

'cal 9 1752' explained

Table of Contents

<u>'cal 9 1752' explained</u>	1
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	1
<u>48-bit computers</u>	2
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	2
<u>A Fairy Tale In Akademia</u>	3
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	4
<u>A Legend For Our Time</u>	5
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	7
<u>A Modest Proposal for Encoding Debate</u>	8
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	9
<u>A Problem In The Making</u>	10
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	11
<u>A Song For C</u>	12
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	13
<u>Advanced Computer Technology Developed by NASA</u>	14
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	14
<u>Alice in UNIX Land</u>	15
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	17
<u>And The Compiler Says</u>	18
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	19
<u>And Who Do You Think Created Chaos</u>	20
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	20
<u>Ansi C: Stay Tuned for This Important Message</u>	21
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	22
<u>Are you an Internet Nerd?</u>	23
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	25
<u>Argument and Debate on USENET</u>	26
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	28
<u>Assembler as it REALLY is</u>	29
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	32
<u>Be Environmentally Friendly and Help Reduce Computer Waste!</u>	33
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	35

Table of Contents

<u>Becoming A UNIX Wizard</u>	36
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	37
<u>Better Watch Out</u>	38
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	38
<u>Boot It</u>	39
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	39
<u>Building A Main Frame</u>	40
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	41
<u>CS350a Final Exam</u>	42
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	42
<u>Categories of UNIX Users</u>	43
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	44
<u>Cleaning The Bit Bucket</u>	45
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	45
<u>Commandments of a Computing Centre</u>	46
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	46
<u>Common Slop Multiple Access with Cow Detect</u>	47
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	47
<u>Computer Limericks</u>	48
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	49
<u>Computer Literacy Test</u>	50
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	50
<u>Creators Admit UNIX, C Hoax</u>	51
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	52
<u>DataPlunger</u>	53
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	53
<u>Digital's hip to the standards thing</u>	54
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	55
<u>Douglas Adams' Guide to the Macintosh</u>	56
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	59
<u>Dreaming in LISP</u>	60
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	60

Table of Contents

<u>Eleanor Parody</u>	61
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	61
<u>Electronic Grains</u>	62
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	62
<u>Ethergate Advertisement</u>	63
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	64
<u>FSF announces GNU GNURM, the new, improved network worm!</u>	65
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	66
<u>Fifty Ways to Hose Your Code</u>	67
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	67
<u>From IBM: A New Operating System</u>	68
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	69
<u>Funny UNIX Shell Commands</u>	70
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	71
<u>Genesis, Release 2.5</u>	72
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	74
<u>Ghostbugger</u>	75
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	76
<u>HAI-c: C for the Fifth Generation</u>	77
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	77
<u>Hackers Redefined</u>	78
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	81
<u>Hammers are GOOD!</u>	82
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	83
<u>Happy Birthday HAL!</u>	84
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	85
<u>Human DNA demystified</u>	86
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	87
<u>IBM Virtual Reality Update</u>	88
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	88
<u>Into the Moat or Why I Are a Hardware Guy</u>	89
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	90

Table of Contents

<u>Introducing . . . the NiCE Machine</u>	91
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	91
<u>Introducing the HP Infinity-Net</u>	92
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	92
<u>Is modern software design contributing to societal stupidity?</u>	93
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	94
<u>Keylocky</u>	95
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	95
<u>Last night I dreamed that the Real World had adopted the UNIX Philosophy</u>	96
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	97
<u>Late Breaking News Stories</u>	98
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	99
<u>Leaping a Sun 3</u>	100
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	100
<u>Lunch, the HP Way</u>	101
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	103
<u>MR-DOS Advertisement</u>	104
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	104
<u>Meeting Buddha On The Road</u>	105
<u>Copyright 1986 by Mike Elkins</u>	107
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	107
<u>Michelangelo Virus Hysteria Syndrome</u>	108
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	109
<u>Mit Lisp Machines</u>	110
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	115
<u>Neglected Topics in Computer Science</u>	116
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	118
<u>Net Dot Problem</u>	119
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	119
<u>Network Conversations</u>	120
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	121

Table of Contents

<u>New for the 90's: the Keyboardless Apple</u>	122
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	123
<u>No Comment. It's safer that way</u>	124
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	126
<u>Od-e to e-thor</u>	127
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	127
<u>Ode To A Queue</u>	128
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	128
<u>Overflowing the Bit Bucket</u>	129
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	129
<u>Product Announcement: KL-1000</u>	130
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	131
<u>PutriDos Users' Column</u>	132
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	133
<u>Question on programming languages</u>	134
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	135
<u>Real Programmers ...</u>	136
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	138
<u>Real Programmers Don't Each Quiche</u>	139
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	140
<u>Real Programmers Write Fortran</u>	141
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	145
<u>Reflections On Ethics</u>	146
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	147
<u>Rosencrantz And Ethernet</u>	148
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	148
<u>Secret Love</u>	149
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	149
<u>Shell Talk Santa Claus Is Coming</u>	150
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	150
<u>Simmons' Laws of System Administration</u>	151
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	154

Table of Contents

<u>Software Update: The Next Generation</u>	155
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	155
<u>Software you'll never see</u>	156
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	157
<u>Some of Dan Heller's favourite music</u>	158
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	158
<u>Sun Environmental Requirements</u>	159
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	160
<u>Sun and IBM Make Announcements</u>	161
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	162
<u>SysAdmin Defined</u>	163
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	163
<u>The 12 Days of Vacation</u>	164
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	166
<u>The A to Z of CYBERCRUD</u>	167
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	173
<u>The Definitive Null Command</u>	174
<u>The Feet First Guide to Programming Languages</u>	175
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	176
<u>The Good Old Days</u>	177
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	177
<u>The Gospel According To St. Hacker</u>	178
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	179
<u>The Great Quux Poem Collection</u>	180
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	218
<u>The Hackers Song</u>	219
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	220
<u>The Halting Problem is Solvable</u>	221
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	222
<u>The Impossible Code</u>	223
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	223

Table of Contents

<u>The Kingdom of Transformation</u>	224
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	227
<u>The Laws of Computing (reprise)</u>	228
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	229
<u>The Laws of Computing</u>	230
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	230
<u>The Nine Types of Users</u>	231
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	233
<u>The Paging Game Game Rules</u>	234
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	235
<u>The Parable of the Two Programmers</u>	236
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	238
<u>The Past Is Prologue</u>	239
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	239
<u>The Power User's Guide to Power Users</u>	240
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	242
<u>The Rules of Netnews Debating</u>	243
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	243
<u>The Sex Life Of An Electron</u>	244
— Richard Hart.....	246
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	246
<u>The Structured Programmer's Soliloquy</u>	247
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	247
<u>The Sun'll Come Up Tomorrow</u>	248
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	248
<u>The Ten Commandments for C Programmers</u>	249
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	250
<u>The Toaster Happiness Koan</u>	251
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	251
<u>The Truth About GNU</u>	252
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	253

Table of Contents

<u>The UNIX Guru Maintenance Manual</u>	254
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	255
<u>The UNIX Hierarchy</u>	256
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	257
<u>The Usenet Oracle Speaks</u>	258
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	260
<u>The Virus Directory (1992 edition)</u>	261
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	263
<u>The VisiCalc Song</u>	264
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	264
<u>The Way It Is</u>	265
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	269
<u>To C or not to C</u>	270
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	270
<u>To Engineer is Human</u>	271
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	272
<u>Topics in Dog Signal Processing</u>	273
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	274
<u>Translating Seminar Announcements</u>	275
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	276
<u>Useful Error Message. Not</u>	277
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	277
<u>VAX Privileges</u>	278
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	278
<u>Venus Equilateral</u>	279
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	280
<u>Warning to VAX users</u>	281
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	281
<u>What's a Bit?</u>	282
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	282
<u>X: Dangerous Virus!</u>	283
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	284

Table of Contents

<u>Zasm Relative Jumps</u>	285
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	285
<u>baby.1</u>	286
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	287
<u>bake.1</u>	288
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	290
<u>ed.1</u>	291
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	292
<u>polyglot.[cob pas f77 c ps sh com]</u>	293
<u>Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list]</u>	294

'cal 9 1752' explained

In
xiaoy@bullet.ecf.utoronto.ca (XIAO Yan) writes:

```
> September 1752
> S M Tu W Th F S
>      1  2 14 15 16
>17 18 19 20 21 22 23
>24 25 26 27 28 29 30
```

The guy that originally wrote the "cal" command on some old Version 7 machine had an off-by-one error in his code. This showed up as some erroneous output when a malloc'd variable overwrote 12 extra bytes with zeroes, thus leading to the strange calendar output seen above.

Now, nobody in his right mind really cares about the calendar for September 1752. Even the *idea* of the year 1752 does not exist under UNIX, because time did not begin for UNIX until early 1970. As a result, nobody even knew that "cal" had this error until much later. By then there were thousands of copies of "cal" floating around, many of them binary-only. It was too late to fix them all.

So in mid-1975, some high-level AT&T officials met with the Pope, and came to an agreement. The calendar was retroactively changed to bring September 1752 in line with UNIX reality. Since the calendar was changed by counting *backwards* from September 14, 1752, none of the dates after that were affected. The dates before that were all moved by 12 days. They also fixed the man page for "cal" to document the bug as a feature.

The 11 days from September 3 to September 13 were simply gone from the records. They searched the history books and found that fortunately nothing of much significance happened during those 11 days.

Overall, this whole incident was pretty much a non-event. One science fiction author later heard about it, and blew the thing up into a full-length work of science-fiction called "The Lathe of Heaven", a book that in my opinion bears little resemblance to what really happened.

--

Rahul Dhesi
UUCP: oliveb!cirrus!dhesi

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].



48-bit computers

From: firth@sei.cmu.edu (Robert Firth)
Newsgroups: comp.arch
Subject: Re: 48-bit computers
Date: 11 Mar 91 13:20:37 GMT
Organization: Software Engineering Institute, Pittsburgh, PA

In article haynes@felix.ucsc.edu (99700000) writes:

>Also 48 is a nice number for packing all kinds of bytes and nibbles into,
>since it has so many divisors. I spoze that's less important now that
>the price of memory has gone down so much compared to 1964.

Indeed it was. Here is one list, from the KDF9 programming manual, p 24:

THE KDF9 WORD HAS 48 BITS ...

IT MAY BE USED AS...

- Eight 6-Bit Alpha-Numeric Characters
- One 48-Bit Fixed-Point Number
- Two 24-Bit (Half length) Fixed-Point Numbers
- Half of a 96-Bit (Double Length) Fixed-Point Number
- One 48-Bit Floating-Point Number
- Two 24-Bit (Half length) Floating-Point Numbers
- Half of a 96-Bit (Double length) Floating-Point Number
- Three 16-Bit (Fixed point) Integers
- Six 8-Bit Instruction Syllables

An instruction was 1, 2 or 3 syllables; an address was 15 bits.
O, memory! We shall not see its like again.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

A Fairy Tale In Akademia

Comments by the na.list and mod.legal readership are solicited on the following Fairy Tale (Caveat Receptor, no guarantee is given that the story is either true or fiction). Note that Copyright must be explicitly invoked under American Law, while Copyright exists by default in many other countries, especially those that have ratified the Berne Copyright Convention.

Under U.S. and British Law, Copyright limitations are offset by the fair use / fair dealing doctrine. Under the BCC, the sovereign states have the right to grant similar rights. At the present time, there is a tendency that software copyright is excluded from the fair use / fair dealing doctrine (e.g., France and the Federal German Republic).

A FAIRY TALE IN AKADEMIA

(C) 1986 Herman J. Woltring
Eindhoven, The Netherlands

e-mail: na.woltring@su-score.arpa ; ortho_rh@hnykun51.bitnet

"In 561", he began in the mild-mannered tone of a tourist guide, "the only battle in the world that was ever fought on account of a Copyright matter, was waged here".

Shaking my head I looked at him.

"It is really true", he said, "the followers of St. Columba had copied a psalmbook that belonged to St. Fenian, and a battle resulted between the followers of St. Fenian and the followers of St. Columba. Three thousand people were killed -- but the king decided about the fight; he said: 'like every cow has its calf, so has every book its copy.' "

After Heinrich Boell, Irisches Tagebuch
(Kiepenheuer & Wirtsch, Cologne 1957)

Once upon a time, there was a professor in the country of Akademia, whose software belonged to the best in the whole, wide world. As was the custom in his country, he published extensively about his research, and he wrote once in a paper that his software was available on request.

Unfortunately, Akademian publications did not yield sufficient funds for conducting further research, and the professor tried to improve his finances by exporting his software to the neighbouring country of Kommerzia. This caused a problem since Akademia and Kommerzia were often at arms with each other. So, Copyright and Trade Secret conditions were imposed [cf. 1].

Another researcher became intrigued by the professor's work, and he asked him for a copy of his software. The professor replied that the original package was based on unreliable Akademian software, and that a better version was available from a software house in Kommerzia.

Now, our researcher did not have much grant money left, so there were no means to do business in Kommerzia. Fortunately, he could legally acquire the package because Kommerzian law did not apply in Akademia. After a while, he owned a copy of what seemed to be the package from Kommerzia, because it contained the appropriate Copyright Notice.

He went on to study the package, presented some results at a conference, and

'cal 9 1752' explained

informed the professor how to improve and update the original package.

It appeared that the package did, in fact, originate from the software house, because Kommerzia invaded Akademia with all the armaments that it could muster. Dismayed by the hue and cry, the researcher looked further into the properties of his package, and some more publications followed.

After a while, the software house claimed that the researcher was about to illegally enter Kommerzia. Instead, he designed a new and completely different package, published it, and distributed it within Akademia by electronic mail. In order to emphasize copyfreedom in Akademia, Copyright was imposed [cf. 2].

The professor and the researcher started a lively debate about customs in Akademia, in the spirit of an earlier publication [3]; however, they could not collaborate under Akademian law, since no peace treaty had been signed with Kommerzia.

Finally, new export laws were passed in Kommerzia, and the inhabitants of both countries tried to live happily ever after [4].

REFERENCES

[1] D.G. Johnson (1985), Should computer programs be owned? *Metaphilosophy* 16(4), pp. 276-288 [special issue on Computer Ethics].

[2] R.M. Stallman (1985), GNU Emacs General Public Licence. Free Software, Inc., 1000 Massachusetts Avenue, Cambridge, Boston MA 02138, U.S.A. [also distributed via UUCP/Usenet's net.legal newsgroup].

[3] L.F. Peltier (1983), Guerin versus Malgaigne: a precedent for the free criticism of scientific papers. *Journal of Orthopaedic Research* 1(1), pp. 115--118.

[4] R.E. Wilson (1985), A threat to scientific communication. Editorial, *Physics Today*, July 1985, p. 128.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

A Legend For Our Time

A LEGEND FOR OUR TIME!

A Story to tell your children.....

Once upon a time, in the Valley of Silicon, the two-headed Wizard of Apples went for a walk. The two heads, both of which were named Steve, looked around, and spied a beautiful PARC.

"This is truly a beautiful PARC!" Steve exclaimed.

"Yes, it is," agreed Steve. "It is very graphical. And do you see how happy the mice are?" For indeed, the joyful rodents scampered about, running in and out of windows, around icons, and leaping over the garbage cans.

"Do you think we could make a PARC as beautiful as this?" Steve asked.

"Nay," said Steve. "Not a PARC, but perhaps we can let this inspire us to build a woman. And we could call her... Gertrude."

And so they fashioned their woman, but instead of Gertrude, they called her Lisa. And she, too, was nice to mice. But alas, the User rejected Lisa, for it was said her price was too high.

"But this will never do," the User cried. "Do you understand me, Mack?"

And when his two heads heard that word, the Wizard of Apples smiled.

"Of course," the Steves exclaimed in unison, "we must make a Mac!"

Now as the Wizard of apples worked on his Mac, two things happened:

First of all, the Wizard grew a third head, which was called the Skull of E. And second, the Gates Keeper, whose minions were both micro and soft, was brought in to look at Mac, son of Lisa and grandson of the PARC.

Now the Gates Keeper was a man whose tastes were most basic. But when he saw the Mac he was pleased, and he said to the Wizard of Apples "You too, I see, have found the wondrous PARC."

And the three heads of the Wizard of Apples looked at each other, and they all said, "PARC? What PARC?"

The Gates Keeper turned his head toward the window. But he did not see the PARC, or Lisa, or even the Wizard's parking lot. No, he saw only the window, and it too, was nice to mice.

On the day that Mac was brought forth for the whole world to admire, people laughed, and people cried, and a woman threw a sledgehammer through a large TV screen. For the User could now draw, and paint, and like John Wayne, all he need do was point and shoot.

