But it did also ask, "is the chip fully inserted?" I thought the chip was fully inserted, but obviously I was mistaken; in fact, it took a good bit of pressure to get the chip fully into the socket. Having not bothered to remove the motherboard, I was watching it bend under the pressure of pushing down on the chip, and having visions of cracking the board. But somehow the chip yielded to gentle persuasion, and did get wedged down far enough to make electrical contact, and somehow the board did not break, and the machine booted right up, and I was VERY relieved.

Cont. on p. 12

The last part of the installation involves verifying that your new CPU is indeed operating at the speed at which it is supposed to; virtually any diagnostic utility can check this, and Intel even includes their own utility disk with the processor. It is not at all uncommon that you might need to readjust some jumpers on your motherboard to get everything to work. On an older machine, you might even need a BIOS upgrade, but just as an example, my machine is two years old, and I did not need to make any adjustments to it other than the actual chip replacement. So although caution, patience and a good understanding of your own system are advised, installation of an OverDrive processor is really not a difficult project.

By now, you're probably dying to know the bottom line, meaning just what sort of results this upgrade can deliver in the real world, running real applications. Well, of course the processor runs at twice its former speed, but in the real world, processor speed is not everything.

A certain percentage of the computer's time is spent doing, for example, disk and video operations, and the relative amounts of these various operations depend on the speed of your disk and video, as well as on the nature of the particular application. If your video is particularly slow, or if you're a little short on memory and need to read from your hard disk fairly often, a large portion of the computer's time will be spent on these operations, which are largely independent of processor speed.

In general, though, if the system is fairly well balanced, disk and video operations consume about 1/3 to 1/2 of the computer's time, so you would expect an overall improvement of about 1/3 by doubling the processor speed. This pretty much matches what I observed in my system. In fact, I am really quite pleased with the performance, and it was certainly cheaper than a whole new system. So the Intel OverDrive chip is an option which I think any 486 owner ought to at least keep in mind.

Software Review: For the Birds

Frank Eaton

Reprinted from *p.bug*, Palm Beach UG

Avisys 3.5

Perceptive Systems, POB 3530

Silverdale, WA 98383

800-354-7755

This article is not exactly "for the birds," but it is for "birders." And just to be sure we are communicating with words you know, a "birder" is a serious, dedicated and list-making bird watcher.

It is the list-making aspect that this article addresses. My wife is a list-making, hardcore birder, and I have been helping organize her lists of sightings. There are lists and lists and lists—such as:

- * Yard List
- * City List
- * County List
- * State List (one for every state)
- * USA List
- * World List
- * Trip Lists
- * Birds I Have Seen at 60 MPH List
- * Et al

As you can imagine, the computer data management programs are ideally suited for organizing bird-sighting lists. But choosing the particular software can be daunting. After doing some research, asking knowledgeable birders, and trying out some programs, I can highly recommend "Avisys" by Perceptive Systems. Not only is it eminently easy to use, the personal assistance is most remarkable. The 800 number puts you right through to the author. No waiting and button pushing. He is helpful, patient and pleasant.

The current version is 3.5, lists at \$89.95, plus \$4.00 shipping, and is available direct. The author, Jerry Blinn, will answer the phone, unless the answering machine takes over; the line is open 24 hours per day, seven days a week.

Jerry's slogan is "Fast as a Falcon — Powerful as an Eagle — Friendly as a Chickadee."

And it is actually true.



Book Review: QBasic for the Masses

Bill Allen

Reprinted from *Orange Bytes*, North Orange County Computer Club

Teach Yourself ... QBasic

Chuck Butkus

MIS: Press. \$24.95

Want to learn computer programming for only \$24.95? You can get your start by learning to use the QBasic programming language included with every DOS 5 or DOS 6 package. Yes, it is a complete programming language based on Microsoft's QuickBasic.

Go to your DOS directory. You will find two QBasic files, QBasic.hlp and QBasic.exe. Run QBasic.exe. (QBasic.hlp is the interactive help file for the QBasic program.) You are now ready to program your computer. But wait! The DOS manual has practically nothing in it about QBasic. Where do you go for help? That's where the \$24.95 comes in. Chuck Butkus' new book, *Teach Yourself QBasic*, leads you through the rudiments of QBasic programming. Most importantly, Chuck teaches you the right way to program by learning to use modular building block routines instead of pounding out "spaghetti" code. He emphasizes the good habits of flow diagramming, using comments extensively, etc., to start you off the right way.

