

The Analytical Engine

JOURNAL OF THE COMPUTER HISTORY ASSOCIATION OF CALIFORNIA



Scientific Data Systems 930

NOAA / Herb Sauer

The Analytical Engine

JOURNAL OF THE COMPUTER HISTORY ASSOCIATION OF CALIFORNIA

THE X-PROJECT, Part Two

The star of riches is shining upon you.
— latest fortune cookie

Yeah, right. Not quite.

For the X-PROJECT to work, many things must be in place. As I write, one thing is in one place. A large, heavy, Xerox mainframe is in Boulder, Colorado — at the Table Mountain Observatory of the Space Environment Laboratory, United States Government.

So far as we can tell, this is the last known, complete, running, XDS or SDS computer in the world.

It began to be installed and configured at Table Mountain in 1963, and was in full operation by 1965. At that time it was one of the fastest, gutsiest real-time scientific computer systems in the world. SDS' computers were good enough to worry DEC, which was a direct competitor and about to go public; good enough to worry the mighty IBM, which had just bet the company on the System/360. And the 930 was then SDS' newest, biggest and fastest computer. The Space Environment Laboratory put three hundred thousand dollars on the line, and the computer arrived.

Ah, but what they got for all that money. The 930 was supremely agile and versatile, programmed in bare-metal machine language for speed above all. It could take in or send out data while it was performing computations, or running diagnostics. Downtime was minimized with lavish redundancy, multiple power supplies, solid silver connectors, and fat heat sinks. From the first power-up, this was a racehorse. All it wanted to do was run. It could even outrun its own seven-track tape drives, and ended up with a truly giant drum for main storage.

SEL set it to work at one of the most demanding tasks in computing — continuous real-time data

acquisition, no rest for a machine that never wearied. From 1965 to 1970 the 930 served as the main computer for the government's HANDS (High Altitude Nuclear Detection System) early-warning program. Thereafter it acquired data from the GOES series satellites and many other spacecraft. This is a computer justly famous on the strength of its accomplishments alone. But it's also (did we mention?) the last known, complete, running, XDS or SDS computer, in the world.

Nor are we talking cold, hulking racks in a dim warehouse. The CHAC can take over this computer in operating order, with schematics, full docs, bales of parts, and complete software on tape....still warm, so to speak. This chance will never come again; and this computer, built in Santa Monica and dedicated to longer service than was ever foreseen, deserves a proper home in California. It can still belong to scientists and engineers, historians, the American people, and posterity.

Against this rare and lofty virtue, we must set one common and mundane problem. This computer, being a CPU, core racks, main drum, tape drives, console, and the parts and docs as mentioned, fills a room thirty by twelve feet. By ordinary industrial standards, that is not a lot of space, but to the CHAC, it's a gargantuan requirement. Combine that with the expense of moving this device from Table Mountain to Palo Alto, and it compels hard questions.

The Computer History Association of California, for the first time, asks you to use your powers of persuasion on your co-workers, managers, marketing directors, CEO's and companies. This rescue needs serious, corporate money — enough to transport the computer, set it up here, and keep it (at least) safe.

The alternative, naturally, is metal pleated, phenolic crushed, gold and silver stripped, as blind brutal force turns a unique computer into awkward, toxic scrap....

Yeah, right. Not this time.

It's *almost* too late, but it's *not* too late. Please, dig deep! Raise hell! Save one of the few remaining functional mainframes designed and built in California! This will be a rescue you can remember — and an exhibit you can enjoy — for the rest of your life.

PARA-TIME SHIFT

Yes, this issue of the ENGINE says May on the cover. Yes, the previous issue was dated October. And no, you haven't missed an issue.

This is the first issue of the ANALYTICAL ENGINE to be sold on magazine racks in bookstores; and bookstore buyers demand that the magazines they sell bear the date that the issue goes *off* sale, rather than — as has been the ENGINE's practice till now — the date that it goes *on* sale. That accounts for three months of the shift. The fourth month is slippage — but, hey, one missing month isn't bad spread over seven issues.

So, from now on: The issue that arrives in February will be dated May. The issue that arrives in May will be dated August. The issue that arrives in August will be dated November. And the issue that arrives in November will be dated February.

We hope this isn't an inconvenience for our readers. And please wish us luck with the bookstore sales!

NEW E-MAIL ADDRESS:

engine@chac.org

As of February 15th, CHAC will have a new Internet mail address, engine@chac.org. This is a high-speed dial-up PPP connection via WombatNet, supplied by our neighbors, the Wombat Internet Guild of Palo Alto. (Local call! Yaaay!) This gives us far more flexibility — including real-time access to the World Wide Web, WAIS, ftp, archie, gopher, and all that other good stuff — while it helps control operating costs. Please e-mail us at engine@chac.org after Valentine's Day, and at cpu@chac.win.net until then.

We'll take this occasion to thank Michael Tague, Bob Tague, Joe Mays, Connie Rogers, and the other fine people at WinNET Communications (formerly Computer Witchcraft) of Louisville, KY. When we began using their service for dial-up mail and news, in April 1993, desktop Internet connectivity outside the world of UNIX was a truly scarce commodity; we were doubly lucky to find not only an affordable port, but a responsible and industrious provider. From that day to this we've enjoyed refreshingly bug-free client software, bulletproof connections, vanishingly small server downtime, and friendly, consistent telephone support. Who'd ask for more? Without WinNET, CHAC and the ENGINE could never have become what they are today.

ARPANET ARTICLE WANTED

The informative lead, "ARPANET is Twenty-Five," in the recent issue of the Charles Babbage Institute NEWSLETTER (see PUBLICATIONS RECEIVED) reminds us that of the four original ARPAnet nodes, three were in California; one at UCLA, one at UC Santa Barbara, and one at Stanford Research Institute in Palo Alto. (The fourth node was at the University of Utah.)

Clearly the ENGINE needs a commemoration of this important anniversary. Will someone who participated in the configuration of one of these three nodes please contact the CHAC, to discuss submission of an article? Thank you!

IN MEMORIAM: GEORGE STIBITZ

Dr. George Robert Stibitz, pioneer of digital computing and remote job entry, died on January 31 at his home in Hanover, NH, USA. He was 90. At the time of his death, he was professor emeritus of physiology at the medical school of Dartmouth College.

In the fall of 1937, while an engineer at Bell Labs, Dr. Stibitz used surplus relays, tin-can strips, flashlight bulbs and other canonical items to construct his "Model K" (for *Kitchen table*) breadboard digital calculator, which could add two bits and display the result. A replica of this device is now on display at the Smithsonian Institution. Bell Labs recognized a potential solution to the problem of high-speed complex-number calculation, which was holding back contemporary development of wide-area telephone networks. By late 1938 the laboratory had authorized development of a full-scale relay calculator on the Stibitz model; Stibitz and his design team began construction in April 1939. The end product, known as the Complex Number Calculator, first ran on January 8, 1940.

On September 11 of that year, during a meeting of the American Mathematical Society at Dartmouth College, Dr. Stibitz used a Teletype to transmit problems to the Complex Number Calculator and receive the computed results. This is now generally considered the world's first example of remote job entry, a technique that would revolutionize dissemination of information through telephone and computer networks.

From 1941 to 1945, Dr. Stibitz served on the National Defense Research Committee, performing important theoretical work on computation. Thereafter he worked as a private consultant in Burlington, VT, developing a precursor of the electronic digital minicomputer in 1954. In 1964 he joined the Dartmouth faculty and applied computer systems development to a wide variety of topics in biomedicine. He continued his research until 1983.

Dr. Stibitz was born in York, PA, on April 20, 1904. He graduated from Denison University in Granville, OH, and received his M. S. degree from Union College in Schenectady, NY, in 1927. After

working briefly at the General Electric research labs in Schenectady, he continued his graduate studies at Cornell University, completing a Ph. D. in mathematical physics in 1930. He was a prolific inventor with an inquiring mind and held 38 patents, not counting those assigned to Bell Labs. In 1965 he received the Harry Goode Award for lifetime achievement in engineering from AFIPS.

The Computer History Association of California extends condolence to Dr. Stibitz' wife, Dorothea Lamson Stibitz; his daughters, Mary Pacifici and Martha Banerjee; and his brothers, sisters and granddaughter.

IN MEMORIAM: ALLEN COOMBS

Allen W. M. "Doc" Coombs, a supervisor of the United Kingdom's earliest digital computing project, died on January 30 at his home in Yealmpton, Devon, UK.

Dr. Coombs was a principal designer of the Mark II COLOSSUS vacuum-tube digital computer, which entered service "by breakfast-time" on June 1, 1944, after unrelenting and almost superhuman effort by Coombs and his engineering staff. Mark II COLOSSUS was the world's first computer to enter series production and was, of course, qualitatively important to the Allied victory in the Second World War. After the first Mark II computer had entered service, Dr. Coombs took over production management for the rest of the series, replacing "Tommy" Flowers, who moved on to other projects.

Dr. Coombs received his B. Sc. degree in 1932 and his Ph. D. in 1936, both from Glasgow University. Much of his engineering work thereafter was subject to the Official Secrets Act, but some details of his career may be found in his 1983 article, "The Making of COLOSSUS," in *Annals of the History of Computing*, vol. 5, no. 3.

The Computer History Association of California extends condolence to Dr. Coombs' wife, Vera.

RESOURCES WANTED FOR WEB PAGE

In barely five years, the World Wide Web has grown from a quirky and daring experiment at a single laboratory — CERN in Geneva — to one of the world's broadest and most diverse repositories of on-line text and graphics. Early experiences with the Web were precarious at best, as the client software used to access it was much less reliable than the matrix itself. The recent development of Netscape™ and enhancement of NCSA Mosaic™ has solved many problems; now almost any 32-bit desktop computer — whether it runs X-Windows, MS-Windows, or a Macintosh OS — can become a Web terminal.

Many institutions with a strong interest in the history of computing, including the Association for Computing Machinery, Smithsonian Air and Space Museum, Cambridge University, the University of Limerick, and Uppsala University, have written and installed Web pages with links to a rich variety of appropriate resources. Sometime soon, the CHAC will put up our own Web home page, with help from our Internet provider and a volunteer writer of HTML.

Personally, we find the Web in general to be one of the nicest neighborhoods in cyberspace; and we want the CHAC's participation in it to exhibit the same style and solidity that have brought honors to the ENGINE. But — like the ENGINE — our forthcoming Web page will need your help to be what it should be.

Over the next few months, we'll investigate *every* Web site, ftp site, or other net.nexus that we think might be valued by students of computer history....obviously in California, but around the world as well. The resulting comprehensive page will be as much of an asset to the Web as we can make it. The Web and its browsers, in turn, will be the ideal vehicle to bring it to an audience of millions.

Please: Be one of the few who builds for the many. If you — or your company, professional society, or academic institution — sponsors an Internet resource on the history of computing, send us e-mail with a pointer to it. We'd like to link you in.

ALERT: CALCULATOR AND INSTRUMENT SHOW

If you're a computer collector, it's only natural to feel a certain affection for calculators. They're useful, technically intriguing, easy to store, and nearly impossible to kill. Their popularity is so well-founded that, in the twenty-two years since the stunning advent of the HP 35, they've become ubiquitous. On the other hand, certainly they're no less fascinating simply because everybody has one.

On Saturday, May 20, 1995, the calculator will enjoy a long-overdue tribute at the Calculator and Antique Scientific Instrument Convention, Show and Swap Meet, to be held at the Arlington Convention Center, Arlington, TX, USA. Show hours are 10 am to 4 pm and admission is free. The event is organized by Skip Solberg with the assistance of our good friends, the International Association of Calculator Collectors.

Early calculators and related advertising and ephemera will be displayed and, although some plans are still preliminary, notable calculator pioneers and manufacturers have been invited to attend. The nearby Arlington Marriott offers inexpensive Super Saver Weekend accommodation; if you're traveling to reach this event, the Convention Center is only 10 to 15 minutes from Dallas-Fort Worth Airport.

Now, here's *our* pitch: The CHAC has been invited to participate in this event, and will send a representative if at all possible. It will be qualitative if our generous readers (i. e. you) can donate frequent-flyer miles, San Francisco-Dallas round trip coach air, Saturday night for one at the Marriott, or just plain cash. This won't be as elaborate as our appearance at Pomona a year ago, but since we'd *like* to bring hardware to display, it will be a production in its own way. Please help!

DETAILS: Exhibitor tables are still available at \$20 for 20 square feet, from Skip Solberg, 717 Salsbury, Arlington TX 76014; you can call Skip at +1 817 467-0368 after 6 pm CST. To reserve at the Marriott, call +1 800 442-7275.

HP's EARLY COMPUTERS

[The Hewlett-Packard Corporation's work with calculators, instrumentation computers, general-purpose computers and workstations has created one of the longest — and most fascinating — histories in California's computer industry.

The ANALYTICAL ENGINE now addresses that history with a projected three-part series. Part One, concerning the Model 9100 desktop "calculator" and the 2116A Instrumentation Computer, follows. Part Two will consider the 20xx and 21xx general-purpose computers, built at the Cupertino factories. Part Three will be an article on the earliest years of the 3000 Series.]

Part One: A CORE PLANE IN AMBER

An Interview with Barney Oliver

[Dr. Bernard M. (Barney) Oliver currently serves HP as Technical Adviser to the President. He joined the company as Director of Research in 1952, held the post of Vice President for Research and Development from 1957 to 1981, and served on the company's Board of Directors from 1973 to 1981. His work at HP, and at Bell Labs previously, has substantially advanced the physical and engineering sciences, particularly the fields of electronic instrumentation, automatic control, and radio physics. He has written over 40 technical articles, holds 52 U. S. patents, and has held innumerable positions on scientific and technical advisory bodies.

On Monday, December 19, 1994, Barney Oliver and Kip Crosby met in Barney's office at HP's Palo Alto research lab for an extended conversation about the company's entry into the market for special-purpose computing devices.]

THE UNDERLYING MATH

KC: Producing a computer was a logical extension of HP's long history in instrumentation, but how and why were the decisions made?