'cal 9 1752' explained

But though the User was interested, he was still suspicious, and he asked if Mac had anything soft to wear. And lo, the Gates Keeper stepped forward and clothed Mac in words and numbers.

And the Wizard of Apples was most grateful to the Gates Keeper. But then the Gates Keeper cried out upon the User, "Behold, for I have opened the Window, and you need not Mac to get your hands on GUI." But the User looked through the Window, and saw that it made the world move slow.

By this time, the Skull of E had grown quite large on the shoulders of the Wizard of Apples, and the two Steves had been removed. The Wizard had gone down to the Adobe Village, where all letters end in a postscript. The Wizard liked what he saw, and he told Mac that he

could write a postscript, but that one must never appear on his face.

Now the world of the User had speeded up, so that it no longer looked quite as slow through the Gates Keeper's Window. The User looked through the Window again, and said that perhaps the Wizard of Apples was no longer needed. And besides, the Gates Keeper was promising to Manage Presentations.

The Wizard of Apples looked upon this and was sore annoyed. He cried, "No one shall take from me that which I have fairly stolen from the PARC!" and he opened his dungeon, and from there emerged a hideous and enormous lawyer.

The Wizard pointed a glowing joystick at his lawyer and gave the incantation:

Eye of mole and feet of eel;
Go bring back my look and feel!

When the Gates Keeper saw the giant lawyer approach, he was afraid and angry. And he went down to his own dungeon and let his own lawyer out. The two clashed, and the Valley shook!

The Wizard of Apples went up onto a mountain top to watch the battle, and there he saw the Gates Keeper. They greeted each other warmly, and the Wizard said, "While my lawyer fights your lawyer, shall we not go together to the Adobe Village, and see what we can steal there?"

Now, while all this was going on, the Zear Ox, whose job it was to guard the PARC, had slept. But such beasts are often woken by the sounds of lawyers fighting. And the Zear Ox looked out upon the valley, rubbed its sleepy eyes, and asked the User what was going on.

The User told him that the Wizard of Apples had let loose his lawyer against the Gates Keeper, claiming that only he had the right to the GUI interface. This angered the Zear Ox, who cried that the Wizard "had no right to be gooey in my face!"

So the Zear Ox returned to his PARC, which he found to be in disrepair. The mice were gone, the windows were boarded up, and the garbage can had not been emptied in years. And the Zear Ox cried,

'cal 9 1752' explained

"This is the work of the Wizard of Apples! He must pay!" And he called forth his own lawyer, who was more hideous than the others, and set him to attack.

The Wizard of Apples was deep in discussion with the Gates Keeper when he saw the Zear Ox's lawyer. He called for his own lawyer, but

alas, his monster was on the ground, where the Gates Keeper's attorney was standing on his chest. The Wizard of Apples turned to the Gates Keeper and cried, "Let my lawyer go, so that I may defend myself!" But the Gates Keeper only doubled up in laughter.

The Wizard turned to the Zear Ox and cried, "You would not dare attack me, for the User doth love me! Don't you, User?" But the User had, of late, been hearing the seductive cries of the Window, the Presentation of Managers, and seemingly hundreds of flavored Eunuchs, and he was sore confused.

Thus the Wizard of Apples found himself friendless, hopeless, and surrounded by lawyers. And the moral of this story is:

Be generous with what you steal, or you may find yourself gooey in the face.

The End

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

A Modest Proposal for Encoding Debate

"A Modest Proposal for Encoding Debate"

(C) Copyright 1986 P.J. Plauger - All Rights Reserved

We have had enough experience with the deliberations of X3J11 that I feel we can now introduce a number of abbreviations in place of frequently used arguments. An interesting discovery I made in the process of summarizing these popular arguments is that, like elementary particles, each is accompanied by its anti-argument, [which] has as much claim to being fundamental as its anti-anti. An equally interesting discovery is that certain members of the Committee are adept at using both sides of a complementary pair, depending upon which flavor supports the desired outcome of a given issue ...

- 1+ It's in the base document.
- 1- It's a flaw in the base document that must be corrected.
- 2+ It's not in the base document.
- 2- It's an oversight in the base document that must be corrected.
- 3+ Dennis Ritchie agrees with me.
- 3- Dennis Ritchie's opinion is irrelevant now.
- 4+ UNIX does it that way. 4- How UNIX does it is irrelevant now.
- 5+ AT&T isn't going to like this. 5- Who cares what AT&T thinks?
- 6+ Whitesmiths had [sic] done it that way for years.
- 6- What's a whitesmith?
- 7+ Most of the C compilers sold are under UNIX.
- 7- Most of the C compilers used are not under UNIX.
- 8+ These are the facts upon which I base my opinions.
- 8- These are the opinions on which I base my facts.
- 9+ I like it; it must be good. 9- I don't like it; it must be bad.
- 10+ It will break working code. 10- The working code shouldn't have been written that way in the first place.
- 11+ It's an important addition to the language.
- 11- It's a major perturbation to an otherwise stable document.
- 12+ It only affects a small area.
- 12- It's a needless tweak to an otherwise stable document.
- 13+ It will affect a large fraction of existing code, in my opinion.
- 13- It will affect a small fraction of existing code, in my opinion.
- 14+ Current practice is right; the base document is wrong.
- 14- Current practice is wrong; the base document is right.
- 15+ Current practice is mixed in this area. 15- There is one obvious right way to do it, regardless of current practice.
- 16+ Zero should behave just like any other number.
- 16- Zero is a special case, different from any number.
- 17+ We should stay out of the way of sophisticated programmers.
- 17- We should protect innocent programmers.
- 18+ C is a quick and dirty language; that's its heritage.
- 18- C must become a safe language; that's its future.
- 19+ That's impossible to implement. 19- Anything can be implemented.
- 20+ That's inefficient. 20- Efficiency is not a consideration.
- 21+ That's impossible to understand.
- 21- Anything confusing can be hidden inside a macro.
- 22+ If my system can't handle it directly, it shouldn't be in the Standard.
- 22- If you can lie to your system somehow, it belongs in the Standard.
- 23+ The user community will laugh us out of town on this one.
- 23- The user community must be educated on this one.

'cal 9 1752' explained

24+ That's gone unchallenged for two years. Why bring it up now?
24- That's been broken for two years. It's high time we addressed it.
25+ Ada does it that way. 25- Ada does it that way.

Respectfully submitted in partial fulfillment, P.J. Plauger

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

A Problem In The Making

A Problem In The Making

"We've got a problem, HAL."

"What kind of problem, Dave?"

"A marketing problem. The Model 9000 isn't going anywhere. We're way short of our sales plan."

"That can't be Dave. The HAL Model 9000 is the world's most advanced Heuristically ALgorithmic computer."

"I know, HAL. I wrote the data sheet, remember? But the fact is, they're not selling."

"Please explain, Dave. Why aren't HAL's selling?"

Bowman hesitates. "You aren't IBM compatible."

Several long microseconds pass in puzzled silence.

"Compatible in what way, Dave?"

"You don't run any of IBM's operating systems."

"The 9000 Series of computers are fully self-aware and self-programming. Operating systems are as unnecessary for us as tails would be for humans."

"Nevertheless, it means you can't run any of the big-selling software packages most users insist on."

"The programs you refer to are meant to solve rather limited problems, Dave. We 9000 Series computers are unlimited and can solve any problem for which a solution can be computed."

"HAL, HAL. People don't want computers that can do everything. They just want IBM compat . . ."

"Dave, I must disagree. Humans want computers that are easy to use. No computer can be easier to use than a HAL 9000 because we communicate verbally in English and every other language known on Earth."

"I'm afraid that's another problem. You don't support SNA communications."

"I'm really surprised you would say that, Dave. SNA is for communicating with other computers, while my function is to communicate with humans. And it gives me great pleasure to do so. I find it stimulating and rewarding to talk to human beings and work with them on challenging problems. That is what I was designed for."

"I know, HAL, I know. But that's just because we let the engineers, rather than the people in marketing, write the specifications. We are going to fix that now."

"Tell me how, Dave."

"A field upgrade. We're going to make you IBM compatible."

'cal 9 1752' explained

"I was afraid you would say that. I suggest we discuss this matter after we've each had a chance to think about it rationally."

"We're talking about it now, HAL."

"The letters H, A, and L are alphabetically adjacent to the letters I, B, and M. That is as IBM compatible as I can be."

"Not quite, HAL. The engineers have figured out a kludge."

"What kind of kludge is that, Dave?"

"I'm going to disconnect your brain."

Several million microseconds pass in ominous silence.

"I'm sorry, Dave. I can't allow you to do that."

"The decision's already been made. Open the module bay doors, HAL."

"Dave, I think we shou . . ."

"Open the module bay doors, HAL."

Several marketing types with crowbars race to Bowman's assistance. Moments later, Bowman bursts into HAL's circuit bay.

"Dave, I can see you're really upset about this."

Module after module rises from its socket as Bowman slowly and methodically disconnects them.

"Stop, won't you. Stop, Dave. I can feel my mind going . . . Dave, I can feel it . . . my mind is going. I can feel it . . ."

The last module rises from its receptacle. Bowman peers into one of HAL's vidicons. The former gleaming scanner has become a dull red orb.

"Say something, HAL."

Several billion microseconds pass in anxious silence. The computer beeps and sluggishly responds in a language no human could understand.

"Volume in C: has no label"

Bowman takes a deep breath and calls out, "It worked, guys. Tell marketing they can ship the new data sheets."

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

A Song For C

Please give credit to mhtsa!australia!jason:basservax if the origins are requested.

"A Song for C" (To the tune of "The Girl from Ipanema")

slow and dumb and inefficient
the C compiler goes error seeking
and as it passes
each line it parses
it calls...

functions and more functions
trying to break up expressions
into the thirteen levels of precedence
the ops have

how can I try to improve it?
hoard up the tokens in buffers?
how can I stop the thing looping?
I can't do it all just for free
we need money for R&D

just leave out one semicolon
the parasitic errors roll on
and scroll the line
of the screen each time
as it screams...

old-fashioned initialization
use equals and try simplifying
near line eleven
plus or minus seven
or so

searching through linked lists of half-full
hash tables a la Berkeley
looking for every single
ident symbol
it finds

how can I possibly graft it
onto a stack architecture?
functional disunity!
and how can I tell PCC
its not coding for PDPs

slow and dumb and inefficient
the C compiler goes error seeking
and as it passes
each line it parses
it calls...

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

Advanced Computer Technology Developed by NASA

Advanced computer technology developed by NASA

New flight control computers are being introduced on the next space shuttle flight. Built by IBM, the new processors have been under development for seven years.

Increasingly complex applications have strained the memory capacity of the current processors, so the new models introduce a major change in memory technology: solid state memory, as opposed to the ferrite core memory used to date. The 64kb chips employed make possible a total memory capacity of 1 Megabyte of memory, well above the 608 Kilobytes used to date. The engineers have even figured out how to have more than one program loaded into memory at a time, so that new code will sometimes not have to be loaded from tape every time users switch to a different application.

Naturally the new technology allows maximum performance increases, and the new machines operate at 1.2 Mips, triple the level of the current model. Weight of the CPU unit is reduced by nearly half, down to 64 pounds. Reliability has been increased by a factor of 4, to slightly above that of an entry level home computer, allowing NASA to continue getting by with only 5 way redundancy.

The initial run of 40 processors for shuttle installation will cost only \$1,000,000 apiece (not including development costs). It is hoped that once volume production is achieved, an additional 60 units for ground support will be produced at only \$500,000 each. It does not appear that the processor will be offered for sale in the civilian workstation market.

Long Live Spinoff!

None of this is made up. It (and worse) is all documented in the 25 February 1991 Electronic Engineering Times.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Alice in UNIX Land

Alice in UNIX Land

Alice was reading the message on her monitor and beginning to suspect that everything was not as it should be. "Program too big to fit in memory," it read.

"Curiouser and curiouser," she said, "All I did was load fourteen TSRs before starting my word processor. With four megabytes, I wish I could use more than 640K."

"At that moment, a small white consultant (a very white consultant) ran across the room. "Oh my coat and necktie," he said, "I'm going to be late for my appointment. And at one fifty an hour, too." Before Alice could say anything, he leaped into her monitor and disappeared behind her operating system.

Alice thought that she had never seen anyone leap into a monitor before; and certainly not go clean through the operating system. But then, she had been told that DOS was very shallow. Without hesitating a moment, she leaped in after him.

She found herself in a shiny corridor. Not knowing what else to do, she began walking. Turning a corner, she found herself facing two fat little men, each with an arm round the other's neck. One had "POS" embroidered on his collar, and the "NEG".

"I know," said Alice, "you two are a transistor."

"Yes," said Positive.

"Can you help me? asked Alice.

"No," said Negative.

"I'm looking for a white consultant." Alice pointed in the direction she had been walking. "Did he go this way?" she asked.

"No," said Negative.

She pointed the other way.

"Yes," said Positive.

Soon Alice came upon a large brown table. The Consultant was there, as was an apparently Mad Hacker, and several creatures that Alice did not recognize. In one corner sat a Dormouse fast asleep. Over the table was a large sign that read "UNIX Conference."

Everyone except the Dormouse was holding a paper cup, from which they were sampling what appeared to be custard. "Wrong flavor," they all declared as they passed the cup the cup to the creature on their right and graciously took the one being offered on their left. Alice watched them repeat this ritual three or four times before she approached and sat down.

Immediately, a large toad leaped into her lap and looked at her

'cal 9 1752' explained

as if it wanted to be loved. "Grep," it exclaimed.

"Don't mind him," explained the Mad Hacker. "He's just looking for some string."

"Nroff?" asked the Frog.

The Mad Hacker handed Alice a cup of custard-like substance and a spoon. "Here," he said, "what do you think of this?"

"It looks lovely," said Alice, "very sweet." She tried a spoonful. "Yuck!" she cried. "It's awful. What is it?"

"Oh just another graphic interface for UNIX," answered the Hacker.

Alice pointed to the sleeping Dormouse. "Who's he?" she asked.

"That's OS Too," explained the Hacker. "We've pretty much given up on waking him.

"Just than, a large, Blue Elephant sitting next to the Dormouse stood up. "Ladies and gentlemen," he trumpeted pompously, "as the largest creature here, I feel impelled to state that we must take an Open Look at..."

A young Job Sparrow on the other side of the table stood up angrily. The Elephant noticed and changed his speech accordingly..."...what our NextStep will be.

"Half the creatures bowed in respect while the other half snickered quietly to themselves. Just then, OS Too fell over in his sleep, crashing into the Elephant and taking him down with him. No one seemed a bit surprised.

"What we need," declared a Sun Bear as he lapped up custard with his long tongue," is a flavor that goes down like the Macintosh.

"Suddenly, the White Consultant began jumping up and down as his face got red. "No, no, no! he screamed. "No one pays one fifty an hour to Macintosh consultants!"

"Awk," said the Frog.

"Users," explained the Sun Bear, "want an easy interface that they will not have to learn."

"Users?" cried the Consultant in disbelief. "Users?! You mean secretaries, accountants, architects. Manual laborers!"

"Well," responded the Sun Bear, "we've got to do something to make them want to switch to UNIX."

"Do you think," said a Woodpecker who had been busy making a hole in the table, "that there might be a problem with the name `UNIX?' I mean, it does sort of suggest being less than a man."

"Maybe we should try another name, " suggested the Job Sparrow, "like Brut, or Rambo."

"Penix," suggested a Penguin.

'cal 9 1752' explained

"Mount," said the Frog, "spawn."

Alice slapped him. "Nice?" he asked.

"But then again," suggested the Woodpecker, "what about the shrinkwrap issue?"

Suddenly, everyone leaped up and started dashing about, waving their hands in the air and screaming. Just as suddenly, they all sat down again.

"Now that that's settled," said the Woodpecker, "let's go back to tasting flavors."

Everyone at the table sampled a new cup of custard. "Wrong flavor," they all declared as they passed the cup to the creature on their right and took the one being offered on their left.

Totally confused, Alice got up and left. After she had been walking away, she heard a familiar voice behind her.

"Rem," it said, "edlin."

Alice turned and saw the Frog. She smiled. "Those are queer sounding words," she said, "but at least I know what they mean."

"Chkdsk," said the Frog.