Why would you want to learn to program your computer? Beyond a natural curiosity about how your computer's innards work or a desire to better understand shop talk you will find that QBasic can solve many on-the-job problems for you, in fact reducing frustrations and delays while learning a useful and profitable computer skill. You may find it may do certain things better than can your spreadsheet. If you count programming time as well as runt time it may be faster also. Butkus' book illustrates QBasic's usefulness with the programs he provides on the floppy disk included with the book.

The first four chapters of Butkus' book lead you through the process of correctly structuring any computer program, including the critical steps that should be done prior to typing the first line of QBasic code. Take it from someone who has had to unlearn most of the bad coding habits, Butkus' book can be of inestimable help.

Mastering QBasic opens the door to QuickBasic, a more high powered Microsoft Basic language that can compile standalone Basic

programs that generally run in a fraction of the time of QBasic. In turn you can later progress to Pascal, Fortran, Ada, and ultimately to the various versions of C in which most commercial software is now written.

QBasic will run most of the existing GWBasic and BasicA programs that you find on various bulletin boards. A note of caution: many of the "QBasic" programs on such boards are in fact Quick Basic, not Qbasic programs. They may or may not run in your Qbasic environment since Quick Basic has more functions/statements than does QBasic, in effect being a superset of QBasic. By the time you are ready to download some of these bulletin board Basic programs you might want to consider acquiring QuickBasic, which has been available for as little as \$50 by mail order.

Many engineers, scientists and accountants have needed to learn computer programming quickly to get their work completed on time. They have typically jumped straight into coding in C without first learning the good programming procedures of flow charting, modular structuring, copious commenting, etc., discussed in Butkus' book. Consequently they experienced frustrations and delays, sometimes to the point where they had to hand the job to an accomplished programmer. Developing rudimentary programming skills need not be that difficult. You can get off on the right foot using Butkus' book. Avoid similar frustrations and delays while learning a useful and profitable computer skill.

Take it from someone who has had to unlearn most of the bad coding habits, Butkus' book can be of inestimable help.

SP AUG

Your user group
Use'er or lose'er

Book Review: Unauthorized Windows 95: Developer's Resource Kit

James Alexander

Reprinted from *Blue Chips*, Utah Computer Society

Unauthorized Windows 95

IDG Books, Released Nov. 1994

Andrew Schulman

I never do book reviews for two reasons. They are time consuming and the material is usually six to twelve months out of date. Rule number one: Never say Never. True, this book is time consuming, but when you find something good you enjoy it as much as possible. My wife caught me reading it at nearly 2 a.m. Untrue, this book contains material that had not yet been put on the newsstand until a week after its release. If nothing else, everyone interested in the desktop operating system battle must read the first 128 of the 610 pages. I acknowledge that the last 20 of the 128 pages are somewhat technical, but Schulman does such a great job of explaining what is happening that I actually believe I understand how Windows 95 will work.

Mr. Schulman is very knowledgeable, having authored or co-authored *Undocumented DOS*, *Undocumented Windows*, *The United States versus Bill Gates*, *Extending DOS*, and various articles in Microsoft Systems Journal, Newsweek, and PC Week. InfoWorld's Windows Manager, Brian Livingston strongly endorsed this book Nov. 14 and 21, and Dec. 12, 1994. The Dec. 12, 1994 column reviews the fact that Mr. Schulman was locked out of the Windows 95 on-line beta forums for about two weeks following the release of the book. Several industry notables made inquiries about the condition, and were told that it was a clerical mistake. Mr. Schulman is back on-line to the satisfaction of several leaders.

There are two versions of the book available. The Developer's Resource Kit (DRK) contains a CD-ROM "SMASH Hits for Programmers," and the companion diskette which accompanies both versions. DRK also has some additional printed material assisting programmers in writing better application code for the next release of Microsoft Windows 95. (Dec. 10, 1994 MS moved delivery to August 1995.)