Oliver: The first phase of our getting into computing, which you may be aware of, preceded our early history in the digital calculator field. They were distinct efforts, but both started here in this

lab. We had been looking at the whole question of calculation and had seen, from Friden and other manufacturers, examples of electronic calculators; and we realized that there was a future in that field, which had not been completely exploited, to say the least. The Friden was essentially a mechanical calculator with some electronics substituted for mechanical components. It was obvious that you could do a hell of a lot more once you entered the electronic domain. We were just trying to get our ideas together on that subject when two people visited us independently. One was Malcolm McWhorter from Los Angeles. He and another guy had developed a calculator which would perform transcendental operations, transcendental functions, and he brought this big kluge with him. It was a box about the size of two beehives. They finally got it working and computed a tangent and other trig functions for us. It took over a second to do this.

We were interested not so much in the machine, which was out of date by the time it was built, but in the algorithm they used to do it, which was called a "cordic algorithm." What the cordic algorithm does is look at the generation of functions geometrically. For example, say you want to compute a tangent. You simply set in an angle, the one you want the tangent of, and then you rotate that vector up to that angle. In this way, you can find the sine and cosine, each times a constant. There's a "secanting" error that comes in, you just ignore that. It gives the sine and cosine times the same constant, so then you take the ratio and you've got the tangent. There's the same secanting error in both, and it cancels out.

KC: Brilliant.

Oliver: It's a simple algorithm. You have three registers, two for x and y, let's say; I start out with 1 in the x register and 0 in the y register, and the angle theta in the third register. Since we're doing binary-coded-decimal [arithmetic] now — I subtract from theta an angle whose tangent is one-tenth, which we have as a stored constant in the machine. Then we take one-tenth of x and subtract it from y, and one-tenth of y and add it to x. This is the coordinate transformation. $x' = x \cos \theta - y \sin \theta$, and $y' = y \cos \theta + x \sin \theta$, is the transformational equation as we rotate a vector through that angle. So we rotate it through an arctangent of a tenth. That isn't enough. We now go to two-tenths.

That's too much. Now we start backing up by a hundredth, by the angles whose tangent is a hundredth — through the ordinary division routine that we already have stored, because many other algorithms could be married, if you will, to the cordic algorithm. So you zero in on that angle, and after you've got down to the arctangent of a thousandth, or something like that, the tangent is essentially equal to the angle; so you can add the remaining angle directly, so it's nothing but a cross-add-and-shift, and it's very adaptable to machine calculations.

We also realized that having computed the tangent, you now have the sine and cosine anyway, from simple trig relations. So with nothing more than simple algebra and ordinary arithmetic, you can compute all the trig functions. Then we realized that if we simply didn't change the sign, but cross-added both arctangents in the algorithm, we'd be computing the hyperbolic functions, and so we said, oh my God, we've got a simple algorithm here that will do all the transcendentals, and that kind of excited us.

KC: I can imagine!

Oliver: And exponentials, if you don't shift across to another register, you just multiply and add to the same register, and that's what you do when you compound interest, because it grows exponentially, doesn't it. All right. That's what we were doing in the machine — all of the exponential, hyperbolic and circular functions with a single algorithm, which naturally produced an internal economy. It was very appealing to us.

OSBORN AND THE 9100

So we had the math, but as far as the physical design was concerned, we still hadn't quite decided what to do when the second guy appeared. The second guy was Thomas Osborn, and he had an interesting career even then; he had been an EE student at Berkeley majoring in Computer Science, then graduated and got a job with Smith-Corona-Marchant, who were bringing out a small calculator at the time. He took one look at what they were doing and said, "Oh, that's awful. I can't work on that. I just don't think it's going to fly." They didn't believe him, so after a few months there, he said, "I think the honorable thing to do is to make you an offer. I will resign. I will work on

my ideas on my own time and build a model. I'll bring it around to you, and if you like it, we can talk about price." In other words, I'll freelance this thing for you and try to sell it to you.

Well, they went ahead and still worked on their original design, which they called the Cogito. It was a miserable machine, it took forever to do anything. Tom meanwhile abandoned [then] normal computer practice and went immediately to floating point, and all of his machine worked in floating point. When he came to visit us he had been turned down by SCM, by IBM, and by about twenty other companies who just weren't interested. He dropped in to see us, we looked at it, and we saw a great vehicle to combine with the cordic algorithm and with some other ideas we had about magnetic recording. The result was a programmable, stored-program machine, and we said, "We'll make the whole damn thing, let's go."

Tom was very enthusiastic about that, but he did not want to become an employee; he had kind of a free spirit about him. We said, "Well, it doesn't matter to us. We'll make it a satisfactory arrangement," and we did, and he got quite a bit out of this thing. It was a very interesting machine and very unconventional.

KC: This was the Model 9100.

Oliver: Correct. It was a desktop machine, that would display the x and y and z registers; it had a keyboard that had all the transcendental functions available on one keystroke, and it computed in milliseconds. It was a very fast machine because the 9100, which we introduced in 1968, had in modern-day terms a 64-bit-wide word.

KC: Oh, really!

Oliver: Yes, because IC ROMs were just beginning to appear at that time, and we didn't trust them for our application, because their size was too small, and their reliability wasn't established. So we went ahead with the first, and probably the last, electromagnetic ROM; it was a printed circuit board that had 64 sense lines — little slim hair pins that went across the board and connected to little amplifiers at the open end, so that 64 of these cross loops became sense lines. Down through those came the drive lines, which jogged to the left wherever we wanted a zero and jogged to the right where we wanted a one; and that gave us an up pulse or a down pulse, appropriately. The pattern

was embedded in a PC board, really built into it, and not reprogrammable. But we had a need for completely reliable ROM....

KC: Which amounted to fixed-content core memory on a printed circuit board.

Oliver: Exactly, and nonvolatile.

KC: Totally nonvolatile! Some amateur historians of HP are adamant that the 9100 was in fact HP's first computer, because [besides this] it had a real [magnetic] core plane.

Oliver: Yes, it did. The user memory on this was several registers; first of all there were three display registers. I think there were two computing registers and there were about six storage registers in addition. I think there were 16 registers total in the machine, and the numbers being handled in there were 64-bit numbers, so they were stored as 64 bits in the mag memory.

KC: 16 four-bit bytes.

Oliver: Yeah, I think that's right. I know there were six planes. I'm not sure of the details.

KC: Don't worry, we have full docs on the 9100. Which, of course, are several times the size of the machine itself.

Oliver:(laughs) Very good! There were something like 2200 bits — yeah, 2208 bits of core memory. About 32,000 bits of ROM. But the whole thing was that when you come to a state in the machine — this is an algorithmic state machine, now — depending on the variables around, you make a decision. From the nature of the program you know what the decision is going to be, so you ask the question, and bing, you hit one of these drive lines with it, and immediately 64 bits came out, and they said this operation must be performed, and the results of it go in this, and qualifier so-and-so must be set to this to do this, and you pump out a hell of a lot of information as to what was coming next. And so in a sense it was a pretty wide word. We could handle the 64 bit number, floating point number all at a crack, one fetch and an add. Frankly, at the time, I didn't fully appreciate the extent of the innovation. I knew the machine worked well. I knew it was fast.

KC: It worked well, it was fast, it represented about as much computing power as you could then currently put on a desktop, and furthermore sold for \$7,000. All that made it an extremely popular machine.

Oliver: Actually, for \$7,000 you could have one with the plotter.

KC: Ours has a 9125 plotter. In fact, people have been donating programs for the 9100 that were written, by them or by others, at universities and corporations. It's a highly regarded machine to this day.

Oliver: Among?

KC: Well, just as one example, we have a member who is a computer science professor at the University of Iowa, Dr. Douglas Jones, probably one of the most active computer historians in the Midwest. As soon as he heard we had a 9100, he sent me a program — that he had written personally years ago — to make a 9100 with the plotter into a Spirograph. Other people have donated programs. You say "9100" and this whole subculture comes out of the woodwork.

Oliver: That's exactly right, because it was such an independent start that we didn't rely on anybody else's formats or prior art. We just came out with it — bang! — and it didn't resemble any machine that someone might have used before. But people who became adept at using it were likely to be very devoted to it, and there were enough such people that the 9100 became a considerable success. And that was in spite of some aspects that were far from perfect; for example, my God, it was shy on memory! And so the 9100B came out with twice the memory. By then the whole project had been sent to Loveland, they sent their people here, and we overlapped for about six months, transferring it. Loveland took it over for manufacture, and did an excellent job at that.

GENIUS FORGONE

Meanwhile they had been assigned the calculator business, so they decided to do the next machines, which were the 9810 and 9830, and which had a much more conventional architecture something like the 2116. None of the stuff we had in the 9100 survived. And to my great shock, they were about one-quarter the speed, also. So — this is my version of what happened — the Loveland group went merrily ahead with the conventional thing. It was agonizingly slow, actually, if you'd had a 9100. And of course we chided them on that, we made

their lives miserable. So they buckled down and put in a lot of speed-ups that that they hadn't had before, and the components themselves were getting faster, so the later versions of the calculator gradually sped up until the fastest one — I have one over here, it's a 9825 — was a nice little machine. Finally.

THE DAWN OF HPL

Another virtue of it was — what I think was — a damned good language on it, which we called HPL. In outline it was very much like a BASIC language, except that it made much more sense. For example, in a BASIC language, you say $x = x + 1$, and that's a goddamn lie. x never equals $x + 1$. What I say is $x + 1$, that number, goes into the x register. And the symbol for insertion into the register was the assignment arrow, which was on the keyboard. That made it a key-per-function machine, as a matter of fact Keeper was its code name. And to go to this higher level language, what we did was compile. The basic machine, in the tradition of our calculators, worked in reverse Polish, and we knew that that was a good language to store programs in, because of its economy. You can't do any better.

So we said, why don't we make a machine — I worked with a whole bunch of other people on this — a machine that basically operates in reverse Polish, but has a compiler built into it; so that the user would enter data into the machine using an algebraic language, come to the end of each line, hit <Enter>, the compiler would compile each line and put it in reverse Polish. When you wanted to look at it, it blinked a little bit. It had gone through the cycle of encoding and decoding, and now displayed what it thought you said. The neat feature was that the compilation from the algebraic into reverse Polish, what it did when you said <Enter> at the end of the line, was reversible, which gave immediate editing. If you didn't like what you'd said, you just went to another mode and corrected it line by line. The 9825 was a nice machine which I still use occasionally, even though I have much more resourceful devices at my command. I love to program it and have it do things, and that's the functionality I absolutely miss in the modern PC, which has all the stuff done for you. Things like Windows and so on are

all very clever, but there's more in them than I'm ever going to use, and I have to buy it all.

KC: Well, the gentleman who gave us the 9100 must have felt somewhat the same way. He had a small engineering firm in South San Francisco. When he gave us that 9100, it was in immaculate condition — I mean not in any way dusty, dilapidated, neglected, whatever; and he said to me very significantly, "Young man, I want you to know that I'm giving you this hardware because I have reluctantly concluded that I will never need it again." This was in 1993, and he had been using that 9100 for 25 years.

Oliver: That's easy for me to believe, because I have a couple of 9100s myself, just for old times' sake. If you're starting a museum or somebody needs one, let me know, because I won't use them again either.

KC: Thank you. We find, so far as the museum goes, that HP equipment is some of the easiest to lay hands on — because no one ever disposes of it. The biggest computer in our current collection is a 3000/42 that was given to us by a company in Santa Clara. When people finally have to replace HP equipment, they don't put it in dumpsters. They look for somewhere for it to go.

BIGGER IRON: DSI AND DYMEC

Oliver: I suppose speaking of the 3000 brings us around to the conventional computers.

KC: Perfect. I was curious about HP's original reason for introducing the 2116A, which is sometimes also called the Instrumentation Computer.

Oliver: It became evident, I would say, in the early 1960s that all computers didn't have to be huge devices — that we could in fact do a sizable amount of computing with a much smaller computer; the amount necessary, for example, to steer the instruments in an automatic measuring system. So we set out, in the labs, to make a controller for our measuring instruments. Our grand strategy was to make all of our instruments talk in a single language that we developed, which would be a language common to the computer and to the instruments, and which would let the computer handle the whole situation.

There was a lot to be gained by that. Much of our experience connecting automatic measuring systems came from our subsidiary, Dymec, which

took standard HP instruments and put them together in special configurations. As they proceeded, they developed a lot of the interfacing that we needed to begin with, so we were already down that road when we decided to make our own chain....

KC: Dymec is a matter of some curiosity to computer historians. The question flies around the Net occasionally: "What was that company whose logo was exactly like HP's, only upside down?"

Oliver: That was Dymec, and there's no mystery about it. It was a company formed by contributions from HP executives, who held most of the stock, and its initial mission was to provide some of our fundamental parts — things like transformers. At some point there was a change in direction, and Dymec became, if you will, our pathfinders in the automatic measuring field. That experience convinced us to make all of our instruments programmable, that is to say, responsive to particular codes encompassed by a control language. That became HPL, and the computer to run the show became the 2116.

2116, to the best of my knowledge, started in HP Labs — in this building we're sitting in, 1501 Page Mill [in Palo Alto], and it was begun principally by a man named Kay Magleby. Kay Magleby is not with the company at the present time. He decided to go out on his own, and he's teamed up with John Atalla, who was originally one of our lab leaders, and they've introduced some products and lately have more in development. I just talked with them the other day. So we're on good terms.

About the same time, Packard began to get a little uneasy. We were not keeping up with progress in this automation field, and so he decided that if anything came along that seemed reasonable as a nucleus for a group, he would acquire it if possible. When the Union Carbide group became available, we snapped it up and used the personnel from that design group, along with the people from our lab here who had already given some thought to the problems, to form the initial group for the 2116.

KC: That group from Union Carbide was called DSI, and to my understanding, it was acquired more or less intact. But I don't quite understand why Union Carbide had a computer design group to begin with.

Oliver: We didn't demand an answer to that question — as to what their motives were. It suf-

ficed that it was a ready-made design group with some pretty reasonably talented people in it, and we decided that that would be the nucleus.

Anyway, we no sooner got the 2116 mapped out here in the labs as regarded the size and architecture of the machine, than we merged with Union Carbide. Then the job of our people was to carry the design into [DSI's] hands, let them suggest advantageous modifications, and to get that into the marketplace as fast as we reasonably could.

KC: Which involved rapid development of a common interface to products that were already on the market.