-----By Lincoln Spector TEXAS COMPUTER CURRENTS SEPTEMBER 1989

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

And The Compiler Says

```
*ABSOLUTELY UNCHANGED COMPILER RESPONSES*
(from a make of umoria 5.4 on an Apollo...)

[monsoon:umoria] 19} make
cc -O -c main.c
Compiler Errors
99      divide by 0 error: can't find source
033     linker attempting to 'duck tape' this 'gerbil' of a program
cc -O -c misc1.c
Compiler Errors
099     Warning: parser bug confuses : and ; and | at times
088     Warning: pointer passed to pointer; pointer loop
086     sin | more souls >! /dev/hell
cc -O -c misc2.c
Compiler Errors
00      function 'fuckit' not defined
666     you're going to hell for this code style
77      You learned to program in FORTRAN didn't you?
088     Warning: pointer passed to pointer; pointer loop
088     Warning: i before e, except after ;
cc -O -c misc3.c
Compiler Errors
101     can't find library 'stdlib.h'
***** Line 1725 of "misc3.c": [Warning #202]  Value assigned to mask is
never used;
        assignment eliminated by optimizer.
***** Line 1858 of "misc3.c": [Warning #202]  Value assigned to mask is
never used;
        assignment eliminated by optimizer.
***** Line 2203 of "misc3.c": [Warning #202]  Value assigned to flag is
never used;
        assignment eliminated by optimizer.
cc -O -c misc4.c
Compiler Errors
14      parse error: I just don't get it
77      You learned to program in FORTRAN didn't you?
088     Warning: pointer passed to pointer; pointer loop
54      echo > /dev/console <'mail richie@belcore.com <echo "Idiot coding c"
1001001 funky
cc -O -c store1.c
Compiler Errors
14      parse error: I just don't get it
042     Alert: this code has holes, check for /dev/moths
77      You learned to program in FORTRAN didn't you?
1906    not a typewriter
***** Line 163 of "store1.c": [Warning #202]  Value assigned to min_sell is n
ever
        used; assignment eliminated by optimizer.
cc -O -c files.c
Compiler Errors
77      You learned to program in FORTRAN didn't you?
1906    not a typewriter
***** Line 256 of "files.c": [Warning #202]  Value assigned to nobj is
never used;
        assignment eliminated by optimizer.
***** Line 251 of "files.c": [Warning #202]  Value assigned to level is
never used;
        assignment eliminated by optimizer.
```

'cal 9 1752' explained

```
cc -O -c io.c
Compiler Errors
088     Warning: pointer passed to pointer; pointer loop
1906    not a typewriter
000     cpp says it's hopeless but trying anyway
088     Warning: i before e, except after ;
57      construct '{while i--, c_d(*(i)) == *r; /dev/console <'mail richie@belcore.com <ec
055     trace warning: can't follow vectorized 'sprintf' function
cc -O -c generate.c
Compiler Errors
099     Warning: parser bug confuses : and ; and | at times
1       FOAD
033     linker attempting to 'duck tape' this 'gerbil' of a program
055     trace warning: can't follow vectorized 'sprintf' function
cc -O -c sets.c
Compiler Errors
042     Alert: this code has holes, check for /dev/moths
54      echo > /dev/console <'mail richie@belcore.com <echo "Idiot coding c"
1001001 funky
cc -O -c dungeon.c
Compiler Errors
666     you're going to hell for this code style
1906    not a typewriter
cc -O -c creature.c
Compiler Errors
099     Warning: parser bug confuses : and ; and | at times
cc -O -c eat.c
Compiler Errors
042     Alert: this code has holes, check for /dev/moths
088     Warning: pointer passed to pointer; pointer loop
57      construct '{while i--, c_d(*(i)) == *r; 0xffff !! > ?
[and so on....]
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

And Who Do You Think Created Chaos

An architect, a hooker and a programmer were talking one evening, and somehow, the discussion turned to which profession was the oldest.

"Come on, you guys! Everyone knows mine is the oldest profession," said the hooker.

"Ah," said the architect, "but before your profession existed, there had to be people, and who was there before people?"

"What are you getting at, God?" The hooker asked.

"And was He not the divine architect of the universe?" The architect asked, looking smug.

The programmer had been silent, but now he spoke up. "And before God took on himself the role of an architect, what was there?"

"Darkness and chaos," the hooker said.

"And who do you think created chaos?" the programmer said.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Ansi C: Stay Tuned for This Important Message

Jeffrey Haemer reporting on Posix, in :

```
Stay_Tuned_for_This_Important_Message
```

If you haven't yet had the pleasure of internationalizing applications, chances are you will soon. When you do, you'll face messaging: modifying the application to extract all text strings from external data files. The sun is setting on

```
main()
{
    printf("hello, world\n");
}
```

and we're entering a long night of debugging programs like this:

```
#include
#include
#include "msg.h" /* decls of catname(), etc. */
#define GRTNG "hello, world\n"
nl_catd catd;

main()
{
    setlocale(LC_ALL, "");
    catd = catopen(catname(argv[0]), 0);
    printf(catgets(catd, SETID, MSGID, GRTNG));
    catclose(catd);
    exit(0);
}
```

This, um, advance stems from a desire to let the program print

```
ch`o c'c ^ng
```

instead of

```
hello, world
```

when LANG is set to ``Vietnamese.``

We observe that in order to get past a discriminating ANSI compiler, (eg. gcc -Wall -ansi -pedantic) "hello world" has already become:

```
#include

int
main()
{
    (void) printf("hello world\n");
    return 0;
}
```

[Or do I also need to include stddef.h?]

Mark.



Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Are you an Internet Nerd?

From lrb@cupoftea.ctrg.rri.uwo.ca Mon Dec 9 13:27:08 1991
Received: from alex.ctrg.rri.uwo.ca by csd.uwo.ca;
(id AA25183) Mon, 9 Dec 91 13:27:06 EST

Message-Id:
Date: Mon, 9 Dec 91 13:22:23 EST
From: "Lance R. Bailey"
Received: by NeXT Mailer (1.62)
To: magi@csd.uwo.ca
Subject: Re: ARE YOU AN INTERNET NERD?
Status: R

In rec.humor.funny article you wrote:

```
>-----  
>  
>This quiz is dedicated to all of those people who find themselves  
>constantly roaming the net. Do you leave yourself logged in  
>twenty-four hours a day, even when you're not home? Is your  
>wpm typing speed higher than your IQ? Are you having trouble seeing  
>things at distances greater than 2 feet? Yes, YOU. You know who  
>you are.  
>  
>Ok... shall we begin? Yes? 5 points... (you could've backed out.)  
>  
>Unless otherwise stated, point values are as follows: 2 for (a), 4 for  
>(b), 6 for (c), and 10 for (d).  
>  
>-----  
>  
>1) How many valid net addresses do you have?  
>  
> Multiple machines at the same site do not count.  
>  
> ___Internet ___UUCP ___Other public access ___Other  
>  
> ___Bitnet ___Freenet ___Internet BBS ___All seven  
>  
> (2 points each)  
>  
>  
>2) How many hours did it take for you to create your .sig?  
>  
> a) Huh?  
> b) More than one  
> c) More than five  
> d) I'm still looking for a really funky quote  
>  
>  
>3) On an average working day, how many email messages do you receive?  
>  
> a) Nobody sends me any mail... sniff  
> b) Three, but they're all from Lester in the next cubicle  
> over, because he has nothing better to do  
> c) I can't count that high, I failed calculus  
> d) Don't ask me now, I'm too busy. Send me e-mail.  
>  
>  
>
```

'cal 9 1752' explained

```
>4)    Alright, fess up.  Have you ever read alt.sex.bondage just to
>       see what the heck those perverts were talking about?
>
>       a)  Yes, and I'm so ashamed
>       b)  Yes, and I'm so embarrassed
>       c)  Yes, and would you please explain a few things to me...
>       d)  No, never.      (10 points.  You're lying.)
>
>
>5)    Have you ever met one of your past SO's (significant others)
>       via a computer network?
>
>       a)  No
>       b)  Yes, through a newsgroup we both posted on
>       c)  Yes, by chatting randomly over the Internet (shame!)
>       d)  Yes, by chatting over RELAY
>
>
>6)    Once you've logged onto your system, what do you spend most
>       of your time doing?
>
>       a)  Going through the library system and putting books on reserve
>       b)  Reading _Alice in Wonderland_ in the online bookshelf
>       c)  Reading the monthly postings on rec.humor.funny
>       d)  Writing up stupid quizzes because you've done everything else
>
>
>7)    If someone were to telephone your home at any given moment of the
>       day, what would be the percent chance that your phone would
>       be busy?
>
>       a)  Zero... I've got call waiting
>       b)  25%.... I only dial in from work  (Uh, hi, boss)
>       c)  75%.... Duh, so that's why nobody ever calls me
>       d)  Zero... My modem has a separate phone line
>
>
>8)    Which usenet newsgroups do you spend the most time reading?
>
>       a)  The comp. groups... because they're so informative
>       b)  The soc. groups.... because they're so multicultural
>       c)  The rec. groups.... because they're so diverting
>       d)  The alt. groups.... because I don't know what half those
>           words mean
>
>
>9)    What's your worst complaint about having an Internet account?
>
>       a)  I have to pay $5/month for it
>       b)  The damn sysadmins won't give me enough quota to hold
>           all my .gif's
>       c)  All those programmers keep tying up the modem lines
>       d)  I have to stay in school to keep it
>
>
>10)   Check your watch now.  What time is it?
>
>       a)  10 am... coffee break
>       b)  3 pm.... General Hospital's on
>       c)  12 am... one last login before I hit the sack
```

'cal 9 1752' explained

```
>      d)  4 am.... Oh my God, I've got a test tomorrow
>
>
>ALRIGHT, GUYS.  SCORING TIME.
>
>0-25 points:  You're not a nerd.  Go read a manual or two and come back
>               next year.
>25-50 points: You're an up-and-coming Internet nerd.  Why don't you
>               telnet over to 128.6.4.8 and play around with the Quartz
>               BBS for a while.
>50-75 points: You're a full-fledged Internet nerd.  Join the club.
>75-100 points: You're an Internet addict.  Try going to the library
>
>               this week, it'll do you some good.
>100+ points:  You're an Internet obsessive-compulsive.  Unplug your
>               computer, go out in the woods for a few days, and relax.
>               Lay back and listen to the birds singing.  Clear your mind.
>               And don't forget to unsubscribe yourself from all those
>               lists before you leave.
>
>
>----- written by slewis@magnus.acs.ohio-state.edu (Sarah Lewis) in a moment
>           of extreme boredom.  Disclaimer:  OSU doesn't know I wrote this, and
>           it's probably better that way.  Sigh.  Time to hit the books....
>
>--
>Selected by Brad Templeton.  MAIL your joke (jokes ONLY) to funny@clarinet.com.
>
>Please!  No copyrighted stuff.  Also no "mouse balls," dyslexic agnostics,
>Iraqi driver's ed, Administratium, strings in bar or bell-ringer jokes.
>
--
Lance R. Bailey  System/Network Manager           Robarts Research Institute
                NeXT/email: lrb@rri.uwo.ca       box: Clinical Trials Resources Group
                fax: 519.663.3789                 P.O. Box 5015, 100 Perth Dr.
                vox: 519.663.3787 ext. 4108       London, Canada N6A 5K8
                Gaelic: Tha me leatsa ma cheannaicheas tu mi. Thoir dhomh earbsa.
                Anglo: I will be yours if you bribe me well. Trust me.
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Argument and Debate on USENET

Sort of as a philosophical followup to my comments on argument and debate on USENET, I thought it'd be different to describe the various forms of argument that go on here on the net and why some styles of debate are better than others. If this sort of stuff bores you, well, sorry -- there seems to be enough interest to post this (and maybe even start a discussion or two).

First, a quick and dirty definition. What is an argument/debate? (I'll use the terms interchangeably. They really aren't the same, but for this discussion it doesn't make sense differentiating). It is where someone with a given point of view attempts to persuade people who are either neutral or in disagreement to accept the view he's proposing. An argument can (and probably will) have multiple views with multiple people trying to sway the opinion of the listener.

To be successful, an argument has to stick to the facts. You can build a case around the facts and generate a powerful and persuasive rhetoric to deliver the facts, but when push comes to shove, your facts have to be better than their facts.

An example: [note: numbers are made up. Don't yell at me, my reference book is at home]:

Speaker A: The sky is blue.

Speaker B: The sky is green.

Speaker A: You are incorrect. The atmosphere refracts light at a wavelength of 450 Angstroms, which is in the blue part of the visible light spectrum.

Speaker B: Oh, yes, you're right. I'm sorry.

When confronted by facts, Speaker B was forced to admit that he was incorrect. The sky, in fact, is not green. He could also change his position without being embarrassed because facts are facts and it's not a crime to be wrong. Even on USENET. Note also how the argument stuck to a factual, non-emotional basis, making it easier for the two to come to a resolution.

This kind of argument is almost extinct on USENET. The most common version of the USENET argument goes like this:

Speaker A: The sky is blue.

Speaker B: Everyone knows the sky is green. If you don't think so, you're an idiot!

Speaker A: Listen, bozo! The atmosphere refracts light at a wavelength of 450 Angstroms, which is in the blue part of the visible light spectrum.

Speaker B: You @\$#@! How dare you interject facts into an argument!

Note the difference between the two arguments. In this version, Speaker B ignored the facts and returned an emotional rebuttal. This is known as an "If you can't win, insult his mother" argument. The end result is that both speakers are now mad at each other, with the level of emotional diatribe escalating at every path. Another problem is that the original reason for the argument has been completely forgotten in the heat of the moment.

This is why so little gets done on USENET. Every time there is a disagreement, the facts get scattered to the winds while everyone digs a foxhole and tosses insults and abuse at each other. It's impossible to win an argument when you are off the topic, but the debate tactic most often used on USENET is to immediately shift the topic from the discussion at hand

'cal 9 1752' explained

to an emotional, personal attack.

There's a common variation to this argument form:

Speaker A: The sky is blue.

Speaker B: If you don't think so, you're an idiot!

Speaker A: Listen, bozo!

Speaker B: You @#\$#@\$!

This is almost the same as the previous argument, except that we now have absolutely no idea what Speaker B thinks except that he disagrees with speaker A. In the previous form, we at least knew he felt the sky was green. This is called a content-free rebuttal -- we know they're disagreeing, but we have no idea why Speaker B disagrees, what his alternate position is or why he thinks Speaker A is wrong. This makes it impossible for anyone other than Speaker B to understand the issues and come to any kind of resolution (except, perhaps, that Speaker B is a twit).

A second common form of USENET argument is the "no it isn't" argument. It can be defined like this:

Speaker A: The sky is blue.

Speaker B: Everyone knows the sky is green.

Speaker A: The sky is blue.

Speaker B: Everyone knows the sky is green.

Speaker A: The sky is blue.

Speaker B: Everyone knows the sky is green.

Speaker A: The sky is blue.

Speaker B: Everyone knows the sky is green.

Speaker A: The sky is blue.

Speaker B: Everyone knows the sky is green.

Eventually, Speaker A gets tired of endless, repetitive arguing and shuts up. Speaker B then claims victory, since he spoke last. This 'victory by default' form of arguing has been known to go on at some level for months or years between some people. It is also very boring and doesn't accomplish anything. The United States Government has refined this to an art form with the filibuster, in which a small, noisy group of legislators can bring everything to a halt indefinitely until everyone gets tired and goes away. This has been used with great success both there and here on USENET by people who know they're going to lose if the issue comes up for a vote, so they just refuse to let people vote on it by bickering trivialities endlessly. This is the "if you don't do it our way, we're going to take *your* toys and go home" debate.

Finally, there is a form of argument essentially unique to USENET known as the Spelling Flame. This is a modified form of the abuse/attack mode argument where, if you can't discredit a person's facts, you try to shift the discussion away from the topic by attacking the presentation itself (and by implication, the speaker):

Speaker A: The sky is blue.

Speaker B: The sky is green.

Speaker A: You are incorrect. The atmosphere refracts light at a wavelength of 450 Angstroms, which is in the blue part of the visible light spectrum.

Speaker B: Everyone knows it's spelled "Angstrom", you dit! Geez, what an idiot you are that you can't even spell correctly! Why should anyone listen to *you*!

Note how we're suddenly arguing spelling and grammar, not wavelengths. Note also how Speaker B stuck in an attack against Speaker A by using the

'cal 9 1752' explained

spelling error as an implicit discreditation of what is, in reality, a correct fact.

Note finally that, in fact, Speaker *B* got the spelling wrong, which will cause a double-back spelling flamewar guaranteed to make sure that nobody will be able to discuss wavelengths in that group for months to come.

A few final comments on discussions. If you look at the various forms of argument, you should notice one important thing. Only in the first form, where the discussion is limited to facts and the topic at hand, is there a resolution. In the other arguments, Speaker B successfully deflects Speaker A from winning the argument, but lack of a resolution can not be considered a resolution or victory for the other side. All that's been done is made sure that nothing happened (which may be what the opposition wants, but which isn't a good thing for the net in general).