It should be made very clear that Mr. Schulman is first and foremost a dedicated Microsoft follower. When asked at the press conference if he was dissatisfied with Windows, he made a very strong response that he approved of the tough choices Microsoft made in developing Windows 95. His contention with Microsoft is really an issue of technical reality versus marketing hype. Schulman demonstrates that a lot of the DOS code

that was no longer needed (marketing) is in fact still part of the core code. Though Schulman was dealing with midsummer 1994 beta code for Windows, the core code will not be changing. The reason for no change is simple: changes now would delay the release date for up to four years.

Though there is some serious thought placed upon the idea that previously, vaporware was not allowed, but now marketing-ware is. The other marketing issue deals with 32-bit versus 16-bit operating systems. Based upon the facts, there is still a lot of 16-bit code in Windows 95, but that to be all 32-bit code would require 40% more resources on everything (hard drive, RAM, and even processor). Microsoft's marketing people seem to be greatly hung up on proving that Windows 95 is new from the ground up. In fact a lot of the code has been there since 1990 with Windows 3.0 and some from Windows 2.0 in 1988. Schulman contends that this is good because the code in question is stable and does the job right. More important, most of the key new code comes from Windows for Workgroups 3.11.

Despite the elaborate discussion about the problems with marketing, the book really shines in discussing how the Windows 95 operating system makes and processes calls to the many hundreds of functions handled by the system. This last area makes this book the must have "bible" for anyone writing code for the Windows 95 operating system.

30

SPAUG BUDGET 1995

Revenues

SMALL

Expenses

LARGE

Net Support SPAUG's Membership Drive

Dreamware—The Sequel

Dick Harding, SPAUG Member

In a previous article from the heart of Silicon Valley, I mentioned that I was using a program called Chief Architect version 2.0 to design a home I want to build on a beautiful island in the Caribbean (see drawings). There is an interesting symbolic similarity between an island and a personal computer—the tension between social need and self-sufficiency. An old song says "No man is an island..." and yet there is an axiom that says: "If you want anything done right, you have to do it yourself."

Designing a home on a personal computer addresses both ends of this tension. There are few things in life more socially motivated than designing your own home. There are few things in life that give architects, builders and interior designers more frustration than to try to get the owner to realize what he or she wants. Changes to a house design in the design phase, and especially in the construction phase, can radically drive up the cost of a house, as well as drive the architects, builders and interior designers crazy.

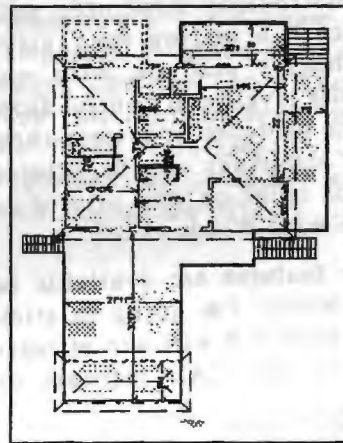
What is needed is a way quickly and easily to visualize a house design both inside and out. Most Computer Aided Design programs start with the complexities of the software rather than the design of the house. I define dreamware as CAD software that allows you to focus on the design of your dream house rather than getting hung-up on the technicalities of the drawing program. Version 3.0 of Chief Architect will allow you to have industrial strength rapid eye movements. It will help your dreams become reality.

Some individuals who are not in the building trades but want to make a few modifications to their homes may think \$995 is too much to pay for a CAD package for their small project. However, I recently spoke to a builder here in California who was making a \$2500 modification to a client's home. The project required \$6000 in permits and 19 sets of drawings. From another perspective, the plan checker for our town uses Chief Architect himself and recommends it.

Recently I was looking at a high end, \$5000 CAD program that has very powerful features. When I expressed concern about the level of difficulty of the user-interface to an architect who uses that program as well as Chief Architect,

he replied, "Well, when you are using Chief Architect, you are in a whole different league." There are a number of very powerful and expensive CAD programs on the market that are simply left in the dust by Chief Architect's ease of use. Chief Architect version 3.0 is not to be considered a complete CAD program for architects but is a superior design program. Chief Architect can be used with other programs that can accept .DXF file formats.