Oliver: Right — which we were busily doing in all our divisions. We had our own meetings on that matter, as did a lot of other groups. The New Jersey Division — I can't recall the name of the town offhand — had developed what they called a multiprogrammer; a thing that would handle a number of instruments from a common drive, so as to distribute the commands and receive inputs back. We called it a multiprogrammer for lack of a better name. We were gradually assembling all the things to make automatic measuring systems.

CUPERTINO

Meanwhile, at the 2116 — the computer — end, we combined DSI, Dymec and our own instrumentation people into a computer division. Just at that time, the Varian building in Cupertino was available for sale, so we bought it and staffed it with that nucleus to create the Cupertino Division. Now, after a couple of years of operation, analysis of their sales disclosed that they were selling many more computers as freestanding units than as computer control of automatic measuring systems. So we decided that this was new business to go after.

KC: Part of that must have been traditional HP ruggedness. Shirley Gilbert tells the story of a 2116 installed in a station wagon for people to drive around to on-site demos. At one point, somebody cracked up the station wagon. The car was totaled, the driver went to the hospital, and all the repair that the 2116 needed was accomplished in about 10 minutes.

Oliver: That's a good story. I hadn't heard that one. Anyway, that's what happened — was that having decided to make a lot of automatic measuring systems, we found we were selling even more computers on the side than we were for people's measuring systems. And the reason was not hard to

come to. An automatic measuring system has a lot of advantages, as you might imagine — because it not only does things quickly, but it can do them much more accurately; because it can measure a known impedance, for example, and then whatever prevalent error exists in the system can be recorded at each frequency, so it makes its own calibration curve before turning to the first piece of equipment to be measured. In fact there are so many advantages that, for microwave measurements, I think we typically picked up about 30 dB of accuracy by the automatic measuring system.

KC: Really!

Oliver: Yes, because, for example in an impedance situation, we measure first with a known short, a known open and a matched load, and then use these readings to correct those of the unknown.

KC: You'd have it just like on a railroad track.

Oliver: Right. All errors from all the devices in there have now been taken into account and you're looking right at the thing itself, so to speak, and that's very nice. But we tried to promote that and people would always say, "Well, we'd rather buy the instruments and put the system together ourselves." Okay, we said, and we sold them the instruments and the computer and let them put them together. As a matter of fact, that's what we still do with our help and guidance department here, but nobody would believe — until afterwards — the cost of the engineering time it takes to really do that.

KC: Oh, that's still true.

Oliver: That's still true. But the point is, you see you're doing one of a kind. The moment you're doing an automatic measuring system on your own, you will buy one. You're doing something that's one of a kind at that point, by definition, almost, and you haven't got a big base to amortize all that engineering time over. We knew that, and we were finding that, and we were charging for it, and they couldn't believe our bill.

KC: Right!

Oliver: But they went ahead and spent the money and —

KC: And had no one to blame but themselves.

Oliver: And had no one to blame but themselves. Meanwhile, back at the farm, we were taking a good look at the computer itself —

KC: I understand there was some effort to compare the 2116 to other machines that would appeal to the same market, and one of our members has suggested that HP once considered OEMing a version of the PDP-8.

Oliver: I don't recall that. The only way to verify that would probably be to go to Dave [Packard] and ask him directly. But certainly by the time of the 2116, when we began to realize we were in the general-purpose computer business, we looked upon DEC as direct competitors. As a matter of fact, I think that even before the period we're discussing here, Packard was considering buying DEC, and I guess Ken Olsen wouldn't say yes, or they couldn't agree on a figure or whatever. Anyway, it never happened but I'm told that was the case.

To get back to Cupertino, I think that in our innocence — in the late sixties — we made a series of management mistakes that very much hampered that division. For example, they came at us at one review with the proposal that they build a computer called the Omega Machine, and so we said, "Well, what is it?" And it turned out to be a 32-bit computer which they were very enthusiastic about. But because their performance had been poor to that point, and their profits were down, this would take much more R&D budget than they had earned, which made the accounting side of the house frown on it. A hell of a lot of people left Cupertino because they saw an opportunity in that to make a real contribution, and HP turned them off. This was a company that had already had the daring, if you will, to build a 64-bit calculator.

KC: Now this Omega Machine would have been a 32-bit machine with what generation of technology? Was it TTL?

Oliver: I believe so, but don't take that as gospel. I don't know.

KC: Was it intended to be a mainframe in the proper sense, or a semi-portable machine like the 2116?

Oliver: It was supposed to be a mainframe in reality and not to be advertised as such. In other words, we conceived of a very fast machine that could perform a lot of traditionally "mainframe" functions, but that we could sell without immedi-

ately getting IBM on our tails. We had a way of acting humble in those days because we felt — rightly or wrongly — that IBM, the computer giant, could become annoyed at us at any point and simply squash our computer business, by overwhelming us with a tenth of their talent. After all, they'd done a hell of a lot to hold onto their market share, especially when they were confronted with competition. So we spent a lot of time trying to keep a low profile and nevertheless make contributions.

Well, we made some bad mistakes here and there, but I think the thing that finally cracked us loose were the Snake [9000/700] workstations that came about just a few years ago, which were the first really high-horsepower machines. We had the PA-RISC principle, but until those machines came along, we hadn't fully exploited it; we had grafted it onto things, but that didn't let it do as much as if we had started with the concept in the beginning. In the Snake machines we did that as a from-the-ground-up design, and it was developed without giving management a lot of knowledge about it. It was one of those under-the-counter things, that turned out to be a saving grace for HP in the long run, because it really has been very successful. All the 9000 stuff has been very good, you know, it was the first stuff we ever put out that got Sun Microsystems worried.

KC: But there was another reason for some hesitation as regarded computer development. A part of management was very dedicated to continued computer development; another part of management saw that much more could be done with calculators; and a third part was committed to refining and upgrading instrumentation, so that you were almost involving three companies in a philosophical sense. That may have created a reluctance to put more than a certain number of eggs in any one basket.

Oliver: I'm not as conscious of that as I was of our trying to do things.... That leads to an issue which I'll try to illustrate. It has been traditional, in development work at HP, that we try to make a contribution in every instrument we bring out. We're not content to put a new face on something; we really want it to perform better in the sense of advancing some specs by significant amounts, or by making a good product more cheaply, whatever, but there must be a contribution. And so when we got into the PC market, for example, we wanted to

make a contribution — we used the paradigm of pushing the spec, but in that market it was less appropriate. There were all kinds of things on those machines that the customer didn't understand, didn't know about, or didn't use, which therefore just sat there and were wasted. Finally we tumbled to the realization that in a PC, what you wanted is not contribution in that sense but compatibility, and the contribution is going to come about through more efficient internal design, or enhancements to the operating system or something like that, or maybe it doesn't have anything, just reliability, and a good price, and then you're in better shape to compute because the software is coming out of Microsoft and everybody else, so we have done better with that philosophy. But we had to have good engines first.

KC: Vectras have always been wicked fast.

CHALLENGES OF THE 2100

We were discussing this attempt to keep yourself under IBM's radar, so to speak, just as you were developing the 2000, 2100 series, and my understanding is that those were presented primarily to the educational market.

Oliver: We showed those to a lot of people. I think they were used a lot in schools perhaps, but we marketed them for the educational field, not so much because we thought they were the machines for education, but because we wanted a greater presence in the minds of the students. We succeeded in getting some of that, but in the larger sense it didn't pay off immediately. HP hasn't really been looked to as a leader in the field until the last few years, till we got some horsepower inside our machines and were designing with a little more savvy.

KC: To my mind, the educational market brought two new challenges. One was that, at that time, HP had not built any consumer or commodity products; so a university student who was working with, say, a 2100, would be encountering the name Hewlett-Packard for the first time, and you wanted that encounter to be pleasantly memorable. The other thing, of course, was that in an academic environment, where you might have people working in shifts to literally put 24 hours of computer time into any given day, reliability was of primary importance.

Oliver: Designing reliability into the 2100 meant ironing out an interesting glitch. They got it all ready to go, and Cupertino had farmed out the design for the power supply in the machine, and they had been told they should allow a half cubic foot for it. And when the models came, they didn't work; not only did they not regulate fully, but they also didn't hack it. They burned up. And so all of a sudden Cupertino was going to make a 2100 with a deep chassis that would be a hell of a big computer instead of a nice little machine.

So they called up HP Labs to say please pull them out of the fire, and we did. We made a very interesting kind of switching power supply. We converted the incoming AC to a high voltage DC; then we took that DC and just square-waved it, and with that high frequency — 400 Hz or whatever it was — we designed a transformer that would come down from that high voltage to all the other voltages required, and the regulating means was the duty cycle of the square wave. In other words, you control that to control the fundamental component that the transformer handled.

KC: Orderly stuff in, orderly stuff out.

Oliver: And we also found that we had so few turns in this transformer that the voltage steps were too high for a single turn, and that led me to jump into the act and invent a transformer that you could get half-turns out of. Want to know how that works?

KC: Sure!

Oliver: Okay. Imagine a normal core with a center leg and two outside legs. Now you put a winding on this thing, which magnetizes the center leg, and the flux returns through these two side paths. There's nothing to cause those two side paths to conduct equal flux. All you've got equal is reluctance, but it isn't very stiff. In other words, if you load down one window by putting a turn around one leg, rather than around the center leg, it would be very soft, very sloppy, and if you draw any current from it, the voltage would sag — because that flux in that window would go down and the flux in the other window would go up. So what I did was put a figure-8 strap between the two windows, like this, to force the flux to be equal. If it wasn't equal, that was a short-circuited turn for the imbalance, and the current flew like hell around that strap to cause the flux to be equal. So I

could now count on one-half of the flux in one window and one-half the flux in the other window —

KC: It was a self-regulating transformer.

Oliver: Well, it wasn't exactly self-regulating. It's just that you forced equality between the two paths of the magnetic circuit by having a turn that was a shorted turn to any inequality.

KC: Okay, got it.

Oliver: So the figure-8 around the top, if you trace it out, goes into two turns for any imbalance. And if you get any flux, that represents an imbalance in the windows, you'll cause a hell of a lot of current in that shorted turn, and it won't let you do it.

KC: And it worked.

Oliver: So I got a patent out of that one, and that enabled us to get a very nice power supply in a half cubic foot.

KC: And that let the 2100 stay the appealing size that it was.

Oliver: Right, exactly so.

KC: One of the computers that's waiting for us in a parking orbit, until we get more space, is a 2100 MX that somebody wants to donate, and it's a nice little machine.

Oliver: There's been a lot of fun in all the development, but much of it was long enough ago that my memory is a little hazy.

QUESTIONS OF INTELLIGENCE

I'd like to be able to tell you more, but we've discussed pretty much all I can be sure of. Are you familiar with Pentti Kanerva's work?

KC: Afraid not.

Oliver: He wrote a book called *Sparse Distributed Memory* that you can buy, or you could buy, at the Stanford Bookstore. And he proposes a memory system that shares so many of the properties of the biological system, it's absolutely uncanny. I'd love to get this company working on this, but I can't get anybody to care a nickel about it.

KC: This was a semiconductor-based memory?

Oliver: Well, it doesn't matter what the medium is. It's the organization that counts. What are some of

the characteristics of our memory? What does our memory do when you ask somebody to recall something? It's a little bit inaccurate.

KC: It uses a lot of fuzzy logic.

Oliver: A lot of fuzziness in it, and it seems to get full as you get older.

KC: Right, but part of that is a tremendous amount of redundancy.

Oliver: That's really what you get down to in this model. What Kanerva says is, suppose you define an experience as a thousand-bit number, that you have maybe ten experiences a second. You want to get a consciousness movie, so to speak, you'd have that bit rate associated with it. How would you store a thousand-bit number? First of all, you want it to be content-addressable. You don't want to have a separate address, since we don't have one in our head.

KC: I'd never thought of that. That is a little scary either way.

Oliver: But that's the case.

KC: Because what we're saying goes off into all kinds of things, like there isn't segmented memory or there is, or there is or isn't an offset —

Oliver: What he proposed was that if you said, well, I'll make it content-addressable, you cannot have 2^{1000} locations, can you?

KC: Not unless you have really fast circuitry.

Oliver: Not unless you have a lot of things, my friend, that's just a hell of a number. I mean, 2^{10} is a thousand, a million would be 20, a billion would be 30, quadrillion 40, so on. 2^{1000} is a hell of a number. And in our brains we have, maybe, 2^{30} locations. So what do you do? And he said, "Well, one thing you could do is to have an urn full of marbles that are marked with numbers from zero to 2^{1000} and all stirred up, and you reach in and pick a number out at random and record that number, and then you agree to store all numbers that are within a certain Hamming distance of that number at that location." The Hamming distance is the number of bit disagreements. And so it turns out that you will find yourself storing a given experience, not at just one location, but at any location in the whole brain, in the whole memory, that is within a certain convergence sphere of it, and so you want that number of locations that you have

to have to be about 2^{30} ; and then it works out that you store that in all these locations, and — let's say there are something like a thousand locations that you store per event, and they are all within this Hamming distance of the number.

Well, now how do you read them out? You go to all of them. How do you store the numbers in the first place? If the bit to be stored is zero, you decrement a counter, if a one, you increment a counter, at that bit location all the way through. So now you go to all these locations. You then come back to this number and say "Where can I find it?", and you do a majority count on all these locations; and because the number in question is added amplitude-wise and the noise is added power-wise, so to speak, you can have thousands of other things stored in these same locations with only very slight degradation of the signal-to-noise ratio. In other words, if I have a thousand other events and my expected noise at any one location is about 30 bits, from a thousand locations, some will be up, some will be down, in a random walk, and so I would have very strong signal-to-noise ratio even though there are thousands of other things stored. I recall this one thing because it's the dominant thing that's adding coherently at all the locations. And that begins to be a little creepy. It feels kind of nice, you know. And then you say, "Well, how did I get to this location?", and the answer is, you don't. Instead of storing, you go to a location, and you store the address of the next location. Now you have a linked list. Enter it at any one point and then you go zinging along those lists and you have the whole motion picture replayed.

KC: In spite of the fact that, along this chain of physical addresses, you probably have several other movies—

Oliver: Hundreds of thousands of interacting ones.

KC: Which are just determined by the additive amplitude, if you will, of the signal down the line.