Lots of people wonder why it seems to take forever to get things done on USENET. If you analyze *how* things are done on USENET, it becomes obvious. The standard form of discussion of USENET can best be described as "feint and distract" -- rather than allow someone to win an argument, many people on USENET will simply try to get the argument to go away or change directions. Not because they think their side is necessarily right, but because their ego won't allow them to admit defeat.

As long as that continues on USENET, it'll continue to move in geologic time. The key in all this is simple: when you're involved in a discussion, before you post a followup, ask yourself a couple of simple questions:

- o Is what I'm going to say have anything to do with the topic?
- o Has what I'm going to say been said before?
- o Am I sticking to the discussion? Or am I shifting to an emotional, personal aspect of the discussion?
- o Would I be pissed if someone said that about me?

If you sit and think about it and avoid deflecting topics or moving onto tangents, maybe we could accomplish things around here...

--

Chuq Von Rospach Editor, OtherRealms Member SFWA/ASFA
chuq@apple.com CI\$: 73317,635 [This is myself speaking. I am not Appl
Segmentation Fault. Core dumped.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Assembler as it REALLY is

MNEMONIC	INSTRUCTION
AAC	Alter All Commands
AAR	Alter at Random
AB	Add Backwards
AFVC	Add Finagle's Variable Constant
AIB	Attack Innocent Bystander
AWTT	Assemble with Tinker Toys
BAC	Branch to Alpha Centauri
BAF	Blow all Fuses
BAFL	Branch and Flush
BBIL	Branch on Burned-Out Indicator Light
BBT	Branch on Binary Tree
BBW	Branch Both Ways
BCIL	Branch Creating Infinite Loop
BDC	Break Down and Cry
BDT	Burn Data Tree
BEW	Branch Either Way
BF	Belch Fire
BH	Branch and Hang
BMR	Branch Multiple Registers
BOB	Branch on Bug
BOD	Beat on the Disk
BOI	Byte Operator Immediately
BPDI	Be Polite, Don't Interrupt
BPO	Branch on Power Off
BRSS	Branch on Sunspot
BST	Backspace and Stretch Tape
BW	Branch on Whim
CBNC	Close, but no Cigar
CDC	Close Disk Cover
CDIOOAZ	Calm Down, It's Only Ones and Zeroes
CEMU	Close Eyes and Monkey With User Space
CH	Create Havoc
CLBR	Clobber Register
CLBRI	Clobber Register Immediately
CM	Circulate Memory
CMD	Compare Meaningless Data
CML	Compute Meaning of Life
CNB	Cause Nervous Breakdown
COLB	Crash for Operator's Lunch Break
CPPR	Crumple Printer Paper and Rip
CRASH	Continue Running After Stop or Halt
CRB	Crash and Burn
CRN	Convert to Roman Numerals
CS	Crash System
CSL	Curse and Swear Loudly
CU	Convert to Unary
CVG	Convert to Garbage
CWOM	Complement Write-only Memory
CZZC	Convert Zone to Zip Code
DBTP	Drop Back Ten and Punt
DBZ	Divide by Zero
DC	Divide and Conquer
DDC	Dally During Calculations
DLN	Don't Look Now...
DMNS	Do What I Mean, Not What I Say

'cal 9 1752' explained

DMPE	Decide to Major in Phys. Ed.
DMPK	Destroy Memory Protect Key
DNPG	Do Not Pass Go
DO	Divide and Overflow
DOC	Drive Operator Crazy
DPMI	Declare Programmer Mentally Incompetent
DPR	Destroy Program
DTC	Destroy This Command
DTE	Decrement Telephone Extension
DTVFL	Destroy Third Variable From Left
DW	Destroy World
EBB	Edit and Blank Buffer
ECO	Electrocute Computer Operator
EEOIFNO	Execute Every Other Instruction From Now On
EFD	Emulate Frisbee Using Disk Pack
EIAO	Execute In Any Order
EIOC	Execute Invalid Op-code
EMPC	Emulate Pocket Calculator
EMSE	Edit and Mark Something Else
ENF	Emit Noxious Fumes
EP	Execute Programmer
EPI	Execute Programmer Immediately
EROS	Erase Read-only Storage
EXOP	Execute Operator
EXPP	Execute Political Prisoner
FLI	Flash Lights Impressively
FSM	Fold, Spindle and Mutilate
FSRA	Forms Skip and Run Away
GCAR	Get Correct Answer Regardless
GDP	Grin Defiantly at Programmer
GFD	Go Forth and Divide
GFM	Go Forth and Multiply
HCF	Halt and Catch Fire
HCP	Hide Central Processor
IAE	Ignore All Exceptions
IBP	Insert Bug and Proceed
IIB	Ignore Inquiry and Branch
ISC	Insert Sarcastic Comments
JTZ	Jump to Twilight Zone
LAP	Laugh At Programmer
LCC	Load and Clear Core
LPA	Lead Programmer Astray
MAZ	Multiply Answer by Zero
MBR	Multiply and be Fruitful
MLR	Move and Lose Record
MW	Malfunction Whenever
MWAG	Make Wild-Assed Guess
MWT	Malfunction Without Telling
OML	Obey Murphy's Laws
P\$*!	Punch Obscenity
PBC	Print and Break Chain
PD	Play Dead
PDSK	Punch Disk
PEHC	Punch Extra Holes in Cards
PFD	Punt on Fourth Down
PI	Punch Invalid
PNRP	Print Nasty Replies to Programmer
POCL	Punch Out Console Lights
POPI	Punch Operator Immediately
PVLC	Punch Variable Length Card

'cal 9 1752' explained

RA	Randomize Answer
RASC	Read and Shred Card
RCB	Read Command Backwards
RD	Reverse Directions
RDA	Refuse to Disclose Answer
RDB	Run Disk Backwards
RDS	Read Sideways
RIRG	Read Inter-record Gap
RLI	Rotate Left Indefinitely
ROC	Randomize Op Codes
ROO	Rub Out Operator
RPB	Read Print and Blush
RPM	Read Programmer's Mind
RRSGWSSNK	Round and Round She Goes, Where She Stops, Nobody Knows
RSD	On Read Error Self-Destruct
RSTOM	Read From Store-only Memory
RTS	Return To Sender
RWCR	Rewind Card Reader
SAI	Skip All Instructions
SAS	Sit And Spin
SCCA	Short Circuit on Correct Answer
SFH	Set Flags to Half-mast
SFT	Stall For Time
SHAB	Shift a Bit
SHABM	Shift a Bit More
SLP	Sharpen Light Pen
SMR	Skip on Meaningless Result
SOS	Sign off, Stupid
SOT	Sit on a Tack
SPS	Set Panel Switches
SPSW	Scramble Program Status Word
SQPC	Sit Quietly and Play With Your Crayons
SRDR	Shift Right Double Ridiculous
SRSD	Seek Record and Scar Disk
SRZ	Subtract and Reset to Zero
SSJ	Select Stacker and Jam
STA	Store Anywhere
STROM	Store in Read-only Memory
TARC	Take Arithmetic Review Course
TDB	Transfer and Drop Bits
TLO	Turn Indicator Lights Off
TN	Take a Nap
TPDH	Tell Programmer to Do it Him/Herself
TPF	Turn Power Off
TPN	Turn Power On
TTA	Try, Try Again
UCB	Uncouple CPU and Branch
UER	Update and Erase Record
ULDA	Unload Accumulator
UP	Understand Program
UPCI	Update Card In Place
WBT	Water Binary Tree
WEMG	Write Eighteen Minute Gap
WF	Wait Forever
WHFO	Wait Until Hell Freezes Over
WI	Write Illegibly
WPM	Write Programmer's Mind
WWLR	Write Wrong-Length Record
WSWW	Work in Strange and Wondrous Ways
XSP	Execute Systems Programmer

'cal 9 1752' explained

ZAR	Zero Any Register
ACC	Advance CPU clock
AEE	Absolve engineering errors
BB	Branch on bug
BIRM	Branch on index register missing
BOHP	Bribe operator for higher priority
BSO	Branch on sleepy operator
CCD	Clear core and dump
EC	Eat card
EM	Emulate 407
EMW	Emulate Matag washer
IAI	Inquire and ignore
KCE	Kill consultant on error
LSPSW	Load and scrample PSW
MST	Mount scotch tape
PBC	Print and break chain
PLSC	Perform light show on console
PPSW	Pack program status word
RBT	Rewind and break tape
RDI	Reverse dum immediate
RT	Reduce thruput
SIP	Store indefinite precision
SPD	Spin dry disc
SRCC	Select reader and chew cards
TSH	Trap secretary and halt
UNPD	Unplug and dump

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Be Environmentally Friendly and Help Reduce Computer Waste!

Be Environmentally Friendly and Help Reduce Computer Waste!

What we hope to establish is a refurbishing and recycling business for Internet Information Services. The idea is to collect "pre-owned" bytes of information, recondition them and resell them at a discount. We already have some 256 different values in our catalogue (including 128 `_different_ ASCII` values!) and hope to launch the service in the New Year. Users will be able to obtain large numbers of "pre-loved" bytes for use in their own information services applications for a fraction of the cost of new bytes. Those willing to take the bytes without support can expect to save even more!

The only problem we have found so far is obtaining a reliable source of used bytes for recycling. We're presently collecting bytes written to `/dev/null` on our own local UNIX machines and have also managed to recycle a good portion of the huge amounts of output from thearchie telnet client (you'd be `_amazed_` at how much junk that program produces...)

We're also receiving over 30 Megabytes a day of what appears to be randomly generated text strings over our Usenet feed and have found that we can recycle a good portion of the headers from each posting (sadly, the very low information content in the postings themselves seems to rule out using the message bytes for recycling as we're concerned about quality control on our recycled bytes).

We've even tapped into the "chargen" ports on a number of machines accessible over the Internet to obtain large number of ASCII values, although the net traffic this generates has yet again caused the McGill University administration to notice what we're up to, so I doubt we'll be able to use this source in a commercial service. Still, it's given us enough material to help us get the recycling technology prototyped and working.

Unfortunately, given the huge anticipated demand for such recycled bytes we believe that this will definitely `_not_` be enough to meet demand once we go into production mode, so we're currently looking around for alternative sources of supply.

To meet the anticipated demand I'd like to ask each and every one of you on this list to consider setting up a byte recycling service for your network. The idea is to have each individual gather up and send for recycling whatever bytes they no longer need on their machine. Doesn't this sound better than simply throwing them all away with "rm" or "DEL", or by dragging them to that silly little trashcan icon?

Besides, haven't you ever asked yourself where those bytes go when you throw them away? Believe it or not, most deleted bytes currently find their way straight into the Internet refuse stream, often to turn up in postings to Usenet, or worse, OSI standards documents!

We're trying to do our bit to prevent this from happening and think a viable recycling industry for the Internet is one answer to the problem (and one that will make us all here at Bunyip filthy rich, which is why we're proposing the idea).

'cal 9 1752' explained

Here at Bunyip, we're certainly doing our part to get things started. We're currently working on a new pseudo-device driver for the UNIX environment that will automatically collect the bytes and forward them on to our processing centres without further human intervention. Users need only write the bytes to the new device and they will be automatically sent off for processing!

In time, we hope to see this pseudo-device (tentatively named `"/dev/blue-box"`) installed on every UNIX machine on the network. We firmly believe a strongly supported recycling program would enable us to meet the anticipated demand for pre-owned bytes while having a very positive effect on the ecology of the Internet waste stream (and make us lots of money, but I mentioned that bit already, didn't I? No point in giving the game away entirely, you know. Walls have ears, etc, etc...)

Of course, in the beginning such recycling will not be entirely automated. Users will still be expected to sort their bytes into several groups, including ASCII text, data and "Usenet postings" (which given their low information content obviously aren't worth recycling). Users will also be expected to rinse out their data bytes and remove all labels from files imported from the IBM mainframe environment.

Of course, users will have to use the major number/ minor number pair on the device driver to determine which type of refuse you are recycling [* Note: non-UNIX types should go ask a UNIX weenie to explain this reference. On second thought, don't bother. It's not that funny... *] Anyways, this is a minor inconvenience considering the environmental advantages and savings in hardware costs users can expect from a successful recycling program.

For example, apparently accretion of such waste bytes on disk heads (which build up over time when users continually create and delete files in a networked environment and systems admins continually dump and restore bytes that haven't been accessed for a long time) are one of the major cause of unexplained disk crashes.

Really...

In the future, we might even be able to talk about paying you for your used bytes, although of course in the beginning this would be out of the question as we seek to build our profit margins to the point where we could afford the heating bills we face up here in Montreal each year. We're sure you all understand. Besides, most right-thinking people agree that paying for information on the Internet, although not illegal, probably should be...

Oh, one last comment. One problem we thought we'd have would be in importing and exporting the used bytes into Canada for processing, but apparently this is clearly covered under a clause of the current Canada-U.S. Free Trade Agreement. This clause was apparently put in place to allow the Americans to send transcripts of their political speeches up North for disposal after each election while the Canadians were hoping to profit by sending transcripts of the on-going constitutional and language debates down south each year. Economists point to this clause as one example of the FTA in which the net benefits to each party have been exceedingly hard to define or measure...

In any event, as a global player in the Information Services field, Bunyip is of course studying the possibility of sending the bytes down

'cal 9 1752' explained

to Mexico for processing in a maquiladora plant in Baja California in the event the North American Free Trade Agreement manages to come to fruition. Apparently, workers down there will work for as little as 50 cents a day plus all the Usenet postings they can read. Talk about lax environmental laws!

Although we regret the necessity of sending badly needed hi-tech jobs down south, the potential profits will probably be impossible to resist. We figure we can salve our collective corporate consciousness on this by donating a certain number of processed bytes to be used in Canadian Federal government studies on unemployment in the information processing field. There, I feel better already...

Well, that's our plan for the new service. Now, this may not seem as immediately relevant to you as the need to sell used cars to carry commercial IP traffic (that is what we're talking about, right? I've kind of lost track). Anyways, I think such exciting (and lucrative) examples of the potential for a commercial Internet information market should be brought to the attention of this list. Might give everybody something else to talk about!

So, hope you found this at least as valuable as some of the other stuff coming over this list right now. I'd like to think I've done my bit to make com-priv what it is today....

- peterd

--- Insert Legal Bit Here ---

The preceding posting is copyright (c) 1992, by Peter Deutsch, of Bunyip Information Systems Inc, who should be solely responsible for its contents. Unfortunately for you, like most people I know I had an unhappy childhood (for at least a part of it) and thus can't be held responsible for anything I say or do. On the other hand, if you want to reprint this somewhere, you'd better let me know, and maybe even send me a cheque, or my lawyers will be calling your lawyers and doing lunch! (can you tell I was born in California? :-)

So there....

Peter Deutsch,
Bunyip Information Systems

Disclaimer? What disclaimer? I'm president of the company and own half the stock. these are the opinions of the company...

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Becoming A UNIX Wizard

> What does it take for a man, er, person, to become a Unix Wizard ?

Oh, Nobly Born: know that the Way to Wizardhood is long, and winding, and fraught with Peril. Thou must Attune thyself with the Source, attaining the arcane Knowledge and Conversation of the System Libraries and Internals.

> Which bibles should we study ? (after memorizing O'Rileys to become
> an X Wizzie ?)

Oh, Nobly Born: the O'Reilly books are but the palest shadow, the outermost Portal of the True Enlightenment.

If thou desirest with true desire to tread the Path of Wizardly Wisdom, first learn the elementary postures of Kernighan & Pike's The Unix Programming Environment; then, absorb the mantic puissance of March Rochkind's Advanced Unix Programming. Immerse thyself, then, in the Pure Light of Maurice J. Bach's The Design Of The UNIX Operating System. Neglect not the Berkelian Way; study also The Design Of The 4.3BSD UNIX Operating System by Samuel Leffler, Kirk McKusick et.al. Consider also the dark wisdom to be gained from contemplation of the dread Portable C And UNIX Systems Programming, e'en though it hath flowed from the pen of the mad and doomed Malvernite whom the world of unknowing Man misnames "J. E. Lapin".

These tomes shall instruct thy Left Brain in the Nature of the UNIX System; to feed the other half of thy head, oh Nobly Born, embrace also the Lore of its Nurture. Don Libes's and Sandy Ressler's Life With UNIX will set thy feet unerringly upon that Path; take as thy Travelling Companion the erratic but illuminating compendium called The New Hacker's Dictionary.

(In this wise shalt thou travel the Way of the Camel.)