What makes Chief Architect version 3.0 so hot? As a computer consultant I can tell you that writing software is in many ways like building a house. It is much easier to build and enhance, if you start out with the right design. As first beta

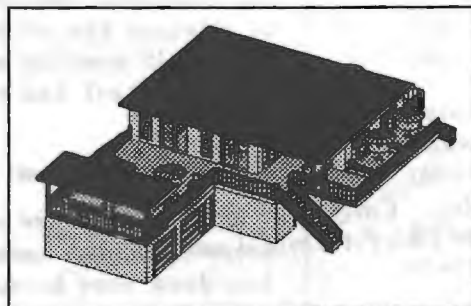


tester and first cheerleader my enthusiasm for Chief Architect is because it was designed to be intuitively easy enough for you to sit down with a client and design a house, or remodel a kitchen or bathroom while making roof, cabinet, window and furniture color and style choices, etc. This ease

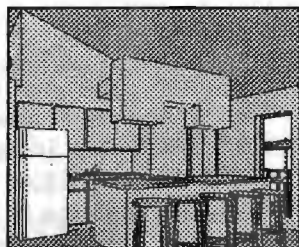
of use was accomplished not only with a superior software design but with a new programming technology called Object Oriented Programming. Writing a program with OOP is similar to building a modular home. You start out with flexible modules and a strong frame and then you can add all kinds of features and amenities on that frame without painting yourself into a corner.

In other words you go from the basic floor plan to advanced features. That is also how you design a house using the software. You don't draw lines. You pull walls to any shape and size desired for a floor plan and then you put windows and doors, etc., in the walls. You then click a button and get a 3D view of the house which you can walk through changing things in 3D mode. You can view the house both inside and out from a number of 2D and 3D views simultaneously including multiple floor views and elevations. If you change one view the others are automatically updated on your screen.

Cont. on p.4



In 2D mode you can get automatic dimension lines in one click, place electrical outlets in a room—one click, a complete Materials List—one click, draw a stair case—one click.

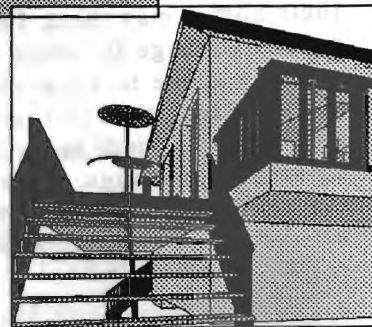
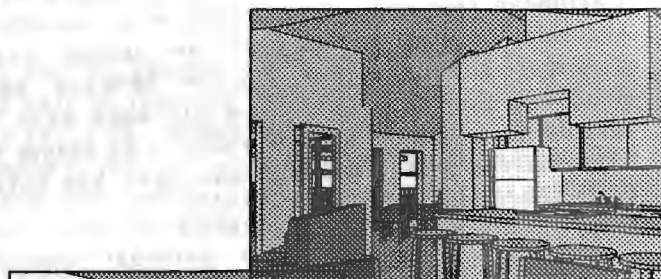


In 3D mode you can change, size, color and move windows, cabinets and furniture, etc. Complex roofs (hips, half hips, gables, sheds, mansards, saltboxes, gambrels, gull wings, Dutch colonials, or combinations of these styles) can be designed automatically and then edited room by room.

You can produce full exterior elevations on multi-floor structures in one click. You can cut a cross-section on your multi-floor structures with one click. You can discard a ceiling in a given room and produce cathedral ceilings with one click. You can design split levels and control floor and ceiling heights room by room and view them in 3D. Both 2D and 3D drawings can be printed off in black and white or color on any printer or plotter that is compatible with Windows 3.1.

These and many more features are available in version 3.0 of Chief Architect. I'm going to stick my neck out. I believe version 3.0 will win at least one award if not more for best CAD software of 1995.

Does this software make you self-sufficient? Well, if you have a minimum (I recommend) of a 486 DX2-33Mhz PC with Windows 3.1, you can develop incredible home designs anywhere there is an electrical outlet. Does this software address social needs? A lot of very sharp people are going to be using this software. It gives you that island feeling. I'm writing this to you as I sit by a long, pink sand beach, dangling my feet in 80 degree, gin clear water with a soft Caribbean breeze wafting through my hair (metaphorically speaking) as I sit in my office and gaze at designs of my dream home in paradise on my computer screen.