Oliver: It's a very powerful concept. And it so simulates many aspects of biological memory, because you cannot localize memory to any great degree in the brain. You can't say "If I take out this cell, then I will remove a specific memory." You do damage the brain a little bit, but the memory is retained, or it seems to be.

KC: What this says, among other things, is that memory is substantially holographic in nature; anytime

you destroy one copy of the memory, you have only increased the fuzziness of the memory as a whole, because it's supported by fewer copies. But now the big question that arises here is, what then determines priority of memory? Why do you remember event A and not event B?

Oliver: You're asking some questions that everyone asks, and so you say to Pentti, "What evidence do you have that this is the way things actually are?", and he replies, "I haven't really strong evidence, but I decided to try to make such a memory using known types of nerve cells, reacting with known reactions that nerve cells can have." And he ends up with a structure which is precisely the structure of the cerebellum.

KC: Oh, okay. That you would have to call experimental evidence.

Oliver: It certainly is very nice, if that's the case. It doesn't prove anything — yet — but the structure is there that would do this. And the cerebellum, my friend, is the most primitive part of the brain; it's entirely concerned with coordination and reflexes and things like that, and tied in with the reptilian complex of the brain. You see, the principal contract between the brain and the individual is an agreement to survive. Anyway, I think you'd enjoy reading this book, called *Sparse Distributed Memory*. It's distributed, for obvious reasons — you store things in a number of places. Sparse, because you don't store them in anywhere near all those places. The sparseness of it permits the pieces to intersperse, or intersperse, with other comparably structured memory.

KC: And at least to my mind — having just been introduced to these concepts — after a while, it starts not looking very much like digital memory.

Oliver: It's different in many respects. You'll know what I mean when you read this. But I've been working on projects involving memory and intelligence — the distribution of intelligence — ever since my retirement.

THE UNIVERSAL SEARCH

KC: What are you working on currently?

Oliver: I'm down at the SETI Institute. I'm trying to make sure that thing flies, because I think that would be one of the greatest contributions of all time, to establish contact between independently

intelligent species across light-years of space. The current era can be compared to the fifteenth and sixteenth centuries, which were pretty exciting times because of the discovery of the New World.

KC: The realization that there were other civilizations.

Oliver: Which had long been suspected, and which Columbus found to be the case. He thought he was in the Indies — which were a locality known to the Europeans — but instead he found something entirely distinct, the Americas. And the excitement of that discovery completely reversed the comparative stagnation of Europe.

I think that this search, if it can be accomplished, would be as great and as positive a change. In the first place, if you contact one extraterrestrial civilization, you probably will contact a network rather than one, because that civilization may well be ahead of us, whether in years, in experience, or in technological aptitude. At that point we find ourselves a member of a community of intelligent cultures, which would mean that the whole natural history of the galaxy might be at our disposal. We could, for example, find out whether DNA is the chemical of life everywhere or whether there are different forms —

KC: Something based on silicon?

Oliver: Well, the silicon-based life is going to be the one we fabricate, I think.

KC: That's true, too.

Oliver: But not the way we're going. I think there is so much difference between the brain and the computer. Their similarities are dwarfed by their differences. We're just going to have to work with multiple models of intelligence and make them cooperate to the best effect we can.

SAIL SPINOFFS

a concise appreciation of the
Stanford Artificial Intelligence Laboratory

by Les Earnest

SAIL grew out of the Stanford Artificial Intelligence Project, which was started by Prof. John McCarthy when he came from MIT in 1962. He and Prof. Marvin Minsky had co-founded the MIT AI Project in the late 1950s, and McCarthy had developed the LISP programming language there.

McCarthy had perceived the need for interactive computing in that era when most large computers were used exclusively as batch processors. In 1959 he wrote a memo that proposed general purpose timesharing. Part of the inspiration for this idea was a special-purpose timesharing system called SAGE, the air defense control system that was then being developed at MIT Lincoln Lab (by a bunch of people, including me) using hardware manufactured by IBM.

Working with Ed Fredkin at BBN, McCarthy developed an early timesharing system using a DEC PDP-1 computer. Fernando Corbato concurrently developed another one at MIT. Shortly thereafter, Project MAC was initiated at MIT to develop this idea further. McCarthy was invited to head that project, but chose instead to remain focused on artificial intelligence. He moved to Stanford a short time later.

In 1963 at Stanford, McCarthy began developing the first display-oriented general purpose timesharing system, also based on a DEC PDP-1, which came to be called Zeus. Among its many innovations were the first display-oriented interactive text editor. Because the PDP-1 was not a powerful processor, however, this system was interfaced to a disk on the Computation Center's nearby IBM 7090 so that jobs requiring a lot of crunching could be passed through the disk buffer, run in the batch system there, and returned to the timesharing system for interactive examination of the results.

I joined McCarthy at Stanford in late 1965 and we subsequently put together the Stanford Artificial Intelligence Laboratory (SAIL) in an abandoned laboratory building in the foothills above the Stanford campus, near Felt Lake. The first

computer there was a DEC PDP-6, installed in June 1966. After a false start with a contractor who couldn't deliver, a 6-console display system that drew text and vectors with a random-access electron beam was added in 1967. The computer system eventually evolved into a dual-processor DEC-10 and continued to provide display-based timesharing services to the Stanford community until 1992. It used a home-grown timesharing system called WAITS that was similar to TOPS-10 in outline but considerably different in detail.

Some people have claimed that "windows" were invented at Xerox PARC or SRI, but their immediate precursors were the "pieces-of-glass" that were part of the SAIL display system from the beginning. The main difference between pieces-of-glass and windows was that the former were transparent (i.e. you could see the lower layers) whereas "windows" were opaque.

A fancier display system, installed at SAIL in 1971, put a terminal using a television monitor on everyone's desk. SAIL was apparently the first system in the world that put terminals in offices — before that, the few computer displays that existed were kept in "display rooms." This display system also included an advanced keyboard that introduced the "Meta" key and other features to facilitate touch-typing. That keyboard design was picked up promptly by MIT and Carnegie-Mellon University and later by Apple, whose Command key is a direct descendent of the Meta key on the SAIL keyboard.

By 1972 the display system included a digital video switch that allowed users to select rapidly from a variety of computer-generated images or other video sources, including commercial television. There was also a speaker on each work station and a novel audio switch that used digital components to allow selection from several audio sources.

The original PDP-6 system had just 64k words of storage (which occupied eight large cabinets) and used microtapes for secondary storage. A fixed-head disc file built by Librascope, added in 1968, was supposed to function both as a swapping store and a permanent file store, but it turned out to be so temperature-sensitive that it was useless for file storage. The six remarkably large discs in this system, which were each 4 feet in diameter, were eventually sold as coffee tables — I have one in my living room. Despite its large physical size, this disc

system had a capacity of only about 100 megabytes. More reliable disks made by IBM, Ampex and DEC were added in later years.

A number of people joined SAIL in the late 1960s, including Don Knuth, who later went off on his own but continued to use the SAIL computer as his main "home" because of its many advanced features. Raj Reddy, who had just finished his Ph. D. at Stanford, continued his pioneering work in speech recognition and eventually moved it to Carnegie-Mellon University.

Another recent Ph. D. named John Chowning developed his ideas on computer synthesis of music at SAIL, leading to a patented synthesizer that was licensed to Yamaha and that made millions of dollars for him and for Stanford. Chowning later formed a computer music research group called CCRMA (Center for Computer Research in Music and Acoustics).

Art Samuel had joined the Lab in 1967 after retiring from IBM. He continued to develop his checkers program, which was the world champion at that time. One of his students developed the most advanced Go program of that era.

Dr. Kenneth Colby joined the Lab in 1968 and his group developed a number of experimental natural-language-understanding programs, including Parry, which answered questions in a manner that simulated the responses of a paranoid person.

Among the user-friendly features of SAIL was an advanced version of Spacewar, a rockets-and-torpedoes game created principally by Steve (Slug) Russell, who had developed the first version while he was at MIT. That idea was further developed by a couple of our staff members into a commercial version using a PDP-11 computer. It became quite popular at a local bowling alley and at the Stanford coffee shop, but the developers knew nothing about how to run a business and their small enterprise went nowhere.

Meanwhile, a guy named Nolan Bushnell picked up the same idea and formed a small company called Atari that developed Spacewar as their first product. Deciding that it was too complicated to be a marketing success, they sold it to another company, and went on to develop a simpler game that turned out to be quite popular; it was called Pong....

A grad student named Don Woods later took a game idea from another person and developed Adventure, which spread over the ARPAnet [predecessor of the Internet] and later evolved in various directions. Today, Adventure is considered the ancestor of almost all text-based computer games.

More serious work on computer gaming included McCarthy's chess program that he had begun at MIT and that was used in a match with one in the Soviet Union. (We lost, but it caused our Russian counterparts a lot of grief when the KGB discovered that we were exchanging telegrams containing what looked like coded messages.)

A DEC consultant named Richard P. Gruen, who used to hang out at SAIL, developed a system for controlling complex program compilations that he called RPG, which theoretically stood for "Rapid Program Generation," but also happened to be his initials. This idea was later incorporated into Unix as the "make" command.

The computer was used for text editing right from the beginning. Bill Weiher and others developed a simple text editor that came to be called SOS and spread throughout the DEC-6/10/20 community. Later a page-oriented editor called E became the primary editor in the Lab. Many features originating with E were incorporated into the emacs editor that was developed later at MIT.

I decided early on that I needed a spelling checker in order to cope with my deficiencies in that area. Fortunately, I happened to have a dictionary of the 10,000 most common English words that I had punched into paper tape when I was at MIT; and during 1960-62, I had developed a spelling checker as a subroutine in a pen-based system for recognizing cursive writing. (This system, which I had also developed, worked at least as well as the handwriting recognizers that are now appearing on the market.) As I later learned, this 1960 system was evidently the first computer spelling checker developed anywhere.

In 1966 I gave the dictionary to one of our grad students at Stanford, and he wrote a new spelling checker in LISP that clanked a bit but did the job. A few years later, another grad student named Ralph Gorin did a faster one in machine language that included spelling correction. That became quite popular in the lab.

SAIL was connected to ARPAnet around that time, and programs and data began circulating between the research sites through a mixture of donation and benign thievery. Our spelling checker spread to DECsystem-10 and -20 computers all over the net and a Unix version was subsequently developed. Such programs were included later in the personal computers that began appearing in the mid-1970s.

Another program called FINGER, which I developed to help keep track of the unpredictable migrations of our staff at all hours of the day and night, was picked up by several other DEC-10 and DEC-20 computer facilities. We later modified it to work through the ARPAnet and track the denizens of remote computers. It too was rewritten for Unix, but the author of the Unix version was not careful about security, and a loophole in it was exploited much later by the infamous Internet Worm.

Another area enriched by cooperation and innocent larceny was the development of raster graphics printing, initially based on the Xerox XGP and later on laser printers developed by Xerox, Canon and others. Larry Tesler and I had developed an early text formatting program called PUB that facilitated printing on line printers, teletypes, and microfilm, which was later modified by a Carnegie-Mellon student to print on the XGP. People at various sites, principally Carnegie-Mellon, Stanford, and MIT, developed font design software and developed a robust collection of typefaces that migrated all over the network.

Inspired by the deficiencies of PUB, a grad student at Carnegie-Mellon named Brian Reid developed another text formatting program called Scribe. Don Knuth also put one together called TeX, which became a pre-eminent standard for scientific and technical page description, and later developed a fancy font design program called Metafont.

I was a member of the ARPA committee that reviewed the initial technical proposals for ARPAnet, and SAIL became part of the original network when it started in 1969, though we had to defer regular network operation until we got enough memory to hold the rather large amount of communications software that was required.

Naturally, development work at this level created a need for food at all hours of the day and night,

accessible with minimal distraction. Around 1972 we developed SAIL's response to this need, a computer controlled vending machine which sold on credit. Called the Prancing Pony after an inn in Tolkien's *Lord of the Rings*, it still operates in the Computer Science Department at Stanford, though both hardware and software have been updated.

In SAIL's enjoyable work environment, researchers did pioneering work on computer vision, robotics, and automated assembly as well as mathematical theory of computation, theorem proving, and "common sense" reasoning. Hans Moravec's system that guided a robot vehicle, using stereoscopic images from a video camera, did pioneering work on navigation and obstacle avoidance.

Several people moved from SAIL to Xerox PARC when it was formed in the early 1970s, including Alan Kay and Larry Tesler, and took the SAIL culture with them. Others later moved to Lucasfilm to develop the computer technologies supporting "Star Wars" and other elaborate flicks.

Some of our students developed the first interactive CAD system for computer design, called SUDS for "Stanford University Drawing System," and used it to design the Super Foonly, which heavily influenced the DEC KL-10 computer. DEC later used SUDS as their primary design tool for over a decade. They also donated a KL-10 to the Lab.

SUDS was also a key resource in the formation of both Foonly, Inc., a small company (now defunct) that made computers that were DEC-10 compatible, and Valid Logic, a pioneering CAD company. SUDS was also used by Andy Bechtolsheim, a co-founder of Sun Microsystems, to design the first SUN workstation (SUN stood for Stanford University Network). Andy continued to use SUDS to design successive Sun workstations, using the 1967-vintage SAIL displays through 1987.

Other commercial spin-offs from SAIL include:

Vicarm, one of the earliest robotics companies, which made high-performance electric arms and was later purchased by General Electric.

Xidex, which developed and marketed a portable compiler called MainSail.

Imagen, which I founded, and which developed and marketed the earliest desktop publishing

systems. The company couldn't get funding, because the venture capitalists had never heard of laser printers and were not convinced that there was a market for them, but it bootstrapped to annual sales of around \$20 million before being purchased by QMS.

Lucid, which developed and marketed LISP compilers and related software.

Cisco, which appropriated Stanford-developed digital communications technology, and eventually got a license from Stanford after being threatened with legal action.

In 1979 SAIL rejoined the computer science department in a new building on the main Stanford campus, but effectively lost its organizational identity in the process. The DEC-10 computer called SAIL continued to operate for another dozen years, providing a comfortable "home" for those who had come to appreciate its features. A party was held on June 7, 1991 to celebrate SAIL's 25th birthday. It was by that time the oldest "living" timesharing system in the world.