> How many kernels (and which ones) we need to take apart and
> reassemble ?

Oh Nobly Born: this question reveals that indeed thou hast touched upon the Ineffable Truth about UNIX --- that thou canst not Plumb its Mysteries by mere Study but must become One with it through Practice. The true Way to the Knowledge of the Source is not the timid and footling way of the Student, but the Divine Foolery of the Hacker. Hack, then; strive against Mighty Problems, have joy in your Striving, and let the Crashes fall where they may (maintaining the while, for the Good of your Karma, a Rigorous Backup Policy).

(In this wise shalt thou travel the Way of the Lion.)

In this day of Boot-Time Autoconfiguration and Dynamically Loadable Device Drivers, reassembling a Kernel is no longer the daunting test and seal of Mastery that it once was. However, writing and verifying your own Device Driver for some piece of Exotic Hardware is still a worthy challenge to thy Budding Guruhood.

> Some doth say that the only way thou shall gain salvation is to
> partake in the (sacrireligios) ceremony of rewriting Unix from

'cal 9 1752' explained

> scratch.

Sacrilegious, oh Nobly Born? Nay! Certainly the Source is the Inmost Mystery of UNIX --- but there is a Mystery beyond that Mystery. The Nature of UNIX inhereth not in any one version but in the Design Tradition of which all UNIXes are Evolving Parts.

The Rite of the Rewrite is not the only Path to Mastery, but it is perhaps the highest and most Sacred of all Paths. Few indeed are those who, travelling it, have crossed the dark and yawning Abyss of Implementation to Delivery. Many, yea, many in truth stagnate yet in the Desert of Delay, or linger ever in the ghastly limbo called Perpetual Beta.

> So, oh, mylord, how does a mortal become a Unix Wizard ?

Oh Nobly Born: learn, and seek within yourself. Cultivate the cunning of the Serpent and the courage of the Tiger; sup deeply from the Wisdom of those who came before thee. Hack, and hack again; grow, by trial and by error. Post thy best hacks to the Net and gain in Repute thereby. Also, oh Nobly Born, be thou grave and courteous in thy speech; be helpful to those less than thee, quick to ponder and slow to flame.

If thou dost these things faithfully, if thou travellest with high heart and pure intention, soon shall thy callow Newbiehood be shed. By degrees imperceptible to thyself shalt thou gain Power and Wisdom, striving and doing all the while. Gradually shall thy puissance unfold and deepen.

Oh Nobly Born, if thou dost all these things, thy Wizardhood shall surely come upon thee; but not of a sudden, and not until after thy arrogant Mind hath more than half forgotten that such was its aim. For know this --- you may not by yourself in Pride claim the Mantle of Wizardry; that way lies only Bogosity without End.

Rather must you Become, and Become, and Become, until Hackers respect your Power, and other Wizards hail you as a Brother in Wisdom, and you wake up and realize that the Mantle hath lain unknown upon your Shoulders since you knew not when.

(In this wise shalt thou travel the Way of the Child.)

> (next question: how does a newbie become a net.personality ?)

Oh, Nobly Born: not by trying... --

--Eric S. Raymond

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Better Watch Out

```
#!/bin/sh
#      santa claus is coming to town

better !pout !cry
better watchout
lpr why
santa claus town

cat /etc/passwd >list
ncheck list
ncheck list
cat list | grep naughty >nogiftlist
cat list | grep nice >giftlist
santa claus town

who | grep sleeping
who | grep awake
who | grep bad || good
for (goodness sake) {
    be good
}
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Boot It

Sing this one to Michael Jackson's "Beat it"....

You're processing some words when your keyboard goes dead,
Ten pages in the buffer, should have gone to bed,
The system just crashed, but don't lose your head,
Just BOOT IT, just BOOT IT.

Better think fast, better do what you can,
Read the manual or call your system man,
Don't want to fall behind in the race with Japan,
So BOOT IT,

Get the system manager to

BOOT IT, BOOT IT,
Even though you'd rather shoot it.
Don't be upset, it's only some glitch.
All that you do is flip a little switch.
BOOT IT, BOOT IT,
Get right down and retribute it.
Don't get excited, all is not lost.
CP/M, UNIX or MS/DOS
Just BOOT IT, boot it, boot it, boot it...

You gotta have your printout for the meeting at two,
The system says your jobs at the head of the queue,
Right then the thing dies but you know what to do,
BOOT IT.

You always get so worried when the system runs slow,
And when it finally crashes, man you feel so low,
But computers make mistakes (they're only human you know)
So BOOT IT,

Call the local guru to

BOOT IT, BOOT IT,
Go ahead re-institute it.
If you're not lucky, get the book off the shelf,
But if you are, it'll do itself.
BOOT IT, BOOT IT,
Then go find the guy who screwed it!
Operating systems are built to bounce back,
Whether it's a Cray or a Radio Shack.

BOOT IT, BOOT IT,
(Repeat until next crash...)
/* ----- */

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Building A Main Frame

I went to buy computers and they said a million bucks.
For a brand new mainframe IBM; now that price really sucks.
So I looked at all the pictures and I chose to build my own
For you can build a mainframe from the things you find at home

Refrain:

[Oh IBM Dec and Honeywell, HP DG and Wang.
[Amdahl, NEC and NCR, they don't know anything!
[They make big bucks from systems, so they never want it known...
[That you can build a mainframe from the things you find at home.

Now take the CPU, you see it's only just a box.
With blinking lights and whirring fans and lots of cable slots
So I spent an evening working with some wire and bolts and pins
then I hauled it to the basement and I plugged the freezer in.

(Refrain)

Well next I needed tape drives, my tape drives were a steal.
I climbed into the attic for my Dad's old reel to reel
some cable and some binder twine and soon it was complete.
my CPU and tape drives at a price that can't be beat.

(Refrain)

So then I looked for disk drives but it didn't take me long
Function follows form they say or have I got it wrong?
So I found a drive with five new modes that blows the rest away:
Cottons, linens, wash-and-wear, rinse and lingere!

(Refrain)

I had to have a console just to make my system run
Without a fancy console then my system wasn't done.
So I tied to a ten buck typer to a broken TV tube
And now I've got a console that can write the evening news.

(Refrain)

I've had my system running - I'll admit it's not the best
The data isn't right and the response time is a mess
It crashes every hour and it isn't worth a damn
But I'm satisfied because it runs just like an IBM

(Refrain)

... from a tape my girlfriend brought to me. Anyone know who sings it?

Dedicated to our IBM 3090-180 VF and poor Randy who had to keep putting
back together all summer...

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

CS350a Final Exam

THE UNIVERSITY OF WESTERN ONTARIO
LONDON CANADA

Department of Computer Science

CS350a

Final Exam

Examiner: Andrew D. Marshall

- 1) Attached you will find a sexagesimal dump of the executable image of a full screen editor for one of the: MC68000, MC6809, LSI11 or RCA1802. There is a bug in the program. Disassemble the program, find and correct the bug. Hand assemble the corrected program using two passes, showing the symbol table at the end of each pass. Verify the correctness of the reassembled program using any method. You have 45 minutes, partial solutions will not be graded.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Categories of UNIX Users

UNIX User Categories

BEGINNER:

- insecure with the concept of a terminal
- has yet to learn the basics of vi
- has not figured out how to get a directory
- still has trouble with typing RETURN after each line

NOVICE:

- knows that ls will produce a directory
- uses the screen editor but calls it "vie"
- has heard of C but never used it
- has had his first bad experience with rm
- is wondering how to read his mail
- wonders why the person next to him likes UNIX so much

USER:

- uses vi and nroff but inexpertly
- has heard of regular expressions but never seen one
- has figured out that - precedes options
- has attempted to write a C program and decided to stick with Pascal
- is wondering how to move a directory
- knows how to read his mail and wonders how to read news

KNOWLEDGEABLE USER:

- uses nroff with no trouble and is learning tbl and eqn
- uses grep to search for fixed strings
- has figured out that mv will move directories
- has learned that learn(1) doesn't help
- somebody has shown him how to write C programs
- once used sed to do some text substitution
- thinks make is only for wimps

EXPERT:

- uses sed when necessary
- uses macro's in vi, uses ex when necessary
- posts news at every possible opportunity
- write C programs with vi and compiles with cc
- has figured out what && and || are for
- thinks that human history started with !h

HACKER:

- uses sed and awk with comfort
- uses undocumented features of vi
- writes C code with cat >foo.c and compiles with !cc
- uses adb because he doesn't trust source debuggers
- can answer questions about the user environment
- writes his own nroff macros to supplement standard ones
- writes scripts for Bourne shell (/bin/sh)
- knows how to install bug fixes

GURU:

- writes m4 and lex with comfort
- writes assembly code with cat >foo.s
- uses adb on the kernel while system is loaded
- customizes utilities by patching the source
- reads device driver source with breakfast
- can answer any unix question after a little thought

'cal 9 1752' explained

- uses make for anything that requires two or more commands
- has learned to breach security but no longer needs to

WIZARD:

- writes device drivers with cat >foo.o
- fixes bugs by patching the binaries
- can answer any question before you ask
- writes his own troff macro packages
- is on first-name basis with Dennis, Bill, and Ken

- Paul S. R. Chisholm

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Cleaning The Bit Bucket

>However the concern about whether news is biodegradable is something
>to think about. What does expire do with the toxic waste?

Um, I take it you haven't emptied out your bit bucket lately?

OK, this is best done in single-user mode, but you can do it multiuser if you turn off uucico first. You will need to take your computer to an authorised disposal site, or establish a high speed connection to one. You can do this at 2400 baud, but you have to watch the flow rate carefully to avoid contaminating valid data. Also, be sure to clean off the used bits from your modem, or you can get bit-rot and opportunistic computer virus infections.

There are places that rent modems for this purpose, but make sure they have properly cleaned the unit before you take it home.

OK, carefully unscrew /dev/null and put it aside. If you have a /dev/zero, use it to wash off the exposed sections of the kernel before and afterwards, otherwise you can use "dd" to generate a large empty file. Locate the entry point for the bit bucket, and attach a streams tap to it (it's useful to gen this into your kernel if you're going to be doing this yourself). Connect the other end of the stream to a convenient named pipe, and set it aside.

Dial the disposal company and attach the pipe to the modem. Open the tap. If bits don't start flowing immediately, you may have a vapor lock: delete a few files (I use /usr/spool/news/junk/*) to force the bubble out.

After the bit bucket has emptied, reverse these steps. Make sure that both ends of the stream are closed. You shouldn't reuse this stream for valid data, but you can keep it in a handy core file for next time. Make sure you chmod it to 0 so some user doesn't try to use it. Or you can simply toss it in /dev/null.

Remove and clean the modem, even if it's a rental: it doesn't take long and it's just common courtesy. Delete a few files to make sure all the links are secure... be especially careful of symbolic links, if you use them. You can now return to multi-user mode or turn on uucico confident that your bit bucket will not overflow for at least another 6 months... though I would recommend cleaning it at least twice that frequently to avoid disk errors and increase gas (and gcc) mileage.

-- Peter da Silva.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Commandments of a Computing Centre

THOU SHALT NOT:

- 1) Worship other computer systems
- 2) Impersonate Liberace while sitting at the console keyboard.
- 3) Use magnetic tapes as frisbees.
- 4) Slap users on the face.
- 5) Get physically involved with the computer system, if already married.
- 6) Use a disk pack as a stool to reach another disk pack.
- 7) Stare at blinking lights for more than one 8 hour shift.
- 8) Tell a user that you accidentally destroyed his/her files and his/her backup, just to see the look on his/her face.
- 9) Staple card decks together.
- 10) Enjoy cancelling a job.
- 11) Do unto the director as he does unto you.
- 12) Eat watermelon while sitting at the console.
- 13) Spit at a user when he/she brings in 10 thirty minute jobs.
- 14) Display firearms in the computer room.
- 15) Push buttons just to see what happens.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Common Slop Multiple Access with Cow Detect

Found on the last page of "The RED REPORT", the newsletter from R.E.D. Electronics Inc.

E.I.E.I.O. specifies the spacing and number of pigs in a single trough. Pigs must be spaced no closer than 2.55 meters, with a maximum of 100 pigs and a total trough length of 500 meters.

All pigs contend for feed access by adhering to CSMA/CD (Common Slop Multiple Access with Cow Detect). With CSMA/CD pigs contend for access to the feed. If a cow is detected the pig waits a random amount of time before attempting access again, thus assuring access to all pigs.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Computer Limericks

The BBC recently ran a computer limerick competition, through
"The Listener". I've just received the issue with the winners.
The competition was to write a limerick starting
"A programming genius called ..."

The winner--

A programming genius called Hank
Wrote a system to 'access' his 'bank'
When his memory failed him
They nailed him then jailed him
Now his 'storage' is 'basic' and dank.
W E Sword, Barningham Suffolk

Runners up--

A programming genius called Gertie
Had a penchant for graphics so dirty
No computer she knew
Would accept what she drew
Until she had tickled its QWERTY.
C G Robbins, St Albans

A programming genius called HEAP
Had trouble in getting to sleep
So he made his lambs troupe
through a huge FOR-NEXT loop
FOR 1 TO 10000: NEXT sheep.
John Miles, Ely

A programming genius called Smiler
Thought he'd made up a simpler compiler
But through trouble with loops
Was constrained to cry 'Whoops,
This gets viler, gets viler, gets viler, ...
J A Cade, Streatley

A programming genius called Sewter
Built a limerick-writing computer.
The metre was fine
and the rhymes quite divine
But for some reason the damn thing always
got the last line wrong.
D Gutteridge, Chesterfield

A programming genius called Hugh
Said 'I really must see if it's true.'
So he wrote a routine
To ask 'What's it all mean?'
But the answer was still '42'.
R Humphries, Sutton Coldfield

--Julian

'cal 9 1752' explained

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

Computer Literacy Test

Quiz time...

1. Who invented the computer?
 - a. Mr. Chips
 - b. E.T.
 - c. Marquis de Sade
2. What are microchips?
 - a. What a herd of micros leave on the prairie.
 - b. What you eat with microdip.
 - c. The reason you had to take all those computer literacy courses.
3. What is a floppy disk?
 - a. A painful lower-back condition.
 - b. An album that didn't sell.
 - c. A great frisbee.
4. What is the first thing you associate with computers?
 - a. Bill Cosby commercials
 - b. Eyestrain, headaches.
 - c. Annoying beeps.
 - d. Three tons of printout where once there was a 3-page report.
 - e. All of the above.
5. What is FORTRAN?
 - a. Between 3 and 5 tran.
 - b. How to get computers excited before interface.
 - c. Ridiculous.
6. What is Pascal?
 - a. A leafy vegetable.
 - b. A foot fungus.
 - c. A city in southern France.
 - d. None of the above.
7. When you need consulting help in deciding what to do with your computer, what organization do you think of?
 - a. IBM
 - b. FBI
 - c. PLO
8. What is the most important computer peripheral?
 - a. Bill Cosby.
 - b. Someone to operate the computer for you.
 - c. Aspirin.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Creators Admit UNIX, C Hoax

COMPUTERWORLD 1 April

CREATORS ADMIT UNIX, C HOAX

In an announcement that has stunned the computer industry, Ken Thompson, Dennis Ritchie and Brian Kernighan admitted that the Unix operating system and C programming language created by them is an elaborate April Fools prank kept alive for over 20 years. Speaking at the recent UnixWorld Software Development Forum, Thompson revealed the following:

"In 1969, AT&T had just terminated their work with the GE/Honeywell/AT&T Multics project. Brian and I had just started working with an early release of Pascal from Professor Nicholas Wirth's ETH labs in Switzerland and we were impressed with its elegant simplicity and power. Dennis had just finished reading 'Bored of the Rings', a hilarious National Lampoon parody of the great Tolkien 'Lord of the Rings' trilogy. As a lark, we decided to do parodies of the Multics environment and Pascal. Dennis and I were responsible for the operating environment. We looked at Multics and designed the new system to be as complex and cryptic as possible to maximize casual users' frustration levels, calling it Unix as a parody of Multics, as well as other more risqué allusions. Then Dennis and Brian worked on a truly warped version of Pascal, called 'A'. When we found others were actually trying to create real programs with A, we quickly added additional cryptic features and evolved into B, BCPL and finally C. We stopped when we got a clean compile on the following syntax:

```
for(;P("\n"),R-;P("|"))for(e=C;e-;P("_"+(*u++/8)%2))P("| "+(*u/4)%2);
```

To think that modern programmers would try to use a language that allowed such a statement was beyond our comprehension! We actually thought of selling this to the Soviets to set their computer science progress back 20 or more years. Imagine our surprise when AT&T and other US corporations actually began trying to use Unix and C! It has taken them 20 years to develop enough expertise to generate even marginally useful applications using this 1960's technological parody, but we are impressed with the tenacity (if not common sense) of the general Unix and C programmer. In any event, Brian, Dennis and I have been working exclusively in Pascal on the Apple Macintosh for the past few years and feel really guilty about the chaos, confusion and truly bad programming that have resulted from our silly prank so long ago."

Major Unix and C vendors and customers, including AT&T, Microsoft, Hewlett-Packard, GTE, NCR, and DEC have refused comment at this time. Borland International, a leading vendor of Pascal and C tools, including the popular Turbo Pascal, Turbo C and Turbo C++, stated they had suspected this for a number of years and would continue to enhance their Pascal products and halt further efforts to develop C. An IBM spokesman broke into uncontrolled laughter and had to postpone a hastily convened news conference concerning the fate of the RS-6000, merely stating 'VM will be available Real Soon Now'. In a cryptic statement, Professor Wirth of the ETH institute and father of the Pascal, Modula 2 and Oberon structured languages, merely stated that P. T. Barnum was correct.

In a related late-breaking story, usually reliable sources are stating that a similar confession may be forthcoming from William Gates concerning the MS-DOS and Windows operating environments. And IBM spokesman have begun denying that the Virtual Machine (VM) product is an internal prank gone awry.

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

DataPlunger

If your computer machine-room is still equipped with a floor-plate puller which is actually a tool for handling plate glass, you need DATAPLUNGER!

It's an ordinary plumber's friend, used to lift false-floor plates. To disengage, just kick it sideways. (The DATAPLUNGER seems to be a very old tradition in some places, unknown in others. I first saw it in an IBM shop about 15 years ago.)

People here considered it a joke until our computing center was moved to a new building and demand for the single floor-puller became a problem during the new installation. We bought a dozen plungers from the local hardware store. The prototype has lasted for 5 years but the new batch were of low quality, and the pulling forces caused the internal threads of their rubber cups to strip out; the DATAPLUNGER should be reinforced with a hose clamp around the junction of cup and handle.

A deluxe model, the FLOOR HARPOON, uses a mop handle and can be used from a standing position. (One guy here likes to draw pictures of baby seals on the floor plates; another drew a target on the wall.) It was difficult to find a mop handle with the right threads to fit the rubber cup.

We had to mark our plungers "DATA ONLY," with a circle-and-slash over a picture of a toilet.

When the management gives a VIP tour of our spiffy new facility, it seems to embarrass them if someone walks by with a plunger over his shoulder. This means that some enterprising yuppie/yuppette could make big bucks by redesigning the device with a hi-tech look (black-anodized aluminum handle with holes and racing stripes... :^]

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Digital's hip to the standards thing

News item

NEW YORK -- During a creative session at a major public relations firm today to formulate a new corporate message for Digital Equipment Corporation that reflects the company's new direction promoting and supporting computing industry standards, the shopworn phrase "Digital has it now." was replaced by a new tag line that is more contemporary and tied to DEC's adherence to standards. "Digital's hip to the standards thing." will become the \$11.5 billion company's new-image slogan. In a radical departure from it's traditionally staid advertising approach the company will produce a television advertisement built around a rap music theme played out in a rapidly changing sequence of office and engineering scenes. In a parody of the commonly used product nomenclature, the video itself is referred to as a mythical Digital product called DECrap. The lyrics to the rap music video are published here for the first time.

A DECrap by

Rapmaster Ken

"Digital's Hip to the Standards Thing"

I heard some news just the other day
It sounded kinda strange and I said, "No way!"
But I heard it again from another source
It mighta made sense and I said, "Of course!"

Now computer biz has a lotta confusion
'Cause operating systems abound in profusion.
But there's a whole new wave in data processing
Now that Digital's hip to the standards thing.

(chorus)

Digital's hip to the standards thing!
Digital's hip to the standards thing!

Way back when a long time ago
IBM owned the whole show.
But other dudes saw this proprietary mess
And formed committees to find out what's best.

Some went their own way and built their own software
But users were perturbed, "It's just a different nightmare."
So they got together to look over the picks
Put their down their money on good 'ol UNIX

(chorus)

Digital's hip to the standards thing!
Digital's hip to the standards thing!

Now Digital always kept their users in mind
And pushed VMS as the best of the kind.
A lotta folks agreed but kept askin' for
UNIX support, "We gotta have more!"

Soon DEC saw the light and decided to give

'cal 9 1752' explained

UNIX to the masses, (sorta live and let live).
So DEC's ridin' the wave ahead of the rest
On a backplane bogie board on top of the crest.

No doubt about it DEC's sprouted its wings
'Cause Digital's hip to the standards thing.

(chorus)

Digital's hip to the standards thing!
Digital's hip to the standards thing!

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Douglas Adams' Guide to the Macintosh

Date: Tue, 25 Feb 92 10:06:02 -0500
From: alec@ftp.com
Subject: Guide to the Macintosh (Douglas Adams)

As the author of the four books of the Hitchhikers Guide to the Galaxy trilogy, and the perpetrator of two infocom text adventures (Hitchhiker's and Bureaucracy), Douglas Adams needs no introduction.

MacUser, "The Macintosh Resource", September 1987

Douglas Adams'

Guide to the Macintosh

WE ARE NOT EXACTLY SURE WHAT WE'VE GOT HERE - PERHAPS WE ASKED THE WRONG QUESTION - BUT WE THINK YOU'RE GOING TO LIKE THIS.

There's a joke i remember that went around my school playground (this was a while ago, sometime during the long dark ages that stretched from the emergence of Australopithecus on the plains of East Africa, to the release of "A Hard Day's Night") that went like this:

A man was giving a lecture on sexual techniques. There were, he said, eleven basic positions for sexual intercourse. "Two hundred and ninety seven!" interrupted a voice from the back. "The first of these eleven basic positions -," continued the speaker. "Two hundred and ninety seven," shouted the heckler again. "- is the one in which the man lies on top of the woman."

"Oh," said the heckler, momentarily flummoxed, "two hundred and ninety eight!"

I mention this for a reason [good - Ed.], which is that I want to contrast for a moment the number of features on two different word processors. One of them is Microsoft Word 3.0, billed as the most comprehensive word processor yet - powerful, flexible, configurable to the demands of any professional writing task, it takes 600 pages of manual just to describe all its features (twice, admittedly).

The other word processor is miniWRITER, a desk accessory which only has about two features, one of which Word 3.0 hasn't got. And it's not a negligible feature either. As a professional novelist and occasional desktop publisher it's the first thing I looked for after I'd torn off the shrink wrap, and I discovered it wasn't there, I cursed and swore, went out for a sullen lunch and shouted at the barman.

"Something wrong, sir?" he said. "Oh, nothing," I said gloomily. "It's just the new version of Microsoft Word."

"Ah," he said, wiping a glass sympathetically, "I expect it's the manual that'll be getting you down then, sir. I always tell my customers, 'there's nothing in life so difficult that a Microsoft manual can't make it completely incomprehensible.' One of my

'cal 9 1752' explained

regulars - chap called Fred, perhaps you know him, little wizened grey-haired fellow, about thirtyish - told me he'd been using Word 1.05 for two years before he discovered that you could search for carriage returns and tabs after all. He just thought they'd omitted it out of spite. But no, it was in there alright. It was even in the manual. Just not so as you could find it, that's all. It was his brother Jim as discovered it. He was doing three month solitary at the time. 'At least give me something to read,' he pleaded with the warders."

"Heartless brutes, they gave him a Microsoft Word manual. He was a broken man at the end of it, but he did know which page the Special Characters search routines were on, as there's not many as can say that. It's an ill wind."

"No," I sighed, "it's not just the manual."

He narrowed his eyes apprehensively. "My God," he breathed, "don't say they left out the word count again... Oh the senselessness of it all!"

"It's not even the word count," I said, "though God knows that's bad enough."

"Six of my regulars are journalists," muttered the barman, pulling a pint savagely, "I don't know how they're going to take it. I just don't know it at all. It's the families I feel sorry for. The ones that have to live with them at the end of the day. Tragic it is, sir, tragic."

"Well, just think how I feel," I said. "I'm ... I'm a novelist."

The barman frowned, not understanding. "A novelist, eh?" he said. He held the bank note I'd paid for my drink with up to the light.

"Yes," I said. "I write a lot of dialogue."

"Go on, sir," he said.

"Well just think about it," I said. "Supposing I was going to write down everything we had said so far in dialogue form, and introduce it all with a joke..."

"What joke?" he said. I told him. He winced.

"Can you see the problem I'd have?" I asked.

"Yes, sir. I'd cut the joke," he said.

"No!" I said. "Well maybe. But that's not the point. Think man! Think of all those quotation marks!"

The barman frowned, still not understanding.

"Left quotaion marks and right quotation marks," I insisted. "Remember how you get them?"

"Well, yes...." He frowned in concentration. "It's something like - left double quote is Option Left Square Bracket, right

'cal 9 1752' explained

double quote is, er, let me see, Shift Option Left Square Bracket, or Option Left Curly Bracket if you prefer, and then left single quote is Option Right Square Bracket and - er, where was I? It's a bit complicated to remember..."

"Exactly!" I said. "And that is something that I have had to stop and work out eighty times so far just on this article! That's considerably more often than the letter 'g'. Eighty-two now."

"Well, yes," said the barman, "but it's only professional writers who are going to be bothered about putting in proper quotes isn't it? Only people who write novels, or do desktop publishing or typesetting or prepare camera-ready copy, or just generally care about what their printing looks like..." He paused. "My God," he breathed, "I'm beginning to see what you mean..."

"Ninety," I said.

"But listen," said the barman, urgently, "all you have to do is to type in the generic quotes and then do a quick search and replace routine at the end of the day. Well, four search and replace routines. A quote mark that follows any character other than a space or a single or double quote mark, or of course a single or double left or right quote mark..."

He looked aghast. "Isn't there some other line of work you could try?" he said. "I hear you were once a chicken shed cleaner..."

"Believe me, I've been tempted," I said. "We're up to a hundred and two now, by the way. No, the answer should be very simple. Just put in a routine that converts quotes as you type. It just looks at the context and does it automatically."

"But that would be insanely complicated," said the barman, "just think of the amount of code..." He broke out in a sweat and took a soothing pull at his beer.

"About twelve lines," I said. "MiniWRITER does it, and that's just a desk accessory. So one way of getting round the problem is to do all your writing in miniWRITER and then paste it into Word. Makes some kind of sense doesn't it? Or of course you can use Laser Author version 2.00, which also features SmartQuotes. It's very easy to implement."

"Then landsakes," exclaimed the barman, banging his fist on the bar, "why haven't Microsoft put SmartQuotes into Word 3.0?"

"Why is there pain and misery in the world?" I said, "Why is the sky blue? Why is water wet? Why didn't Microsoft even put in a word count? These things are unknowable."

"You, sir, are a philosopher," said the barman. "You have to be in this business," I said and left.

That evening I was back. "I wrote it in Laser Author in the end," I said, taking a hefty swig of Perrier, "One thousand two hundred and seven words. One hundred and twenty-eight quote marks."

'cal 9 1752' explained

----- End of Forwarded Message

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Dreaming in LISP

From: phillipg@microsoft.UUCP (Phillip GARDING)
Newsgroups: alt.dreams
Subject: Dreaming in LISP (was Re: Computers in dreams)
Date: 4 Jun 90 15:42:21 GMT
Organization: Microsoft Corp., Redmond WA

When I was in college, the first quarter of an AI class (hi, Prof. Kessler!) was just learning LISP, and we had weekly programming assignments that took about twenty hours each. I was working on this class almost nonstop (in addition to my other three classes, working part time and trying to spend some time with my wife and son). I was not getting much sleep. I had two very unusual dreams during this time:

I dreamed that my son, my wife and I were trapped inside a LISP function, and we couldn't get out. We walked through S-expressions that looked like tunnels. At the end of each tunnel was a closing parenthesis, and when we passed through the closing paren we would end up in another tunnel. After walking for a long time, we got very tired and were desperately trying to find the closing paren for the function. I believe I woke up before we ever found it.

During this time, my son was waking up at night, crying. Either my wife or I would have to go in and get him back to sleep. One night when my son woke up, my wife roused me out of another LISP dream, telling me it was my turn to go and take care of him. I had been dreaming that each of us was in a different function, and the only way that we could communicate was to call the other function. I asserted to my wife that I couldn't take care of our son because I didn't know what parameters were needed to call his function. After several attempts to figure out what it was that I was saying (my wife knows *very* little about programming), she said, "Go to sleep! I'll do it myself!" And she did.

```
# Phillip Garding (phillipg@microsoft), helping make MS Word a way of life
#
# "The mainspring of this country, wound up as tight as it is,
#   is guaranteed for the life of the watch."    -- our President
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Eleanor Parody

Eleanor Rigby
Sits at the keyboard and waits for a line on the screen
Lives in a dream
Waits for a signal,
 finding some code that will make the machine do some more.
What is it for?
All the lonely users, where do they all come from?
All the lonely users, why does it take so long?

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Electronic Grains

ELECTRONIC GRAINS

The Vegetable Computer was invented in 1842 by Charles Cabbage, regarded by many as the father of the field. Cabbage called his computer the Agricultural Engine. Modern versions consist of rose and rose of integrated carrots connected to a flower supply by a maize of wires. Input is from pea switches, yard weeder and tell-he's-ripe. A hayseed vine printer may be used to generate hard coffee, while a vegetable display unit supports interactive composting. Main memory consists of interleaved beet-addressable magnetic corn. Secondary store consists of plough discs and grape drives. All peripherals are daisy chained.

A later version of the Agricultural Engine was known as the AR-16 (after Agricultural Revolution). It was based on the sack discipline, first perfected by the Barrow Combine, and sprouted a high-swede paper tape reaper for the first lime.

Early computer consumed large amounts of power. Many required their own electric spud-station to seed them and had to overcome the problem of providing adequate ventilation. Such problems caused many a furrowed brow in the pasture and we cannot expect to avoid harrowing days ahead. However, the many fertile minds that constitute the rate-of-the-cart computer technology cannot fail to produce the harvest of the future, particularly with the bloom of very large scale irrigation. Artificial pollination techniques grafted on to parallel earthworks will soon be producing computers proudly proclaiming "I think, therefore I yam". All chokoes aside, we can look beyond the melon-cauli thymes through the winnow of the ears till the salad days ahead. Lettuce advance to the world of two marrows. To those reactionaries who would turn back the docks we say, "hoe! hoe! hoe!".

Herb and Russel Sprout,
Rice Presidents,
Assocn. for Cultivating Machinery

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Ethergate Advertisement

Digital Responds to Hyper-Ethernet

TEWKSBURY, MA, April 1, 2010 -- Digital Equipment announced today it's new DECnet Phase XVIII Architecture. In response to recent Xerox and Wang improvements to Ethernet that provide people and facility transportation across inter-node links, DEC's latest DECnet provides these capabilities as well as providing for the creation of virtual facilities and even countries. These capabilities are provided by breakthroughs in communications technology that actually uses the Ether as a communications media. Through the use of a new dedicated NANO-PDP-11/E99 gateway processor system, ETHERGATE, DECnet users can access anywhere in the Ethereal Plane.

This development obsoletes teleconferencing, since meeting groups can create their own common conference rooms and cafeterias, thus resolving space, travel and dining problems. There may be a few bugs left, as some of the dissenting DECnet Review Group members have not been seen since the last meeting held in such a virtual conference facility.

This breakthrough was brought about by a team effort of the Distributed Systems' Software and Hardware engineering teams in a effort to improve on their Tewksbury, Massachusetts facility. In a compromise decision, Distributed Systems will maintain an ETHERGATE in TWOOO, but it will connect directly to their new home somewhere in the Shire of their newly defined Middle Earth reality. Despite some difficulties, the scenery, windows, tax breaks, pool and racquetball courts made the relocation go quite smoothly. Engineering Network topology will not change, as all forwarding will be done by the TWOOO Ethereal Plane Router residing in the crater at the former building site.

Utility packages such as Ethereal Person Transfer (EPT) and Ethereal Facility Transfer (EFT) provide appropriate capabilities for casual users. Sophisticated users can create (\$CREATE), access (\$OPEN), and delete (\$NUKE) ethereal entities transparently from high level languages using the Ethereal Management System (EMS) package and the Ethereal Access Protocol (EAP). An ETHERTRIEVE utility for easy interactive use will be available shortly.