Dick Harding is President of Compucon Inc., California and is a member of the board of the Silicon Valley Computer Society. He is a dealer for Chief Architect and may be reached at (415) 322-9645.

Want to be a Consultant?

Doug Betts

Reprinted from Fresno PCUG via Phoenix PCUG and PCUG of the Redwoods

A Self Interview

Six years ago, I took a computer consultant out to lunch to pick his brain on how to further my career in the computer field. I imagine in our computer user group, there must be some who are at least curious, if not serious, about working the business full time. There are many different kinds of jobs, even within the range of computer consultant. The following is how an interview with me might unfold.

Q. What do you do?

A. I sell computers, software, networks, and related equipment. I install all these items and teach people how to use them. I perform hardware upgrades, customizations, and repairs.

Q. How many clients do you see in a month?

A. According to my mileage records for May and June, I had 50 to 70 business destinations per month. Some of these destinations were not clients, of course. Some clients had multiple visits during each month. According to an accounts receivable printout at the end of May, I had 22 clients that were billed that month.

Q. How much do you have to know to become a consultant?

A. Not as much as you might think. If you get a job with a consulting company, you can learn a lot on the job, I did. What is more important is how quickly you can pick up on new knowledge. You cannot treat computer consulting as a 40-hour per week job and be successful. You will work to resolve problems after hours and learn new things after hours.

Q. Where do you get your clients from?

A. I have display ads in the three newspapers and the yellow pages of both phone books. I am very involved with the Chamber of Commerce and the Fresno PC Users Group. During my first year, I taught Novell Networking classes at an adult school and offered a number of free seminars. All of these sources have produced enough business to make them worthwhile. I also get new clients as referrals from existing clients and from personal contacts.

Q. What is the most important key to succeeding?

A. Keep the client satisfied through quick response time and standing behind your work and

advice. You need to talk with the client in terms he can understand, without talking down to him.

Q. What is the biggest pitfall?

A. With all the different hardware and software in use and different configuration possibilities, there are many instances where it can take a long time to track down a problem. The key is to enjoy the challenge, use a methodical approach, and don't charge for all the time if it was excessive.

Q. What do you like best?

A. It is fun to have clients think you walk on water when you fix a problem that other people couldn't fix. Recently, a mortgage broker client asked how I could know as much as I do about computers. I turned the question around to her by asking how she could know as much as she does about all the different home loans available and the forms associated with them. I think it is important for computer consultants to remember their clients are experts in their own fields.

ॐ

Glossary of Computer Terms

Reprinted and excerpted from Golden Gate Computer Society via *Redwood Chips*

Users: Collective term for those who stare vacantly at a monitor. Users are divided into three types: novice, intermediate, and expert.

Novice Users: People who are afraid that simply pressing a key might break their computer.

Intermediate Users: People who don't know how to fix their computer after they've just pressed a key that broke it.

Expert Users: People who break other people's computers.

SPAUG MEMBERSHIP

Annual dues \$35



New Member



Renewal

Name

Address

City

State

Zip+4

Business Phone

Residence Phone

Fax Number

E-mail Address

Occupation

Computer System
& Peripherals

Main Software Apps

How did you hear
about SPAUG?

Member List (for member use only)

- ☐ Ok to include my name and ...
☐ Home phone ☐ Office phone
☐ Please do not include my name at all

A Membership Card will be mailed to you.

Please make check payable to SPAUG
and remit at General Meeting or mail to

SPAUG
POB 3738
STANFORD, CA 94309-3738

DISK OF THE MONTH

April 1995

Volunteer Needed

SPAUG Thanks**Renewing Members**Maurice Green
Frank La FetraStanley Hutchings
Larry Weinberg**PLEDGES Dinner**Wednesday, May 10, 6:00 PM, 198 Pine Lane, Los Altos
RSVP Mildred Kohn, 415-949-1833**PRinT SCreen Mailing Party**Monday, May 8, 7:30 PM, 198 Pine Lane, Los Altos
For info, call Mildred Kohn, 415-949-1833**PRinT SCreen Advertising Rates**