However, SAIL was no longer maintained, and began exhibiting the computer equivalent of senile dementia. The computer was powered down for the last time and dismantled on October 4, 1991, but is still fondly remembered by many who used it for work and play over the decades. It was replaced by a small DEC workstation running Unix, also called SAIL, which has much more memory and happens to be much faster than the old SAIL computer; but it has much less character.

THE DISCOLOURATION OF PLASTIC COMPUTER CASES

by Dr. Edward Then

Senior Conservator (New Materials)
Science Museum
London SW7 2DD, UK

The yellowing or discolouration of computer cases is an extremely common phenomenon. The problem is not unique to cases made by one manufacturer, nor is it restricted to computer casings. This chemical process is comparable to the discolouration of an apple skin, and is similarly irreversible. Fortunately, in most instances the damage associated with discolouring affects only the surface of the artifact.

Background: POLYMER AGING

The polymer most commonly used in casings and housings for electrical equipment and computers is ABS. The acronym is derived from the initial letters of the three main monomers used for its manufacture — acrylonitrile, butadiene and styrene. ABS polymer was first made available in the early 1950s and, since then, has become one of the most widely used industrial polymers. It is valued by producers for its excellent mechanical properties (impact resistance, stiffness, surface quality), thermal properties (good dimensional stability at high temperature) and electrical resistance. It also offers significant resistance to chemical and stress cracking.

Polymers, including ABS, can be described as large molecules made up of simple repeating units; the word *polymer* is derived from the Greek words 'poly' and 'mer' meaning "many" and "part" respectively. Many types of polymers can be created by varying the molecular composition of the repeating unit. The total number of repeat units in a polymer chain, often referred to as the *degree of polymerisation*, may typically be hundreds or more.

During degradation, different chemical reactions occur along the polymer chain. These can result in the breaking and rearranging of chemical bonds, causing (among other things!) discolouration of the polymer. Degradation may be initiated or accelerated by numerous factors including ultraviolet light (UV), visible light, ozone and

other extraneous pollutants, intrinsic manufacturing impurities, oxygen, and heat. In the case of our computer housing, I think UV and light are the main causes of deterioration. The rate of deterioration is thought to be approximately proportional to the light intensity.

Deterioration-fighting chemicals are commonly added to polymers during manufacturing; these may include antioxidants, antiozonations, light stabilisers, UV stabilisers and fire retardants. The type of additive used will be determined by the composition and application of the finished product. As the polymer ages, most of these additives are consumed while they hold back the degradation process; once the stabilisers are used up, the polymer is often left unprotected and will deteriorate very rapidly. Attempts have been made to restabilise polymers, but it is not known how well these will work, and the topic demands considerable exploration. The Science Museum in collaboration with other institutions is currently sponsoring research in this area.

WHAT SHOULD WE DO NOW....

and how can we extend the life span?

The best advice is, perhaps, to do nothing. Personally I would advise that discoloured surfaces should be left untreated. Maybe, one day, the discolouration will be seen as desirable or inevitable, like the patina on metals! In any case, each example must be evaluated individually, preferably by a conservator who deals with plastics.

Dirt and grime are a separate problem, and may be cleaned with distilled or deionised water. Stubborn stains can be removed with a non-ionic detergent. The cleaned surface must then be dried immediately. A word of CAUTION: When cleaning with water, use a cloth or cotton wool that is only slightly damp, and avoid making contact with metal parts — which may corrode — and with the electronics. Avoid using solvent; some solvents may appear harmless on contact, but will react with the plastic over time, crazing or cracking the object later. Use only soft cloth or cotton wool to dry the case to avoid abrasion or scratching.

Until there is a solution to this problem, the only prudent strategy is preventive conservation. Try to

keep the computer away from strong light, especially direct sunlight and other strong UV, and from any heat source. Also keep it covered when it is not being used, to forestall the build-up of dust. I will try to keep the editor informed on the progress of the research projects.

BERKELEY'S UCBVAX CHANGES JOBS

by Cliff Frost

Network Services

Computer Science Department, UC Berkeley

On Friday, August 19, 1994, at about 2 p. m. Pacific Time, programmers gathered in the old CS department computer room, fourth floor of Evans Hall, UC Berkeley, amongst the scattered remains of network wiring, ancient hardware, ghosts of legends, and general debris, for a mysterious and moving ceremony; a rite of passage for a computer, and perhaps for its human caretakers. A semiologist could write a thesis on this event, but here we confine ourselves to the facts. Keith Sklower opened with a brief history of ucbvax:

“In the summer of 1978, the computer science department took delivery of the campus' first Digital Equipment VAX computer, obtained via a grant from NSF (due in large part to the efforts of Prof. Richard Fateman). In fairly short order, it was running a variant of UNIX developed by Bell Labs. (Local CS people were interested in adding virtual memory support, which AT&T UNIX lacked, which eventually led to the widespread interest in BSD, but that's a different story).

“The people at Bell Labs offered to have their computer call up ours in order to facilitate research collaborations, using the UUCP protocol. We had to choose a node name, and ‘ucbvax’ seemed to follow their naming conventions. Netnews and the birth of USENET followed closely.

“After about three years, the mail-handling and news functions of the departmental VAX were chewing up more than half the cycles, so it was decided to segregate those functions onto a separate machine, when the opportunity arose. So ‘ucbvax’ became a VAX/750 devoted specifically to those functions. For a long time, it was one of two gateways between the ARPAnet and the Berkeley campus.

"As the load on ucbvax-the-750 began to exhaust its capacity, there was talk of replacing it with some flavor of SUN workstation, but a DEC sales person got wind of this and thought it would be much better for DEC if ucbvax were to stay a VAX, and managed to 'upgrade' the 750 to a DECstation 3200, its final incarnation."

Keith then assured the assembled company that although the machine was retiring, it would be following recent local tradition by immediately coming back to work in another capacity — as a card-key access system controller for the UC Police department. His eulogy was followed by Eric Allman's moving tribute:

"Alas, poor ucbvax! I knew him, Horatio. A machine of infinite jest, of most excellent software. He hath borne my mail in his queue a thousand times. And now how abhorred in my imagination it is! My gorge rises at it. Here hung those disks that have spun I know not how oft. Where be your news now? your dialins? your routes? your flashes of congestion that were wont to set the department on a roar?"

The actual power-down was delayed by about 20 minutes, as about 74 pieces of queued email were moved to another machine for eventual processing. Following this, Keith and Eric halted the computer that had carried the name of "ucbvax.berkeley.edu" for the last several years of that venerable name's history. Kirk McKusick turned the power off, an honor due to him as the first person to power-on that particular piece of hardware. At the time it was turned off, ucbvax was the last operational VAX in CS, so — to stretch the truth only a little — it was both the first and last VAX in the Computer Science department at UC Berkeley.

The assembled multitude of T-shirted and blue-jeaned programmers (more than a few of us also sporting just a touch of silver in the hair) applauded enthusiastically, then dug into carrot cake and diet Pepsi generously provided by Keith.

Although ucbvax's IP addresses are retired, significant email traffic is still being supported under that name, rerouted earlier (through the magic of MX records) to a machine supported by Information Systems and Technology's department of Data Communication and Network Services. This computer is named, with full cognizance of the irony implicit, "nak.berkeley.edu".

AUSTRALIAN COMPUTER MUSEUM SOCIETY ANNOUNCES INCORPORATION

The Australian Computer Museum Society is now incorporated as from the 16th December, 1994, in New South Wales, under the Associations Incorporation Act 1984.

The President is:

Graeme PHILIPSON
phone 02-286 5900
fax 02-267 2094
e-mail graemep@spg.mhs.compuserve.com.

The Secretary is:

Michael CHEVALLIER
phone 02-498 3383
e-mail chevallier@decus.org.au.

The Treasurer is:

John GEREMIN
phone 02-764 4855
fax 02-764 4679
e-mail geremin@decus.org.au.

The society's ADDRESS is:

Australian COMPUTER MUSEUM Society Inc.
P.O. Box 103
Killara, NSW, 2071
AUSTRALIA.

The Society is seeking *business sponsorships* from computer-related businesses to help pay the rent on its storage space at Homebush.

In the new year the Society will be looking for *vendor sponsorships* from computer manufacturers to underwrite employment of part-time curators.

The Society also needs *commercial sponsors*, and *donations* from any interested business, to help us establish offices and to institute special projects.

Current membership fees are AUS\$25.00 for individuals, AUS\$10.00 for students or pensioners. Membership forms are available from the Secretary or Treasurer.

[Note: At this writing, AUS\$1.00 equals slightly more than 75 US cents. — Ed.]

RETROCOMPUTING MUSEUM ON THE WEB

Some programming languages may live forever. (Not that FORTRAN thread again!) Some die gracefully and pass to the care of historians. And some are, well, roadkill.

At the end of a hard day grinding code, whether in your favorite or least favorite dialect, it's salutary to relax with a splendid showcase for some of the world's most perplexing ideas. Think of the write-only text editor TECO — uniquely meaningful because any conceivable ASCII string means *something* in it! Think of BLISS, whose hopeful acronym belied its tortuous internals! Think of.... I was going to mention INTERCAL, but I won't.

Now dozens of forgotten and justly ignored languages have been collected by Eric S. Raymond, Lord of the Jargon File, and displayed to great advantage in the *Retrocomputing Museum*. Some of them have actual merit — there's an implementation of Konrad Zuse's *Plankalkül* in the works; some are sillier than the above, if that's possible. Many of them can clutter up *your* hard disk through the magic of ftp. If you crave to savor constructs even weirder than ALTER in COBOL, point your Web browser to <http://www.ccil.org/retro/retromuseum.html>. Just don't say we didn't warn you.

NEWS OF MUSEUM ACTIVITY IN SWEDEN

by Anders Hultman

On the West Coast of Sweden, in the tiny town of Stenungsund, about 40 km north of Gothenburg there is a home computer museum. Probably the world's only museum devoted entirely to home and personal computers, it has been operated by the software company Hogia for about two years.

Groundwork was done in the late 70's and early 80's when Hogia developed software for different platforms, mostly CP/M based; they requested a computer of each type to be able to test the software on it. When they had the whole attic full of old computers they opened the museum. It contains several hundred personal computers from many countries, and also interesting information

about the persons who started the personal computer business.

During the winter it's only opened on request, but during school holidays and all summer it's open almost every day. It's located in Hakenas Gard at the southern entrance of Stenungsund, and can be reached at:

Hogia persondatormuseum
Hakenas Gard
S-444 28 STENUNGSUND
Tel. no. +46-303 696 48
Fax no. +46-303 819 97
BBS +46-303 675 89 (not open 24h)
Internet: aah@hogia.se

Aside from the museum they are involved in a "saving old computers" project run by volunteers at the University of Uppsala.

COLLECTOR'S GUIDE: SECOND EDITION IN PROGRESS

Having received a few inquiries as to whether there would be a second edition of the invaluable *Collector's Guide to Personal Computers and Pocket Calculators*, we asked Dr. Tom Haddock, who says he is actively collecting material for the book. He asks that anyone with a micro not included in the first edition fill out and submit a copy of the Collector's Register form at the back of the book, or contact him at Box 2626, Ann Arbor, MI 48106. Photos are especially welcome.

The second edition "will not be out real soon" but apparently we can look forward to a substantial advance in coverage. (Come to think of it, the CHAC does have one micro he hasn't listed....)

Tom asked us, while we're on the subject, to offer his apologies to those who've sent him letters and not received replies. He's "literally months behind" on his mail but promises to answer every letter eventually.

SUN HARDWARE REFERENCE UPDATE

James Birdsall's amazing SUN Hardware Reference is....well, we won't presume that it's complete, but we can't imagine how it could be *more* complete. Any SUN enthusiast will want a copy, whether as a support library or simply for enjoyable reading.

This document is available from the CHAC Request Daemon; send mail to engine@win.net with the subject `sunref_all_z`. The mail you receive in return will be decodable to the file `referenc.zip`. Note that you **MUST** have UUdecode and an unZIPping utility! If you have FTP on your system, use anonymous login to [ftp.netcom.com:/pub/ru/rubicon/sun.hdwr.ref](ftp://ftp.netcom.com/pub/ru/rubicon/sun.hdwr.ref). The file `referenc.zip` (88K) contains all five parts; individual parts are available as plain ASCII (217K total) in the `reference.parts` directory. Here's the TOC:

PART I: OVERVIEW

CPU/CHASSIS

Sun-1, Sun-2, Sun-3, Sun 386i, Sun-4/SPARC

General descriptions of the models, including processor/fpu/speed, bus, chassis type, OS support, etc.

Processor Info:

SuperSPARC, microSPARC, etc.

PART II: FAQ

ROM Monitors:

How to use the ROM monitor built into every Sun (boot instructions and other tips).

Using a Terminal as Console:

Notes on using a serial terminal instead of a Sun framebuffer and keyboard.

Memory Display on Startup:

How much memory a system has.

Miscellaneous Questions and Answers

Facts in Search of a Home

Miscellaneous Pinouts

PART III: BOARDS

CPU, memory, video, SCSI:

Descriptions of boards by type and part number, including pinouts, jumpers, DIP switch settings, and LEDs.

PART IV: BOARDS (cont'd)

non-SCSI disk controllers, tape controllers, Ethernet, serial/parallel/other commo, floating-point/system accelerator, backplanes, other, cross-referenced by bus

Descriptions of boards by type and part number, including pinouts, jumpers, DIP switch settings, and LEDs.

DISKS

SMD, MFM, ESDI, SCSI:

Descriptions of models commonly used, including jumpers and switch settings.

KEYBOARDS

Types 1-5c:

Descriptions of types of keyboards, what CPUs they work with, and any configuration information.

MICE

Sun-1, Sun-2, Sun-3, Sun-4:

Descriptions of types of mice, what CPUs they work with, and any configuration information. Alternatives: Trackballs, etc.

MONITORS

ECL mono, TTL mono, color:

Descriptions of monitor types, video boards they work with, and any configuration information.

FLOPPY DRIVES

Descriptions of models commonly used, including jumpers and switch settings.

TAPE DRIVES

9-track, QIC-11, QIC-24:

Descriptions of models commonly used, including jumpers and switch settings.

PART V: APPENDICES

Cardcage configuration tables:

What cards go in which slots in which machines.