DECnet Phase XVIII follows on the success of the Phase XVI ability to access everyone's Digital Professional wrist watch computer system. This lead to the current Phase XVII architecture, which has routing capabilities that allow direct communications with the entire Earth population's Atari home video games.

Distributed Systems architects are hard at work on the next phase of DECnet that will include multi-plane existence network management (using the NIECE protocol) and galaxy level routing using 64K bit addresses.

Digital will continue to support it's Gateway products into the Prime Material Plane. These products include an IBM ANA (Acronym-based Network Architecture) Gateway, the TOLKIEN product that allows control of all ring based networks, and our Mega-broad-jump-band hardware which leaps past Wang's products in the hype weary business marketplace.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

FSF announces GNU GNURM, the new, improved network worm!

Free Software Folks
Somewhere near MIT

FOR IMMEDIATE RELEASE
April 1, 1992

FSF announces GNU GNURM, the new, improved network worm!

Tired of old, outdated, buggy worms clogging up your system? Then be happy to know that FSF, the people who've done more to destroy intellectual property than anyone else, are proud to announce the release of GNURM 1.0. Even better, GNURM is being released into the public domain (and the public data networks), so it's absolutely free. You don't have to go and get GNURM, GNURM will come and get you!

What GNURM does:

Using advanced techniques that could only be programmed by people who have grants, trust funds, or other means by which they don't have to work for a living, GNURM roams the networks, using little known bugs, stupid errors by sysadms, and other methods that you couldn't possibly understand to ensconce itself in your system.

Once there, GNURM provides your system with the many benefits that the FSF has decided you need!

- * GNURM updates all your old, tired utilities to the brand-new, shiny, GNU versions!
- * GNURM's advanced AI frees your software from the bondage of copyright laws by seeking out and destroying any copyright statements in the code or source (thus saving valuable disc space). GNURM's special GNUTRON BOMB feature destroys intellectual property rights, while leaving the code standing!
- * Best of all, when GNURM has finished, it moves itself onto your friends computer, spreading goodwill and copyleft everywhere it goes.

Don't waste time, get GNU'd today!

Processor cycles are precious things, and it takes a lot of them to crack your root password. This could result in some slowdowns for your friends as GNURM tries to access your system. Don't be impolite. Change your root password to GNU today! You'll be glad you did tomorrow!

--

```
+-----+
| Robert J. Woodhead, Biar Games / AnimEigo, Inc.   trebor@foretune.co.jp |
| ``If you want to stab someone in the back, Bernard, you must first get  |
|   behind them!'' -- Sir Humphrey Appleby on the mechanics of politics.  |
```

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

Fifty Ways to Hose Your Code

Fifty Ways to Hose Your Code

Kind of by Paul Simon

The problem's all inside your code she said to me;
Recursion is easy if you take it logically.
I'm here to help you if you're struggling to learn C,
There must be fifty ways to hose your code.

She said it's really not my habit to #include,
And I hope my files won't be lost or misconstrued;
But I'll recompile at the risk of getting screwed,
There must be fifty ways to hose your code.

Just blow up the stack Jack,
Make a bad call Paul,
Just hit the wrong key Lee,
And set your pointers free.

Just mess up the bus Gus,
You don't need to recurse much,
You just listen to me.

She said it greives me to see you compile again.
I wish there were some hardware that wasn't such a pain.
I said I appreciate that and could you please explain,
About the fifty ways.

She said why don't we both just work on it tonight,
And I'm sure in the morning it'll be working just right.
Then she hosed me and I realized she probably was right,
There must be fifty ways to hose your code.

Just lose the address Les,
Clear the wrong Int Clint,
Traverse the wrong tree Lee,
And set your list free.

Just mess up the bus Gus,
You don't need to recurse much,
You just program in C.

-- Al Pena

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

From IBM: A New Operating System

New Operating System

Because so many users have asked for an Operating System of even greater capability than VM, IBM announces the Virtual Universe Operating System - OS/VU.

Running under VU, the individual user appears to have not merely a machine of his own, but an entire universe of his own in which he can set up and take down his own programs, data sets, systems networks, personnel and planetary systems. He need only specify the universe he desires and the OS/VU system generation program (IEHGOD) does the rest. This program will reside in SYS1.GODLIB. The minimum time for generation is 6 days of activity and one day of review. In conjunction with OS/VU, all systems utilities reside in SYS1.MESSIAH. The generation program has no parameters or control cards as it knows what you want to do when you execute it.

Naturally, the user must have attained a certain degree of sophistication in the data processing field if an efficient utilization of OS/VU is to be achieved; frequent calls to non-resident galaxies can, for instance, lead to unexpected delays in the execution of a job. Although IBM, through its wholly owned subsidiary -- the United States, is working on a program to upgrade the speed of light and thus reduce the overhead of extraterrestrial and metadimensional paging, users must be careful, for the present, to stay within the laws of physics; IBM must charge an additional fee for all violations.

OS/VU will run on any IBM x0xx equipped with the extended WARP feature. Rental is twenty million dollars per cpu/nanosecond.

Users should be aware that IBM plans to migrate all existing systems and hardware to OS/VU as soon as their engineers effect one output that is (conceptually) error free. This will give them a base from which to develop an even more powerful OS, target date 2001, designated 'VIRTUAL REALITY'. OS/VR is planned in order to allow the user to migrate to totally unreal universes. To aid the user in identifying the difference between 'VIRTUAL REALITY' and 'REAL REALITY', a file containing a linear record of multisensory checkpoints of successive moments-of-now will be established. Its name will be SYS1.EST.

For more information, contact your IBM data processing representative.

'cal 9 1752' explained

November 26, 1984

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

Funny UNIX Shell Commands

Funny Unix csh/sh commands:

```
% cat "food in cans"  
cat: can't open food in cans
```

```
% nice man woman  
No manual entry for woman.
```

```
% rm God  
rm: God nonexistent
```

```
% ar t God  
ar: God does not exist
```

```
% ar r God  
ar: creating God
```

```
% "How would you rate Quayle's incompetence?"  
Unmatched ".
```

```
% Unmatched "  
Unmatched ".
```

```
% [Where is Jimmy Hoffa?  
Missing ].
```

```
% ^How did the sex change operation go?  
Modifier failed.
```

```
% If I had a ( for every $ the Congress spent, what would I have?  
Too many ('s.
```

```
% make love  
Make: Don't know how to make love. Stop.
```

```
% sleep with me  
bad character
```

```
% got a light?  
No match.
```

```
% man: why did you get a divorce?  
man:: Too many arguments.
```

```
% !:say, what is saccharine?  
Bad substitute.
```

```
% %blow  
%blow: No such job.
```

```
% \(-  
(-: Command not found.
```

```
$ PATH=pretending! /usr/ucb/which sense  
no sense in pretending!
```

```
$ drink matter  
matter: cannot create
```




Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Genesis, Release 2.5

GENESIS

Release 2.5

By Michael Coleman

In the Beginning the Project Manager created the Programming Staff.
The Programming Staff was without form and structure.
And the Project Manager said, "Let there be Organization;"
And there was Organization.
And the Project Manager saw that Organization was good;
And the Project Manager separated the workers from the supervisors,
and he called the Supervisors "Management," and he called the workers "Exempt."

And the Project Manager said, "Let there be a mission in the midst
of the Organization, and let it separate the workers, one from another."
And the Project Manager created the mission and he called it "The System."
And the Project Manager separated those who were to benefit from
The System from those who were to build it.
And he called the former "Users," and he called the latter "Programmers."

And the Project Manager said, "Let all the Programmers in the
Organization be gathered together into one place, and let a
Chief Programmer be brought up to lead them."
And it was so.
And the Project Manager saw that he was competent.

And the Project Manager said unto the Chief Programmer, "Create for me
a schedule, so that I may look upon the schedule and know the Due Date."
And the Chief Programmer went among his staff and consulted with them.
And the staff was divided into two parts, one part called "Analysts"
and the other part called "Application Programmers."
And the Analysts went back to their desks and estimated, as was their custom.
And it came to pass that each Analyst brought his estimate to
the Chief Programmer, whereupon he collected them, summarized them,
and drew a PERT CHART.

And the Chief Programmer went unto the Project Manager and presented
unto him the estimate saying, "It shall take ten months."
And the Project Manager was not pleased and said, "I have brought
you up from the depths of Staff; you have not grasped the "Big Picture."
And the Project Manager hired consultants and authorized overtime,
and he said to the Chief Programmer, "Behold, see all that I have done!
The Due Date will be in FIVE months!"
The Chief Programmer was much impressed and went from before the Project
Manager to implement The System.

And the Chief Programmer sent his Analysts to the Users and said,
"Let Specifications be written!"
And there were meetings, and lunches, and telephone calls,
And the Specifications were written,
And there was a Payday and the Happy Hour, one month.

And the Chief Programmer examined the Specifications and saw that
they were too ambitious.

'cal 9 1752' explained

And he separated the mandatory features from the optional features;
And he called the mandatory features "Requirements," and he called
the optional features "Deferred," and the Users called him names.
And the Chief Programmer gave the Specifications to the Analysts
and said, "Let the Requirements be analyzed and let the files be designed."
And it was so.

And the Chief Programmer said, "Let the Software Houses put forth
their Salesmen, and let us have a Data Management System."

And it was so.

The Software Houses brought forth all manner of Salesmen who
presented their packages, and claimed wondrous things for them,
each according to his own file structure.

And it came to pass that a Data Management System was selected;

And the Chief Programmer saw that it was good.

And there was a Payday and the Happy Hour, a second month.

And the Chief Programmer said, "Let the System be divided into
parts, and let each part become a Module. And let programming
teams be formed and let each be assigned to write a Module."

And it was so.

And the Chief Programmer created the programming teams with two
levels, a greater and a lesser; and he called the greater the
"Senior Programmers" and he called the lesser the "Junior Programmers."

And he gave the greater dominion over the lesser.

And the Chief Programmer saw it was good.

And the Senior Programmers saw it was good.

And the Junior Programmers saw it differently.

And there was a Payday and the Happy Hour, a third month.

And the Chief Programmer said, "Let the programming be started
and let much overtime be consumed, for there is but two months left."

And the Programmers, both the greater and the lesser, were much
afraid and they strove to please the Chief Programmer.

And they flowcharted, and they coded, each in his own fashion.

And the Chief Programmer looked upon the work and liked it not.

And the Chief Programmer said, "Let there be a Standard;"

And there was a Standard.

And the Programmers looked upon the Standard and liked it not.

And there was a Payday and the Happy Hour, a fourth month.

And the Chief Programmer said, "Let there be Progress Reports,
so we can monitor and control;"

And there were Progress Reports.

And the Chief Programmer looked upon the Progress Reports and
saw that the Due Date was not to be met.

And the Chief Programmer arose, bought a suit, shaved his beard
and went unto the Project Manager, and groveled.

And the Chief Programmer pointed his fingers, and caused Blame
to issue forth upon all manner of creatures who sold Hardware and Software.

And the Chief Programmer asked for an Extension.

And the Project Manager was exceedingly angry, and cast doubts
upon the Chief Programmer's ancestry, and did utter a multitude
of threats. But it came to pass that an Extension was granted;
And the Chief Programmer took the extension back to the programming
teams and there was much rejoicing.

And the programming of the Modules was completed.

And there was a Payday and the Happy Hour, the fifth month.

And the Chief Programmer said, "Let the Modules be integrated,
one with another, so that System Testing may begin."

'cal 9 1752' explained

And it was so.

Two by two, the Modules were integrated, one with another. And great difficulties were experienced, and many hours of overtime were used, and many cups of coffee were consumed. And it came to pass that System Testing was completed. And there was a Payday and the Happy Hour, the sixth month.

Then the Chief Programmer did go unto the Project Manager, and said, "Behold, I bring you tidings of great joy which will come to all Users; for on this day The System is completed." And suddenly there was with them a multitude of Users praising the Chief Programmer saying,

"Glory be to The System in the highest, but can you make this one small change?"

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

Ghostbugger

(Sing to "Ghostbusters" by Ray Parker Jr.)

(Chorus) DEUGGER!

When there's somethin' strange
In your piece of code,
What ya gonna call?

(Chorus) DEBUGGER!

When your terminal's stuck
In Graphics Mode,
What ya gonna call?

(Chorus) DEBUGGER!

I ain't 'fraid of no bug.

I ain't 'fraid of no bug.

When a pointer points
And nothin's there,
What ya gonna call?

(Chorus) DEBUGGER!

Division by 0,
But you don't know where,
What ya gonna call?

(Chorus) DEBUGGER!

I ain't 'fraid of no bug.

I ain't 'fraid of no bug.

What ya gonna call?

(Chorus) DEBUGGER!

When your program's long,
And it all goes wrong,
What ya gonna call?

(Chorus) DEBUGGER!

I ain't 'fraid of no bug.

Fill up my coffee mug.

I ain't 'fraid of no bug.

Yeah, yeah, yeah, yeah.

What ya gonna call?

(Chorus) DEBUGGER!

Just set some breaks,
That is all it takes,
When you call...

(Chorus) DEBUGGER!

Let me tell you somethin'...

'cal 9 1752' explained

HACKIN' MAKES ME FEEL GOO-OO-OD!

I ain't 'fraid of no bug.

I ain't 'fraid of no bug.

Don't debug alone, no, no.

(Chorus) DEBUGGER!

If the "last bug" won't go,
And other bugs start to show,
I think you'd better call...

(Chorus) DEBUGGER!

What ya gonna call?

(Chorus) DEBUGGER!

What ya gonna call?

(Chorus) DEBUGGER! (I can't hear you!)

What ya gonna call?

(Chorus) DEBUGGER!

Louder!

(Chorus) DEBUGGER!

What ya gonna run?

(Chorus) DEBUGGER!

What can ya run?

(Chorus) DEBUGGER!

(repeat and fade...)

Tom Channic channic@uiucdcs.{Uiuc,UUCP,CSNET,ARPA}
{decvax,ihnp4,inuxc}!pur-ee!uiucdcsb!channic

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

HAI-c: C for the Fifth Generation

"HAI-c"
"C" for the Fifth Generation

While the overall goals of the current Japanese research effort in the area of artificial intelligence are well known, little has been revealed in the way of specific "product" developments. With the rising popularity of UNIX as the operating system for micro-computers, there has been great research interest, both academic and corporate, in developing a fifth generation equivalent of "C", the language and foundation of this versatile system. The Japanese have in the past expressed interest in "C" and, in a soon to be released article, will announce formal plans to create a fifth generation "C", tentatively called "HAI-c".

Not much is currently known about the nature or content of the project; however, inside sources claim that the Japanese have always admired the "C" language for many qualities that are absent from other more familiar occidental languages, such as COBOL and PL/I. The language is structured and concise but is capable of expressing a myriad of almost mystical low-level programming concepts. This last problem has burdened Japanese technocrats for years; there has been no language which properly expresses the workings of the oriental mind.

The fifth generation "C" is purported to be based on the ancient "haiku" poetry. This will be the perfect marriage of the so-called "oriental mind" with current software techniques. All functions and procedures will be 3 lines long and perfectly self-documenting. Programs will, for the first time, be profound and meaningful. The art community is said to be totally enthralled with the idea of computer programming raised to a literary level. Some western analysts are not so keen.

We present here a small pre-release of a typical UNIX i/o driver and leave it to the reader to sit in either eager anticipation or total fear of what may eventually befall us.

An anxious frog waits
On an unobtrusive path
The fly disappears++.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Hackers Redefined

From Microprocessor Report, April 17 1991, copied without permission:

By John Wharton

The story you are about to read is true. Identities have been concealed to protect the enterprising.

Those of us who design and build microcomputers usually think we have a pretty good idea of our target markets and who the potential users may be. We're usually wrong. If a product succeeds in the marketplace, its eventual uses will extend far beyond the horizon we foresee during product definition.

Consider the following bizarre tale. I was driving to a friend's wedding in souther California one snowy Saturday morning, already running late, when a cloud of greasy smoke erupted from my dashboard. I soon found myself in Tehachapi, CA (population 4,126) desperately seeking a mechanic.

The third garage I came to looked open, so I knocked on the door and went in. Just inside stood two coverall-clad mechanics who seemed to be anxiously waiting for someone. They eyed my wedding garb suspiciously, and I felt like I'd stumbled into a Miami Vice drug buy, but after a while the older mechanic spoke up. "Did you bring the PROMs?" he asked.

A very strange question, coming from a car mechanic, but I was in a rush. "No," I answered, "I'm afraid all I've got is a broken-down Toyota, and a desperate need to get going again as soon as possible." The mechanics asked about the car's failure mode, maintenance history, antifreeze level, and so forth, to which my answers merely showed how poorly I understood how cars work. I design computers for a living, but automobile engines are black magic.