Business card	\$ 10
Quarter-page	\$ 25
Half-page	\$ 40
Full-page	\$ 50

★ Free to Members -
Noncommercial, computer-related ads

SPAUG RESOURCE CENTER

Elected OFFICERS

President	Brian Christopher	415-952-5632
Vice President	Cookie Cook	415-282-0474
Treasurer	Beverly Altman	415-329-8252

Volunteer MANAGERS

BBS Sysop	Brian Christopher	415-952-5632
Book Library	Mildred Kohn	415-949-1833
Disk of Month	Volunteer Needed	
Membership	Bev Altman	415-329-8252
POB Pickup	Mildred Kohn	415-949-1833

PRinT SScreen NEWSLETTER

Editor	Nancy Helmy	415-326-1911
Copy Editor	Mildred Kohn	415-949-1833
OCR Production	Don Campbell	415-286-7510
Mailing Party	Mildred Kohn	415-949-1833
Mailing List	Beverly Altman	415-329-8252
Printing	Junglecopy	415-326-7622

Q&A RESOURCE for SOFTWARE and HARDWARE

Accounting	Larry Mehl	415-329-6037
CD-ROM	Charlie Wiener	408-255-1081
dBase/FoxPro	Dick Harding	415-322-9645
Lotus 1-2-3	Larry Mehl	415-329-6037
Paradox	Quinn Wildman	408-335-7892
Quicken	Catherine Haynes	408-973-1808
	Floyd Kessler	415-493-7780
R:Base	Larry Mehl	415-326-6037
Telecommunications	Larry Mehl	415-326-6037
Windows Products	Jan Altman	408-243-5955
	Catherine Haynes	408-973-1808

SPARC, the SPAUG BBS

415-321-4497 14.4bps 8-N-1

GENERAL MEETINGS

Last Wednesday of the month
 Varian Associates, Bldg. 7 Auditorium
 3075 Hansen Way, Palo Alto
 7:30 PM

PRinT SScreen Articles

Deadline

The second Friday following a General Meeting. The newsletter must be at the printer shortly thereafter and the mailing must take place two weeks before the next General Meeting.

Submissions

ASCII format is preferred, although most word processor formats can be converted. If you compose your article in a word processor, save it in ASCII (or MS-DOS Text) for submission.

Submit on a diskette (or in a file via e-mail). Include name, date and daytime voice phone.

Deliver via mail, bulletin board, or hand deliver to the editor at a meeting.

Editor's Swan Song

Editing PRinT SScreen has paid off—I've learned a lot about Word 6 and have had some fun putting together the newsletter. Of course there's been a generous supply of confusion, frustration, and what Jim Seymour characterizes as the "fritter factor"—time-wasting at the computer. Not to worry, the clouds of mystery have lifted to a fair degree. Word 6 seems to be capable of some fairly sophisticated DTP.

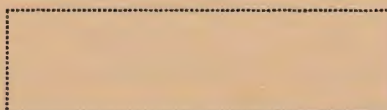
It's time to do a little bragging about the volunteers who have reduced newsletter production and postage costs. Arlan Kertz was responsible for the Government dealings that put us in a position to qualify for bulk mailing rates. Esther Felix scouted out the US Postal Service to find out how to handle bulk mailings. Mildred Kohn, bless her heart, loves to type and edit. Don Campbell offers his system's OCR capabilities. Beverly Altman sorts and prints out labels monthly into zipcode groups. Jim Bailey and his party regulars implemented the mailings on a regular basis, labeling, sacking and delivering to the USPS. Millie Kohn took over Jim's job in February. Vendors have helped out too. Junglecopy copies and assembles the newsletter. And Member Contributors are PRinT SScreen's raison d'être. Thanks to us all! We also owe thanks to our past newsletter editors for their work and for setting a standard.

Nancy

Stanford Palo Alto Users Group for PC
POB 3738
Stanford CA 94309-3738

Nonprofit Org.
US Postage
Paid
Palo Alto, CA
Permit No. 191

Address Correction Requested



MAP
SPAUG GENERAL MEETING
LAST WEDNESDAY OF THE MONTH
7:30 PM

NOT TO SCALE

KN