Part number index:

Index of all known part numbers, with references to larger descriptions, if any, in the main body.

Repairs and Modifications:

Repair and modification information as contributed by various net.people.

Announcement Dates/List Prices:

Announcement dates and list prices for various configurations.

Author's Notes.

Miscellanea.

Bibliography/Acknowledgments:

Contributors, and documents used in compiling this reference.

SPOTTER ALERT

Copies of the ENGINE, the FAQ, and project information have been pouring out to print and broadcast media, especially in Silicon Valley. We do have tearsheets of most of the ink we know about. But is there ink we haven't heard of? Once more, with feeling: If you spot any mention of CHAC or the ENGINE in any periodical, *please*,

- If your copy of the piece is clippable, clip and mail to the Palo Alto address.
- If you can't spare the physical copy, send the text as net.mail to engine@chac.org, or photocopy and fax to the Palo Alto address.
- If you're too busy for that, just send the publication name, date and page number and we'll do the hunting.

Thanks! (And thanks to the spotters who have given us invaluable help with keeping up so far.)

MONEY, rev 2.3....

Are you one of the 174 people who have promised to subscribe to the ANALYTICAL ENGINE — but never quite managed to write us a check?

If all 174 of you sent your check today, we'd have about \$5,000 more than we do in the bank — an amount that would keep the CHAC solvent for a surprisingly long time. It would also bring *you* extra benefits like a thicker ENGINE, a Web page, ready acceptance of your donated hardware, and maybe even....exhibit space? Think of it.

The CHAC is longing to grow and rarin' to go. But only *your dollars* can get us off the dime. If you've promised us a sub, please, make it surface today.

(If you haven't, you could still subscribe out of the blue and surprise us.)

YOU PUBLISH! OR WE PERISH!

Call it a victory of sorts! This issue of the ENGINE is no thinner than the last one. We splattered a nag for articles all over October's back cover, in italics this time. We received.... the material you see here, and half a dozen e-mailed apologies for articles not written — most of them pleading indifferent writing skills, or lack of time.

Look, people. It's just NOT that difficult.

Remember the ten-minute war story you told at the last Chinese dinner? Remember the mainframe nightmare that somehow gains polish with every iteration? Remember the very first — or very worst — [insert name of specialty] job you ever got paid for?

WRITE IT DOWN. Don't wait and try to make it flawless. Our staff will apply the debugging, tweaks and lacquer job that brings your prose to the enviable standard set by every issue of the ANALYTICAL ENGINE. But your article, unlike a hundred-thousand-line COBOL program, doesn't have to be perfect on the first pass. (Case in point: When I typed that last sentence I made a spelling mistake. Can you see it now? Exactly!)

Make *your* mark in the history of computing in California. (If you've made one, make another one.) Write an article for the ANALYTICAL ENGINE. You'll be glad you did. So will we.

OVERVIEW OF BUREAUCRATIC PROCESSES

INTERNSHIP

The ENGINE now has a student intern, Arjun Kanodia of Palo Alto. Mr. Kanodia began by putting many of our records in order, which they desperately needed, and has progressed to contacting small museums and galleries in the South Bay to assess their interest in a proposed CHAC exhibit. We appreciate his conscientious labors and look forward to working with him frequently.

GIVING UP ON PLASTIC

We've concluded that we won't be able to accept payment by VISA card in the foreseeable future. Two knowledgeable and patient advisers, one an executive of a local bank and the other a private consultant, were encouraging at first but concluded (correctly) that the whole process would be too expensive. Our attempt to negotiate with Visa International directly was rebuffed because our volume didn't interest them.

Before the next issue of the ENGINE appears, we will introduce the CHAC to local brokers of digital cash transactions ("e-cash") including CommerceNet, CyberCash, First Virtual Holdings, and Netscape. These services are experimental but promising, and maybe encrypted e-mail will be a better choice for us than plastic.

Book Review: THE MAC BATHROOM READER

Owen Linzmayer

Sybex, Inc., 1994

306 pages, US\$10.00 (paper)

Reviewed by David T. Craig

CompuServe 71533,606

Owen Linzmayer's *The Mac Bathroom Reader* is an extremely accurate history of Apple Computer, focusing on the Macintosh computer's first decade, but embellished with several sections of irreverent humor. In Linzmayer's words, this book concentrates "on the events, people, products, and companies that made the Macintosh what it is today....a unique collection of amazing anecdotes, interesting lists, exploded myths, silly stories, embarrassing quotes, and other useless factoids meant to be enjoyed at your leisure, wherever you happen to enjoy reading."

The book contains over two dozen chapters, but a sample will suggest how well Linzmayer knows his Mac history:

Ron Wayne: The Forgotten Founder
What Were They Thinking?

Broken Breakout Promises

IBM: The Strangest Bedfellow of All

Folon's Forgotten Logo

Bold Intros and Quiet Exits

Lemmings: Why 1985 Wasn't Like 1984

The Great Caffeine Conspiracy

The Official History of the Dogcow

Windows: How Sculley Betrayed the Mac

If these titles leave you hungry for more, you're getting the idea! As a long-time Apple observer I was impressed not only with the facts that Linzmayer uncovered, but also with his verification of these facts; I've read most of the books concerning Apple's history, and *The Mac Bathroom Reader* is by far the most meticulous. He interviewed many key participants in Apple's strategic development, including the inventor of the Macintosh, Jef Raskin, who has called this book "outstandingly accurate" in recent e-mail to me. Important historical buttressing, in my opinion, is provided by the bushel-basketful of quotations; almost every other page appears to contain a quote from some notable Apple personage.

To my mind, two chapters stand out as extremely interesting: "The Apple /// Fiasco" and "Lisa: From Xerox With Love," which provide insight into Steve Jobs' decision-making before he became a key player in Macintosh development. Linzmayer clearly demonstrates Jobs' failure to learn from his /// and Lisa mistakes, and its influence on his direction of the Macintosh project. In the Lisa chapter, Linzmayer analyzes Apple's leveraging of Xerox's work with the Alto computer system. He even includes a photo of a Xerox Alto, although a nice addition would have been a sample Alto screen showing its bit-mapped architecture, graphics capabilities, and raster fonts. The chapter concerning the Alto may contain one of the book's few mistakes; Linzmayer claims that a group of Apple people saw an Alto running the Smalltalk environment in 1979. I've read elsewhere that the system demonstrated to the Apple executives was a Xerox Dorado, but I need to check this with the Xerox employee who gave the Smalltalk demo. In the Lisa chapter, also, Linzmayer did an "insanely great" research job; I've owned a Lisa since 1984 and collected a tremendous number of Lisa-related artifacts, and I found only one statement here that I considered questionable.

The "Making of the Macintosh" chapter is a *tour de force*, a blazing fusillade of accuracy and detail. Linzmayer profiles Jef Raskin, Apple employee #31, who started at the company as Manager of Publications. In 1979 Raskin was given the opportunity to develop a game system, code-named Annie, with a target price of \$500. After some study Raskin changed the project specification to a text-based but bitmapped system, and on 11 September 1979 renamed the project Macintosh. Raskin remained with Apple and with the Macintosh project until early 1982, by which time Steve Jobs had turned the nascent Mac from a simple-to-use system based on Raskin's ideas into a face-saving mini-Lisa, less capable and innovative than the original but still a startling break from the "TV typewriter" paradigm of the CP/M and MS-DOS platforms. The amount of change required here is summed up in a terrific chart showing the Macintosh's evolution from May 1979 paper specifications — a 6809 CPU, 64K RAM, 200K 5.25" floppy, no mouse pointer, \$500 target price — to the January 1984 customer machine with a

68000 CPU, 128K RAM, 400K 3.5" floppy, mouse pointer, and a retail price of \$2495.

"Macintosh Insiders: Where Are They Now?" ends the book on a great note: a list of the names that appear inside the case of early Macintosh computers, all 50-or-so people who worked in the Macintosh division as of February 1982. Included with each name is a copy of the signature as found in the case, a description of what the person did for the Macintosh, and where that person could be found at the time of publication.

Copious photos, many of them rather rare, include the Apple 1 singleboard, the Xerox Alto computer, and a snapshot of a bulldozer and a Lisa keyboard in the Logan, Utah, landfill where Apple buried about 2,700 Lisa computers as a tax write-off in September 1989.

If you'd like a second (equally positive) opinion on this book, check Jef Raskin's review on page 145 of the February 1995 issue of WIRED magazine — in which even *he* admits to a red face. But I hope I've told you enough here to interest you in buying a copy from the author. Send US\$10 cash, check, money order, or Visa/MasterCard, for one autographed book, to:

Owen W. Linzmayer
2227 15th Avenue
San Francisco, CA 94116-1824, USA

CompuServe: 71333,3152 (preferred)
AppleLink: Owen
America Online: Owen Ink
eWorld: Owen Ink

For USPS Priority Mail service on domestic orders (optional, but faster,) include \$3 extra. Non-US customers should add \$2 for international shipping.

ACQUISITIONS

[Officially, the CHAC is not acquiring at this point. This is a heavy burden to us since, by the time we have more storage space, some of the material we've agreed to accept may have hit the dumpsters instead. But what can we do about this most painful contradiction? Only wait, and work, and hope. To those of you who have promised to donate and are waiting, we can only entreat your patience. — KC]

COMMODORE 64

Mark Greenia

The noted author of the Lexikon Services computer history stack sent us a C= 64, probably a comparatively recent one and in very nice condition. This is, we hesitate to admit, the first Commodore computer we've ever owned, so we don't know a lot about it; but it's a technically interesting device and its historical importance is unquestionable. (Anybody got a 1541 they're not using? We look forward to firing this up.)

Mark also donated copies of the *1993 Computer Industry Almanac* by Juliussen and Juliussen — which hasn't left our office desk since we got it — and of *IBM's Early Computers* by Bashe et al. We thank our brother in history for his thoughtfulness.

BOOK PURCHASES

There's no end to computer books, it seems. And they're so small, so cheap (used) and such fun to read.... The overriding consideration is that, if we pass over a copy of a significant title, we may never find a second one. Recent purchases of particular interest include:

8085A Cookbook, Titus, Larsen and Titus (Blacksburg)

Apple Machine Language, Inman and Inman (Reston)

Beginner's Guide to Computers & Microprocessors, Adams (TAB)

Handbook of Microprocessor Applications, Kuecken (TAB)

TV Typewriter Cookbook, Lancaster (Sams)

LETTERS

CURTA CALCULATOR FOR SALE

A friend of the CHAC has donated a Curta Model 2 (large) "peppermill" calculator, and instructed us to sell it to pay operating expenses. The calculator itself is in mint condition; the black metal case nearly so, with some scuffing only on the bottom. Full original docs are included.

Recent sales of Curta Model 2's at retail or auction have realized US\$550 to \$625. We will accept the best offer over \$550 that we receive by May first, 1995.

Kip Crosby
engine@chac.org

SAVE THAT SDS 930!

Save that SDS 930, if at all possible! The Berkeley Timesharing System, an important and significant ancestor of UNIX developed at the University of California at Berkeley, was written for the SDS 940, which was a slightly modified SDS 930 — modified to support demand paged virtual memory. That was the first machine I ever used, back in high school in 1968, and it was only later that I learned about its historical significance.

My first paid job as a programmer was at Com-Share in Ann Arbor, Michigan, in the summer of 1972. One of the first things I had to do was to port a big customer application written in FORTRAN from their SDS 940 system (billed as Commander I, but really still running the Berkeley Timesharing System) to their new XDS Sigma 7 system (billed as Commander II). The biggest problem I encountered in that job was that the character handling parts of the FORTRAN code used H3 format on the 940 and H4 format on the Sigma 7 (H3 was 3 characters per word, H4 was 4 characters per word). There was also an ASCII-to-EBCDIC change, but that was only a minor hassle.

I still have my 1972 vintage Commander II FORTRAN manual, but that's one of only two relics I have of my connection to Com-Share. The other one is an aluminum bar with "Scientific Data Systems" engraved across the top. It once graced some SDS computer, but by the time I got to it,

the relay rack had been stripped of everything but the nameplate.

Doug Jones

MORE ON THE PERIPATETIC 74x181

I read with interest the article in ENGINE 2.2 about the 74181 ALU chips that were used in the Xerox Alto. These chips were also used in a lot of other minicomputers of the time — many Unibus PDP11's contained them; the PDP11/45, for example had 4 74S181's on the data path module, another 3 on the memory management address path, and yet a lot more in the floating point coprocessor. They were also used in the Philips P850 which contained a pair on the ALU module (what was that board called? I forget), and I guess other machines. The idea was quite simple: it's well known that there are 16 possible 2-input gates. Think of filling in the 4 lines of the truth table — there are 2 ways to do it for each combination of inputs, leading to 2^4 for the entire table. The 74181 had 4 select inputs which allowed you to select one of these functions — yes, all were available — to be applied between the 2 4-bit input words.

But there was more; a carry system that could be enabled by the fifth control input, or M bit. This turned the XOR function into addition, and also gave useful(ish) functions for the other select codes. I'm not sure if any machine ever used those though. Many machines had a ROM between the instruction decoder/microcode word and the ALU inputs. Thus not all the functions were available to the user. This carry system allowed propagate/generate carry between cascaded chips, rather than the slower ripple-carry system. A companion look-ahead carry generator, the 74182, was used for this purpose. Logic diagrams and equations for both these devices are in any good TTL data book.

Now for another machine that used 74181's, and which was related in a sense to the Alto. This is the PERQ, probably the first commercial workstation. The 20-bit CPU contained 5 of the 74181's. The machine was soft-microcoded, and a 4-bit field in the microcode word selected which operation was performed. Twelve of the 16 logic operations (i.e. with the carry logic turned off) were available, along with two versions of add and subtract, with or without carry-in. On the later PERQ 2

machines, a bit more logic allowed the fundamental steps of shift-and-add multiplication, and even of non-restoring division, to be performed in 1 microcycle. I believe a small ROM is used to decode the ALU field, the carry bits, and the appropriate signals from the multiplier register and to provide the control signals and carry input to the 74181's.

The PERQ was a fine machine that has been largely forgotten. They had a graphics accelerator, the Raster-Op machine, which was used to efficiently combine 2 bitmaps on the screen; built-in Ethernet; hard disks, and so on.

TTL logic is qualitatively important to hardware of its period and, as was mentioned in the article, a lot of TTL chips are likely to disappear very soon. Even the AMD 29xx series of bit-slice stuff is officially on the endangered species list! Those chips (the 2901, 2910, etc) turned up in a lot of late '70s hardware such as VAXen, PERQs, disk controllers, and much else. A UK organisation which I am involved with, the P850 User Group, is trying to find ways to replace them, possibly by programming some of the advanced GAL-like chips now available. We've not found anything suitable yet, but we're actively looking. That's one way to assure that some of this great hardware will still be usable in years to come.

Tony Duell
ard@siva.bris.ac.uk

HELP WANTED WITH RESEARCH ON CALCULATORS

I'm presently doing research on the history of early pocket electronic calculators and I could use your readers' help. The tremendous growth and rapid price drops of the (then) newly developed integrated circuit chip was heavily fueled by the popularity of pocket calculators in the consumer market during the 1970 to 1975 time period.

As a result, there were a number of medium and small sized companies (many from California) that jumped onto the calculator manufacturing bandwagon. They pushed that early technology to the limit. Unfortunately, information about these companies is hard to come by — so far impossible, in many cases.

Companies like Berkey/Litton, Busicom, Calcupen, Calfax, Corvus, Datalog, Kings Point, Litronix, M.I.T.S., Melcor, National Semiconductor, Netronics, Rockwell, Summit, TeleSensory, Universal Data, Unicom, Victor, and Vista had a major impact on the way we calculate (and compute) today. And it would be a shame to let that information die away.

If you have ANY information on any company (named or not) that was involved in calculators at that time, please contact me. Any and all information would be appreciated.

Contact:

Guy Ball
14561 Livingston Street
Tustin CA 92680 USA
email mrcalc001@aol.com
fax/message +1 714 730-6140.

[Note: This is for Guy and Bruce's forthcoming book which intends to list "every LED/LCD-type, nixie-type, or fluorescent-tube-type pocket electronic calculator ever made." A tall order, but an ambition that absolutely compels applause. If you have any information that Guy might find useful, please make it a priority to contact him. — KC]

QUERIES

[Queries are sorted by subject, and within that, by model if applicable.

If the person querying has permitted us to publish an e-mail address, we have done so, and please reply directly to it; otherwise, reply to engine@chac.org or the Palo Alto address, and we will store and forward.]

AIR DEFENSE INFORMATION WANTED

I would be most grateful if anyone out there in cyberspace could point me toward any available sources on the Air Defense Integrated System (ADIS), a failed air defense project of the early 1950s conducted by the Willow Run Research Center at the University of Michigan.

The ADIS project began about 1950 and continued until 1953. In sharp contrast to the centralized, digital-computer-controlled SAGE system which eventually won out, ADIS was based on analog

computing aids and semi-automated communication among decentralized command posts. I have so far been unable to locate more than passing references to the project in any published literature.

Thanks!

Paul N. Edwards
Program in Science, Technology and Society
School of Engineering, Bldg. 370
Stanford University
Stanford, CA 94305-2120
pedwards@pcd.stanford.edu

APOLLO: HELP WANTED WITH ORPHAN SYSTEMS

I've been given two Apollo Domain series 3000 machines, about which I know nothing. One is a server and one appears to be diskless; they are connected using Apollo token-ring. Both will happily boot to the login prompt. Unfortunately (you guessed it), I have no docs, no operating system disks (one machine has a 5.25 floppy, but no tape), and no information about the accounts which are on there, if any.

So: how does one bring these machines up from scratch to allow a new root password to be set? I know such information isn't parted with lightly, and you have no guarantee that I'm not some pimply hacker trying to crack somebody's system, but you'll just have to believe me.

Given that I have done this, is there anything useful I can do with these machines? We have a lab Linux network and they would make two excellent X terminals — is this feasible? I don't even know if the window system is an X-like system, or whether they have/can be given an Ethernet connection — I have never used these machines before.

Please help to save these machines from ending up in the skip.

Mark Robinson
mark-r@spec0.ee.man.ac.uk

APPLE II SOFTWARE: SOURCES URGENTLY WANTED

Where the h**l can people still buy Apple software, as distinct from Macintosh? We get 5 or 10 people a week asking. E-mail on this subject would be more than appreciated.

thanks,

Dan Henry
Daniel.J.Henry@dartmouth.edu

[We've said it before and we'll say it again; the Apple II remains a remarkably healthy platform. Let's, please, get together on this and help Mr. Henry out. — Ed.]

APPLE LISA: TIME TO DIG OUT THE CARDBOARD BOX

I am writing a book about the rise and fall of the Apple Lisa computer. Anyone with comments, gossip, folklore, materials, etc. is encouraged to contact me by e-mail at any of the addresses below. If you have hard copy material, I will gladly pay copying/shipping charges and give credit in the book.

Michael Posner
Lisa Lives Users' Group
cwld@ix.netcom.com
AOL: Mac n Lisa
CIS: 72262,447

CALCULATORS WANTED

I continue to look for mechanical calculating devices and slide rules to add to my collection. Some of my interests include older wooden slide rules and small adders (dial-type, stylus-driven type, etc.), unusual slide rules (circular or spiral). lever-set calculators (such as Brunsviga, Marchant, Walther, and others) and arithmometers (Peerless, Bunzel, Archimedes, Thomas, Burkhardt, and others). I collect all types of devices which calculate in some way. I also enjoy discussing this topic and sharing historical information on old calculating devices.

Please respond to me either by e-mail, rdecesar@pcocd2.intel.com, or phone me at (916) 356-5769. Thanks.

Bob De Cesaris

CDC CYBER SYSTEMS: INFO REQUESTED

I am doing a paper on an old computer architecture, the CDC Cyber 70 and model 6000 series, typically systems like the 72/73/74 or the 6400 and 6600 mainframe computers.

If anyone has information on how the core was used between jobs, core addressing methods, PPU programs, channel allocations, buses and any other interesting data about the architecture, I would really appreciate mail or email.

Thanks.

Gregory Pilkington
Tape Division, StorageTek
plg@qualifier.stortek.com (129.80.216.162)
(303) 673-3531
Fax: (303) 673-7694
2345 Clover Basin Drive, MS 0224
Longmont, CO 80501 USA

DEC: DECODING WANTED FOR ACRONYMS

I was looking in a local charity shop (computer history appears in the oddest places), and found a copy of 'Programming Languages', a handbook of the DEC-supplied languages for the PDP-8, from 1970. I quickly bought it for 45p (<US\$1). Anyway, there are a couple of things in it I'd not heard before, and I wondered if anyone else had comments.

1) I have read in several books that FOCAL stood for FOrmula CALculator. According to this book it's 'Formulating On-line Calculations in Algebraic Language'. Has anyone else ever heard that one?

2) The character tables in appendix B2 seem to be ASCII with the high bit set, but are headed ANSCII-1 Character set. ANSCII stands for American National Standard Code for Information Interchange. When was the National dropped from the name?

Tony Duell
ard@siva.bris.ac.uk

DEC PDP-11: PLUG-COMPATIBLE PARTS SOUGHT

I'm trying to find a source of replacements for spares for products built by a company called ACC, Associated Computer Consultants.

In the early 1980's ACC sold communication controllers and other boards that plugged into the DEC PDP-11 Unibus. We are still running a control system that uses some of their boards. They're working fine, but we're interested in obtaining some spares. The boards we have are ACC products called UMC TIO and UMC Z80.

I have an old phone number for ACC in Santa Barbara but it's no longer in service. I looked through some DEC-oriented magazines but didn't find them.

Jonathan Jacky
jon@radonc.washington.edu

DEC VAX: IN SEARCH OF SECRETS

I'm interested in material concerning Digital's VAXes, especially on the design of VAXes or VAX variants — chips, systems, or architecture varieties — whether or not the results were ever actually built.

This specifically includes such things as modifications of the microcode (in microcoded variants), embedded systems, and VAX clones, whether authorized or not. I'd also appreciate references to published literature (except Digital's own public product documents) and suggestions for additional contacts. At your request I'll keep your identity confidential.

Peter Kaiser
kaiser@acm.org
+33 92.95.62.97
FAX +33 92.95.50.50

GEC 4000 SERIES, ANY TAKERS?

I was just wondering, has anyone ever spent much time working with GEC 4000 series machines? When I worked at Prestel, some years ago, we had roomfuls of the things. I'd be interested/ intrigued to know if anyone else ever came across them! Are they still made/supported, and what else were they used for? Info/trivia most welcome,

cheers,
Phil Clark
phil@stimp.demon.co.uk

GO COMPUTER: RESCUE INFO NEEDED

I have been, er, "gifted" with a "GO" pen based computer. The catch is that it doesn't seem to boot right now. The BIOS message appears when power is applied, but nothing further takes place. I recall reading about this system back in 90/91 (which is also the copyright date on the BIOS message) so it isn't totally historical (hysterical... perhaps). The physical description is:

Unit is about 9" x 13" x .5" and heavy

The screen is touch sensitive (passive)

A slot at the top of the unit, presently empty, appears to be for a removable hard-drive

A D-type connector at the top appears to have a 50-flange (i.e. Centronics style) male connector

The external floppy drive has a 25-pin SCSI connector, another of the odd "mini-Centronics" connectors, and no visible power supply connector

I have no cables or other hardware.

I have a bunch of PenPoint OS disks and the PenPoint developers tool kit (whee!)

So: is anyone out there familiar with this machine? Can anyone provide some information about how to get the darn thing to start up? How about a supplier for cables and such? My memories of what I read about this machine are that PenPoint is an OS/environment based upon a "notebook" metaphor with tabs along one side of the "notebook" to activate another app. I will hazard a guess that the mysterious flanged connector is intended to connect to a "docking station" of sorts which has a SCSI interface, rendering the floppy-drive useless without this additional hardware. Apparently the company sank without a trace

before the recent PDA craze. This seems like a classic "orphaned hardware" story and I have a personal reason to look for info: my director has challenged me to "do something cool" with this beast. How can I but try? Feedback via email or post, please, of any information that seems generally interesting.

Fritz Lowrey

University of Southern California
jlowrey@ucs.usc.edu

**HP-10:
URGENTLY WANTED BY EX-OWNER**

Stephen Meyer of Monterey, CA wishes to hear from anyone willing to sell an HP-10 printing calculator — the original, larger model circa 1977-78. He had one in use until a few years ago and would very much like to replace it. Anyone with an HP-10 or a lead to one, please contact:

Stephen H. Meyer
Cypress Tree Inn
2227 North Fremont
Monterey CA 93940 USA
+1 408 372-7586 voice
+1 408 372-2940 FAX

**INTEGRATED SOLUTIONS SUN CLONES:
ANYTHING!**

I'm looking for information on SUN (680x0) "clone" systems made by Integrated Solutions. Please contact me if you know anything about these systems. I'm specifically interested in later systems with a desktop-GUI, but any info would be appreciated.

I'm also *very* interested in acquiring one of these systems, working or otherwise, and any software, docs, or sales/marketing literature for these systems.

Thanks!

Bill von Hagen
wvh@transarc.com

MATTEL AQUARIUS SWAPS WANTED

I would be curious if anyone knows much about Mattel's Aquarius computer. I'd like to hear if anyone ever tried the CP/M box or modem that was apparently developed for the Aquarius and then cancelled. Also, I have some Aquarius stuff I'd like to trade for other Aquarius items, particularly the following:

Biorhythms cartridge (loose; I could get a photocopy of the instructions though)

Mini Expander (with two control pads — I could provide a photocopy of the instructions)

Aquarius CPU (with original Guide to Home Computing and Simplified Instruction Cards)

The Aquarius is a Z80-based personal computer Mattel released in 1982-83. The MiniExpander adds a second cartridge slot for extra RAM, and hand controllers for game play.

I'm most interested in OTHER Aquarius items (cartridges such as Night Stalker, Utopia, the RAM cartridges, the printer, etc.) but will consider trades for Intellivision or other classic video game items. Thanks,

David
6500dtpt@ucsbuxa.ucsb.edu

ARTICLES NOTED

"Computer Creators: Adam Osborne," Mike James, *Computer Shopper*, December 1994, pp. 557-558. A summary of Osborne's adventures as a writer, developer and entrepreneur.

"Ensuring the Legibility of Digital Documents," Jeff Rothenberg, *Scientific American*, January 1995, pp. 42-47. Problems of durability and readability in stored digital media. Highly recommended!

"Heavy Donation," *Leading Edge*, November 1994, p. 19. Sidebar with photo on the donation of a Cray 1S to the National Atomic Museum at Kirtland AFB, New Mexico.

PUBLICATIONS RECEIVED

Australian Computer Museum Society Newsletter #2, 25 October 1994. Member profiles; Teaching the History of Computation; Aims and objectives; Philipson profiled in AUSTRALIAN; more. 8 pp. From Jim Walsh.

Charles Babbage Institute NEWSLETTER, Volume 17 Number 1, Fall 1994. ARPAnet at 25; The Year at CBI; Recent publications; CBI on gopher; Research in the History of IP; IBM Oral Histories; Burroughs Records Project. 10 pp. From Judy O'Neill.

The Computer Museum: *Annual Report 1994*. Ten-year history; balance sheet; photo highlights; list of sponsors and donors. 24 pp.

The Computer Museum NEWS, Winter 1994. Recommendations on children's software and holiday gifts. 4 pp. From Gail Jennes.

Hewlett-Packard *Journal*, recognizing and publicizing technical contributions made by HP personnel.

Volume 45 Number 6, December 1994. Digital audio tape; state machines; debugging; wavelet analysis; test vectors in manufacturing; software project management; clock design and measurement; more. 1994 index. 120 pp.

Volume 46 Number 1, February 1995. Special optics issue with particular attention to fiber-optics, reflectometry and improved lasers. *Caution*: Underlying physics is of breathtaking complexity. 124 pp. From the editors.

HISTORY OF COMPUTING: An Encyclopedia of Computer History by Lexikon Services. Version 3.1, December 1994. Hyperstack for MS-DOS or MS-Windows, greatly expanded from previous versions, with digitized photographs. Approximately 800 pages. US\$19.95. From Mark Greenia. *Caution*: This version exhibits compatibility problems with Novell DOS.

International Calculator Collector, Issue 7, Fall/Winter 1994. Sinclair Story; Upcoming Texas calculator/instrument show, Craig profile; Photo Album; Pricing trends; classifieds, resources, more. US\$12 per year with membership (\$16 foreign). From Guy Ball.

The Z-Letter, newsletter of the CP/M and Z-System community.

Number 33, September/October 1994. Palmtech CPUZ180; CP/M CD-ROM in beta; DISCAT, TERM III, WDCOUNT; YASMIO design history; Amstrad PCW8256; correspondence, resources and technical discussion. 20 pp.

Number 34, November/December 1994. Novell/DRI and Echelon products; CP/M CD-ROM release version; Kaypro 10 Tinker Kit; reviews of Augarten, Haddock, Greenia; Z-System scripting; correspondence, resources and technical discussion. 22 pp.

US\$18 for 12 issues (2 years); Canada/Mexico, US\$22; International, US\$36. From David A. J. McGlone.

ADDRESSES OF CORRESPONDING ORGANIZATIONS

Australian Computer Museum Society, PO Box 103, Killara 2071, NSW, Australia. Michael Chevallier, secretary.

Charles Babbage Institute, 103 Walter Library, 117 Pleasant Street SE, Minneapolis, MN 55455. Judy E. O'Neill, associate director.

The Computer Museum, 300 Congress Street, Boston MA 02210. Brian C. Wallace, curator of historical computing.

Hewlett-Packard *Journal*, Hewlett-Packard Company, Box 51827, Palo Alto CA 94303-0724. Richard P. Dolan, editor.

Historical Computer Society, 2962 Park Street, #1, Jacksonville FL 32205. historical@aol.com. David A. Greelish, director and editor.

Hogia persondatormuseum, Hakenas Gard, S-444 28 Stenungsund, Sweden. Åse Hogsved, director.

International Association of Calculator Collectors, 14561 Livingston Street, Tustin CA 92680-2618. Guy Ball, Bruce L. Flamm, directors.

International Interactive Communications Society, 2601 Mariposa Street, San Francisco CA 94110. Sheila Farrell, membership secretary.

Lambda Software Publishing, 149 West Hilliard Lane, Eugene OR 97404. David A. J. McGlone, editor and publisher.

Lexikon Services, Box 1328, Elverta CA 95843. lexikon2@aol.com. Mark Greenia, director.

The Mathematical Intelligencer, Springer-Verlag New York, 175 Fifth Avenue, New York, NY 10010. Chandler Davis, editor-in-chief.

Unusual Systems, 220 Samuel Street, Kitchener, Ontario N2H 1R6, Canada. Kevin Stumpf, president.

THANKS TO....

Bob Crosby for the Curta calculator, and the Computer Shopper article.

Myron Piker for the Scientific American article.

Emery Rogers for suggesting the Oliver interview; and Frank McConnell and Jason Zions for help with preparation.

John at CCW for making us think, once again, that we really can run a PowerMac.

Len Shustek for his donation.

Tim Swenson for the Leading Edge article.

Larry Tesler for his donation.

Susan Winn for a stellar job of transcription on the Oliver interview.

Whole Access for subsidizing KC's attendance at the Fremont/Newark Board Survival Training workshop.

NEXT ISSUE / COVER ART

The HP 2000 interview; articles, letters and queries; and, we hope! the enthralling, hour-by-hour rescue of the SDS 930.

Cover: The console and drives of the 930, photographed *in situ* at Table Mountain Observatory by Herb Sauer. Scanned on a LaCie Silverscanner II at Creative Computer Workshop, image file processed in Adobe Photoshop and printed from MS-Word for Windows on an HP LaserJet 4P.

GUIDELINES FOR DISTRIBUTION

The ANALYTICAL ENGINE is intellectual shareware. Distribution of *complete, verbatim* copies through online posting, Internet mail or news, fax, postal service or photocopying is encouraged by the Computer History Association of California.

Excerpting or brief quotation for fair use, including review or example, is also permitted, with one exception: Any material copyright to or by a third party and reprinted in the ANALYTICAL ENGINE by permission shall not be used in another periodical or context, unless the permission of the copyright holder is separately secured for the new use.

Alterations, abridgments or hacks of the ANALYTICAL ENGINE which change the intent or meaning of original content; or which contrive to bring income to any person or organization other than the Computer History Association of California; or which contrive to injure the Computer History Association of California, its officers, contributors, volunteers or members; are PROHIBITED. Reproduction of the ANALYTICAL ENGINE without its subscription coupon is abridgment in this sense.

GUIDELINES FOR SUBMISSION

The ANALYTICAL ENGINE solicits manuscripts of 750 to 2500 words on the general topic of the history of computing in, or with significant reference to, the State of California. Articles should focus on one interesting or illuminating episode and should be written for a technically literate general audience. Submissions are welcome from both members and non-members of the CHAC. Article deadlines are: July 15 for the November issue, October 15 for the February issue, January 15 for the May issue, and April 15 for the August issue.

Each author may publish a maximum of one signed article per year. This restriction does not apply to letters, queries, book reviews or interviews. Thank you for cooperating to protect diversity of voices and topics. Previously published material will be republished only in clearly

attributed quotations or citations; or when its publication in the ANALYTICAL ENGINE will bring it to the attention of a significantly broader audience; or when the original publication is materially obsolete or inaccessible.

Decision of the editors is final but copyright of all published material will remain with the author.

The preferred document file format is Microsoft Word for DOS or Windows, but almost any DOS or Macintosh word processor file will be acceptable. Submit manuscripts on DOS 5.25" or 3.5", or Mac HD (1.4) diskettes. Alternatively, please send your article as ASCII or ISO Internet mail. Please avoid submitting on paper unless absolutely necessary.

The ANALYTICAL ENGINE

Volume 2, Number 3

May 1995

ISSN 1071-6351

newsletter of the Computer History Association of California, is published four times a year at Palo Alto, California.

The CHAC is a public charity dedicated to preserving and presenting the history of electronic computing in the State of California. Donations are tax-deductible and very welcome.

Subscriptions are available with Association membership at \$25/\$35 per year individual, \$75/\$85 per year corporate or institutional, or \$15/\$25 per year low-income, student or senior. Use the coupon to subscribe, or contact the Association at:

3375 Alma Street
Suite 263
Palo Alto, CA 94306-3518 USA
FAX: +1 415 856-9914
Internet: engine@chac.org

Kip Crosby, Managing Editor

NINES-CARD

FORM LETTERS THE TELETYPE WAY

by Dave Breneman

The PC and Macintosh "CPU on every desktop" fad has, for the last 10 years, provided job security for both network administrators and their doctors. At the same time it has molded a generation of office workers who depend on graphical interfaces and rodentiform pointing devices for the completion of the simplest tasks. Rather than being set free by automation, we have become so reliant on it that something as benign as a power outage can literally bring the typical office to a halt. Although we cannot turn back the tide of progress in the case of electricity (after all, there are no wood-fired microwave ovens) we can at least deflect the impulse to obtain every new toy and gadget.

Nothing, not even a Corvette in the hands of a teenager, depreciates faster than a computer. Yet there is still much utility left in those HP 9000/300s, NCR Towers and AT&T 3B2s. How much more environmentally friendly it would be, as well, to wring the last ounce of productivity from a tried and true technological friend, rather than to consign it to the trash heap after 2 or 3 years. Consider also that IS directors are under constant pressure from management to keep budgets in line. All of these issues are addressed by a business forms solution which is low-overhead, environmentally conscious, simple, and ultimately challenging and fun for the worker. Yes, you — and your office staff — can produce impressive form letters on an ASR-33 Teletype™.

Now, many will object that a Teletype has no lower-case characters, and these people do have a point. In bulk mailings, however, this is an advantage! THINK HOW MUCH MORE EYE-CATCHING AND URGENT YOUR MESSAGE WILL APPEAR IN ALL CAPS. IT LITERALLY JUMPS OFF THE PAGE LIKE A USENET POST FROM AOL.COM! If you too find this an unexplored virtue of the teleprinter typeface and technology, read on.

Begin by assembling all the elements for your office-of-the-future-of-the-past. You will, of course, need your mailing list. Also procure a Teletype

Corporation ASR-33 teleprinter. For reasons that will shortly be apparent, your ASR-33 must be equipped with an automatic, rather than a manual, tape reader. The automatic tape reader is easily identified because its control lever has 4 positions (Manual Start, Auto, Manual Stop and Free) as distinct from the manual tape reader's 3 positions (Start, Stop and Free). Make sure that the lever is in the Auto position. For the best appearance, you should put a new ribbon in your ASR-33; most office supply stores can order such a ribbon for you at quite reasonable cost, often under \$20. You will require a roll of printer paper (still available in case lots from several business forms dealers,) a bottle of white glue, and a roll of 1" lubricated paper tape. The location of this last item is left as an exercise for the reader.

With your Teletype set up and ready to go, place it in Local mode and turn on the tape punch. Start with a series of about 10 or 12 <NUL>s, which produces a nice leader. Tear off any paper remaining above the cutting edge. This will help you position your text. Next, <LINE FEED> down until you reach the area where the recipient's address goes. This is where we begin to put technology to work for you. Type ^T^T. We all know that ^T turns off the tape reader. When you read your tape back, your Teletype will stop at this point and allow you to enter the address. Always enter two ^Ts. This is because the tape reader will frequently coast a bit after it shuts off, and if the character over the sensing pins is not a ^T, it will start up again. I like to include a ^G before the ^T, to notify the user that the ASR-33 has stopped printing and is awaiting input. Now enter a <CARRIAGE RETURN> and two <LINE FEED>s. This gets you to the salutation line. Type "DEAR ^G^T^T ;<CR><CR><LF>". This way, when the machine stops for your input, you don't need to include the ";" or any of the carriage-control characters every time you begin a new letter. Your ASR-33 adds them automatically!

Continue through the rest of the letter. Don't forget to terminate each line with two <CARRIAGE RETURN>s followed by a <LINE FEED>. This gives the carriage plenty of time to get back to its stop, especially if the carriage return dashpot cylinder vent is restricted. After your letter is complete, using a ruler, continue entering <CR>s until the 66th line of

your prototype letter (11") is even with the cutting edge. Several ^Gs and the two ^Ts will remind your operator to tear off the letter and start the next one. Now enter enough <NUL>s to get all of your tape out past the cutting edge of the tape punch, and several more <NUL>s to provide a good trailer on the end. Carefully tear the tape off. The cutting edge of the tape punch provides a nice V-shaped tear that helps guide the tape through the tape reader in the next step. Press the CLR button on the Call Control Unit panel to turn off the motor.

To create the tape loop necessary for printing multiple letters, apply a thin coating of white glue to the top of the tape leader and, being careful not to introduce any twists into the tape, press the end of the trailer on top of the leader. Remember to always press the trailer on top of the leader, as this keeps the edge of the overlap from actuating the Tight Tape bail in the tape reader as it passes by. Make sure all the perforations line up correctly.

After the glue has dried, place your tape loop in the tape reader at the point where the leader and trailer overlap. Put the ASR-33 in Local mode. It will begin printing and stop at the space provided for the recipient's address. Enter the first address followed by the Tape On command (come on, ^R, remember?). Do the same for the recipient's name in the greeting, etc. If you get really fancy, you can include lines like this:

NOW YOU, M^G^T^T, CAN BE THE FIRST
IN ^G^T^T TO OWN THIS PRODUCT!

...Then just fill in "RS. JONES" and "CHICAGO" at the appropriate pauses. If you anticipate sending these letters repeatedly, turn on the tape punch before you begin. Then you'll have all the letters, complete with names and addresses, ready to go again when necessary. Note, however, that you will probably want to use the tape punch backspace and <NUL>-out the ^Ts in your tape copy.

That's all there is to creating professional correspondence the ASR-33 way! Now form letters aren't drudge work, they're fun!

CONTENTS

Editorial: THE X-PROJECT, Part Two.....	1
PARA-TIME SHIFT	2
NEW E-MAIL ADDRESS: engine@chac.org	2
ARPANET ARTICLE WANTED	2
IN MEMORIAM: GEORGE STIBITZ	3
IN MEMORIAM: ALLEN COOMBS.....	3
RESOURCES WANTED FOR WEB PAGE.....	4
ALERT: CALCULATOR AND INSTRUMENT SHOW	4
HP's EARLY COMPUTERS, Part One, A CORE PLANE IN AMBER: An Interview with Barney Oliver	5
SAIL SPINOFFS, Les Earnest	15
THE DISCOLOURATION OF PLASTIC COMPUTER CASES, Edward Then	18
BERKELEY'S UCBVAX CHANGES JOBS, Cliff Frost.....	19
AUSTRALIAN COMPUTER MUSEUM SOCIETY ANNOUNCES INCORPORATION	20
RETROCOMPUTING MUSEUM ON THE WEB.....	21
NEWS OF MUSEUM ACTIVITY IN SWEDEN, Anders Hultman	21
COLLECTOR'S GUIDE: SECOND EDITION IN PROGRESS	21
SUN HARDWARE REFERENCE UPDATE.....	22
SPOTTER ALERT	23
MONEY, rev 2.3....	23
YOU PUBLISH! OR WE PERISH!.....	23
OVERVIEW OF BUREAUCRATIC PROCESSES.....	24
Book Review: Owen Linzmayer's THE MAC BATHROOM READER, by David T. Craig.....	24
ACQUISITIONS	26
LETTERS	26
QUERIES.....	28
ARTICLES NOTED.....	31
PUBLICATIONS RECEIVED	32
ADDRESSES OF CORRESPONDING ORGANIZATIONS	32
THANKS TO.....	33
NEXT ISSUE / COVER ART	33
GUIDELINES FOR DISTRIBUTION	33
GUIDELINES FOR SUBMISSION.....	33
NINES-CARD, Dave Breneman.....	34

SUBSCRIBE!

US\$5.00

UK£3.50

10DM

¥700

SAVE the SDS 930!

YOUR GENEROSITY will prevail!

SUBSCRIBE OR DONATE NOW!

Computer History Association of California
3375 Alma, Suite 263
Palo Alto CA 94306-3518 USA

**FIRST CLASS
TIME VALUE**

If your address label is YELLOW your subscription ends with THIS issue!