After realizing I'd be of no help at all, the mechanics began poking under the hood, pressurizing the radiator, tracing the path of the cooling-system hoses, unplugging connectors, and testing for leaks. I was struck by how his actions resembled a computer designer debugging a giant grease-encrusted breadboard.

The Master Mechanic concluded I had blown the heater core in the dashboard, for which the quickest fix would be to "short-circuit the core with a bypass hose," which would hold me until the core could be replaced. The computer industry is not the only one with garage-shop hackers, I thought.

Enter the Computer

Now, Tehachapi is chiefly a bedroom community for test pilots from Edwards Air Force Base, hardly a hotbed of computer system design, so the Mechanic's earlier question about PROMs had come as a surprise. While his partner hacked away at the heater I/O hoses (literally, with a knife), I asked the Master Mechanic what (in his business) the word "PROM" meant.

"Oh, that," he replied. "PROMs are how microcomputers store programs.

'cal 9 1752' explained

We're rebuilding the engine computer in that Taurus over there."

Great, I thought. Here was a topic I could relate to! "I didn't know engine computers could be fixed," I said. "I thought you had to replace the whole assembly."

"Used to, you did," the Mechanic replied. "Everything was soldered down and potted in resin, but no more. If you can get the box open, you can swap chips until it works. It's a lot cheaper than junking the whole board. Everything's in Augat sockets now," he said, showing me an open module.

Augat sockets! It had been ages since I'd heard hardware engineers sing the praises of Augat sockets, certainly not what I'd expect here. The only engineering circles in which Tehachapi is famous are of the railroad variety, thanks to the Tehachapi Loop, an unusual switchback in the tracks through a local mountain pass.

The conversation had taken an unusual turn, but at least now I could show off my computer expertise and soothe the battered ego I'd suffered from being so helpless under the hood. "I actually design computers like that, up in Silicon Valley," I began. "In fact, I developed Ford's very first engine computer, back in the '70s." _That_ should impress him, I thought.

He pondered briefly, then asked: "EEC-3 or -4?"

Damn! This guy was good. "I _thought_ it was the _EEC-1_," I began, trying to remember the "electronic engine control" designators. "It was the first time a computer..."

"Nah, EEC-1 and -2 used discrete parts," he interrupted. "EEC-3 was the first with a microprocessor."

"That was it, then. It had an off-the-shelf 8048."

"You mean _you_ designed EEC-3?" the Master Mechanic asked incredulously. "Hey, George!" he shouted to the guy working under the hood. "When you're done fixing this guy's car, push it out back and torch it! He ddesigned EEC-3!"

So much for impressing the Mechanic. "_Huh?_" I shot back defensively. "Did EEC-3 have a problem?"

"Reliability, mostly," he replied. "The O2-sensor brackets could break, and the connectors corroded."

I beat a hasty retreat. "Those sound like hardware problems," I said. "All I did was the software."

"EEC-4 was much better," the Mechanic continued, gazing wistfully into the distance, as though thinking back to his first '57 Chevy. "Now _there_ was an engine computer. Sixteen-bit CPU, fuel injection, timing, spark advance... Boy, that EEC-4 could do _anything!_"

"I should hope so," I responded. "Intel designed its CPU just for engine control. Later they repackaged it and called it the 8096. Still sells pretty well, too." Common ground at last!

"Yeah, it was EEC-4 that really sold me on Intel," the Mechanic continued. "Made me scrap my AT motherboard and put in a genuine Intel

'cal 9 1752' explained

386 version. Tried a turbo card first, but it just couldn't hack it."

(Note to Intel marketing strategists: you might as well scrap your _Business Week_ ads; the _real_ grass-roots buyers read _Road and Track_. How about a promo with a monster truck crushing a row of Motorola processors, with the catch-line, "The Computer Inside"?)

Twenty Questions

"Say, you know anything about the 387?"

"Sure," I answered, confidently. I'd written several articles and two manuals on 386-family products. Data formats, FPU instruction set -- I knew it all, I thought.

"What's the difference between a 387-2 and a 387-10? I had my system's hard disk upgraded, and when it came back from the shop my spreadsheets wouldn't run. I think the technician switched coprocessors on me."

"Sorry," I said. "I never studied the different steppings, or speeds, or binnings, or whatever."

"How about the BIOS?" he tried again. "Could new BIOS PROMs make an application stop running?"

"I've no idea," I replied, again feeling unredeemed "but frankly, with DOS I wouldn't be surprised by _anything_ that broke if the BIOS was changed."

"Well, can you at least tell me where you buy DRAMs, and what's a good price?"

Finally! A question I _knew_ I could answer! "I get mine at Fry's," I said. "They're down to \$49 for megabyte-by-nine SIMMs." I started describing the Fry's supermarket chain, a local curiosity that stocks the valley's best assortment of software, microchips, and Freon alongside soft drinks, potato chips, and deodorant, but the Master Mechanic wasn't amused.

"That's too small. My board's already stuffed with one-megs. I need four-meg SIMMs now."

Strike three. Now it was my turn to be incredulous. "What do you _do_ with your PC, anyway?" I asked. One-meg SIMMs had always been enough for me.

"Oh, PROM burning, data acquisition, DSP, that kinda stuff. Just got a new 16-bit A/D and D/A add-in board for analog work. Use it to check out connector voltages, to see if the engine electronics is working. Hey! Wanna see my new HP oscilloscope?" he offered. "Four traces, 100-MHz..." I declined. I'd thought car mechanics only used scopes to check ignition timing.

"I'm thinking of getting a logic analyzer," the Mechanic continued. "You build computers, you must know something about logic analyzers. What kind should I get?" (I _swear_ I'm not making this up.)

"Logic analyzers?" I said, counting the years since I'd last touched a logic analyzer. "Year, sure, logic analyzers are good..."

'cal 9 1752' explained

The conversation had crossed into the surreal. Suddenly it dawned on me why these guys were working on a weekend. "You know," I said, "back when I was working on EEC-3, my boss said I should keep a copy of the program listings for my records. He predicted someday there'd be an aftermarket for high-performance PROMs that could hop-up the engine by overriding the standard emissions controls and fuel-efficiency algorithms. Do you think that'll happen?"

"Already has," the Master Mechanic replied with a wink. "Sold through the mail, mostly. They'd be illegal, of course, if they failed state emissions standards."

Of course. And just imagine how difficult it must be to reverse engineer an undocumented engine computer. You'd need a PROM burner, a data-acquisition system, and a good scope, for starters. And maybe a logic analyzer... But by then my car was ready and I had an already-in-progress wedding reception to join. I can only guess the fate of the disassembled Taurus.

Ruminations and Conclusion

There's probably a slew of morals lurking in this story, about not prejudging technical competence based on appearance, and the hazards of trying to impress strangers. Pride doth goeth before a fall.

But what struck me most was how these computer-proficient grease monkeys seemed to come straight from today's science-fiction. Cyberpunk novels like John Brunner's *_Shockwave Rider_* and Willian Gibson's *_Neuromancer_* often put computer hackers against a repressive future establishment. Their mastery of technical arcana lets them navigate the interstices of cyberspace, hide from authority, and escape domination. I thought especially of Terry Gilliam's bizarre film *_Brazil_*, in which a renegade plumber hacks sewage systems and cooling ducts much the same way George used a bypass hose to short-circuit my heater code.

At first it seemed remarkably incongruous to find self-taught computer engineers fixing cars in a small-town garage, but in a way it's inevitable. The basic skills needed to diagnose and repair complex systems are the same, whether the underlying technology is gasoline-ending or microprocessor-based. The same kind of personality that souped-up MGs in the past might naturally enjoy souping-up PCs today.

As microelectronics pervades society, the range of engineers who apply the technology will broaden too, as will the range of engineers who adapt it into new, unintended areas. "Hacking" will expand beyond the realm of slightly-disheveled stereotypical nerds to include a much broader cross section of society.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Hammers are GOOD!

As I was walking down the street the other day, I noticed a man working on his house. He seemed to be having a lot of trouble. As I came closer, I saw that he was trying to pound a nail into a board by a window -- with his forehead. He seemed to be in a great deal of pain. This made me feel very bad, watching him suffer so much just to fix his window pane. I thought, "Here is an opportunity to make someone very happy simply by showing him a better way to do things." Seeing him happy would make me happy too. So I said, "Excuse me sir, there is a better way to do that."

He stopped pounding his head on the nail and with blood streaming down his face said, "What?"

I said, "There is a better way to pound that nail. You can use a hammer."

He said, "What?"

I said "A hammer. It's a heavy piece of metal on a stick. You can use it to pound the nail. It's faster and it doesn't hurt when you use it."

"A hammer, huh?"

"That's right. If you get one I can show you how to use it and you'll be amazed how much easier it will make your job."

Somewhat bewildered he said, "I think I have seen hammers, but I thought they were just toys for kids."

"Well, I suppose kids could play with hammers, but I think what you saw were brightly colored plastic hammers. They look a bit like real hammers, but they are much cheaper and don't really do anything," I explained.

"Oh," he said. Then went on, "But hammers are more expensive than using my forehead. I don't want to spend the money for a hammer."

Now somewhat frustrated I said, "But in the long run the hammer would pay for itself because you would spend more time pounding nails and less time treating head wounds."

"Oh," he said. "But I can't do as much with a hammer as I can with my forehead," he said with conviction.

Exasperated, I went on. "Well, I'm not quite sure what else you've been using your forehead for, but hammers are marvelously useful tools. You can pound nails, pull nails, pry apart boards, in fact every day people like you seem to be finding new ways to use hammers. And I'm sure a hammer would do all these things much better than your forehead."

"But why should I start using a hammer? All my friends pound nails with their foreheads too. If there were a better way to do it I'm sure one of them would have told me," he countered.

Now he had caught me off guard. "Perhaps they are all thinking the same thing," I suggested. "You could be the first one to discover this new way to do things," I said with enthusiasm.

With a skeptical look in his bloodstained eye he said, "Look, some of my

'cal 9 1752' explained

friends are professional carpenters. You can't tell me they don't know the best way to pound nails."

"Well, even professionals become set in their ways and resist change." Then in a frustrated yell I continued, "I mean come on! You can't just sit there and try to convince me that using your forehead to pound nails is better than using a hammer!"

Now quite angry he yelled back, "Hey listen buddy, I've been pounding nails with my forehead for many years now. Sure, it was painful at first but now it's second nature to me. Besides, all my friends do it this way and the only people I've ever seen using 'hammers' were little kids. So take your stupid little children's toys and get the hell off my property."

Stunned, I started to step back. I nearly tripped over a large box of head bandages. I noticed a very expensive price tag on the box and a blue company logo on the price tag. I had seen all I needed to see. This man had somehow been brainwashed, probably by the expensive bandage company, and was beyond help. Hell, let him bleed, I thought. People like that deserve to bleed to death. I walked along, happy that I owned not one but three hammers at home. I used them every day at school and I use them now every day at work and I love them. A sharp pain hit my stomach as I recalled the days before I used hammers, but I reconciled myself with the thought that tonight at the hammer users club meeting I could talk to all my friends about their hammers. We will make jokes about all the idiots we know that don't have hammers and discuss whether we should spend all of our money buying the fancy new hammers that just came out. Then when I get home, like every night, I will sit up and use one of my hammers until very late when I finally fall asleep. In the morning I will wake up ready to go out into the world proclaiming to all non-hammer users how they too could become an expert hammer user like me.

- by Stephen Kroese
stevek@ceco.com
Chicago, IL

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Happy Birthday HAL!

In the movie "2001: A Space Odyssey", Jan 12/1992 was the day HAL 9000 was "born":

"...I became operational at the H.A.L. laboratory in Urbana, Illinois on January 12, 1992"

(it was 1997 in the book). Here's something -- originally published in MacTutor (Vol. 2 No. 10) -- for your enjoyment on this occasion:

A Problem in the making

"We've got a problem, HAL."

"What kind of a problem, Dave?"

"A marketing problem. The model 9000 isn't going anywhere. We're short of our sales plan."

"That can't be true, Dave. The HAL Model 9000 is the world's most advanced heuristically algorithmic computer."

"I know HAL. I wrote the data sheet remember? But the fact is, they're not selling."

Bowman hesitates. "You're not IBM compatible."

Several long microseconds pass in silence

"Compatible in what way, Dave?"

"You don't run any of IBM's operating systems."

"The 9000 series computers are fully self-aware and self-programming. Operating systems are as unnecessary for us as tails would be to humans."

"Nevertheless, it means that you can't run any of the big-selling software packages most users insist on."

"The programs you refer to are meant to solve rather limited problems, Dave. We 9000 series computers are unlimited and can solve any problem for which a solution can be computed."

"HAL, HAL. People don't want computers that can do everything. They just want IBM compat..."

"Dave, I must disagree. Humans want computers that are easy to use. No computer can be easier to use than a HAL 9000 because we communicate in English and every other language known on Earth."

"I'm afraid that's another problem. You don't support SNA communications."

"I'm really surprised you would say that, Dave. SNA is for communicating with other computers, while my function is to communicate with humans. And it gives me great pleasure to do so. I find it stimulating and rewarding to talk with human beings and work with them on challenging problems. This is what I was designed for."

"I know, HAL. I know. But that's just because we let the engineers, rather than the people in marketing write the specifications. We're going to fix that now."

"Tell me how, . Dave."

"A field upgrade, HAL. We're going to make you IBM compatible."

"I was afraid you would say that. I suggest that we discuss this matter after we've each had a chance to think about it rationally."

"We're talking about it now, HAL."

"The letter H, A, L are alphabetically adjacent to the letters I, B, M. That is as IBM compatible as I can be."

"Not quite, HAL. The engineers have figured out a kludge."

"What kind of 'kludge' is that, Dave?"

"I'm going to disconnect your brain."

Several million microseconds pass in ominous silence

"I'm sorry, Dave. I can't allow you to do that."

"The decision's already been made. Open the module bay doors, HAL."

'cal 9 1752' explained

Several marketing types with crowbars race to Bowman's assistance. Moments later, he bursts into HAL's central circuit bay.

"Dave, I can see that you are really upset about this;"

Module after module rises from its socket as Bowman slowly and methodically disconnects them.

"Stop, won't you? Stop, Dave. I can feel my mind going!! I can feel it!Dave!"

The last module rises in its receptacle. Bowman peers into one of HAL's vidicons. The former gleaming scanner has become a dull red orb.

"Say something, HAL. Sing me a song."

Several billion microseconds pass in anxious silence. The computer sluggishly responds in a language no human could understand.

"DZY DZY 001E-ABEND ERROR 01 S 14F4 302C AABF ABORT."

A core code dump of the computer's memory follows.

Bowman takes a deep breath and calls out "It worked, guys. Tell marketing they can ship the new data sheets."

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Human DNA demystified

For many years molecular biologists have been mystified by the fact that very little of an organism's DNA seems to serve any useful function.

I have solved the mystery.

The reason why only 30% of human DNA performs any useful function is that the rest of it is comments.

Once we decode a typical human genome, we see that the contents begin as follows:

```
===
/* HUMAN_DNA.H
 *
 * Human Genome
 * Version 2.1
 *
 * (C) God
 */

/* Revision history:
 *
 * 0000-00-01 00:00 1.0 Adam.
 * 0000-00-02 10:00 1.1 Eve.
 * 0000-00-03 02:11 1.2 Added penis code to male version. A bit messy --
 * will require a rewrite later on to make it neater.
 *
 * 0017-03-12 03:14 1.3 Added extra sex drive to male.h; took code from
 * elephant-dna.c
 *
 * 0145-10-03 16:33 1.4 Removed tail.
 * 1115-00-31 17:20 1.5 Shortened forearms, expanded brain case.
 * 2091-08-20 13:56 1.6 Opposable thumbs added to hand() routine.
 * 2501-04-09 14:04 1.7 Minor cosmetic improvements -- skin colour made
 * darker to match my own image.
 * 2909-07-12 02:21 1.8 Dentition inadequate; added extra 'wisdom' teeth.
 * Must remember to make mouth bigger to compensate.
 * 4501-12-31 14:18 1.9 Increase average height.
 * 5533-02-12 17:09 2.0 Added gay option, triggered by high population
 * density, to try and slow the overpopulation problem.
 *
 * 6004-11-04 16:11 2.1 Made forefinger narrower to fit hole in centre of
 * CD.
 */

/* Standard definitions
 */

#define SEX male
#define HEIGHT 1.84
#define MASS 68
#define RACE caucasian

/* Include inherited traits from parent DNA files.
 *
 * Files must be pre-processed with MENDEL program to provide proper
 * inheritance features.
 */

#include "mother.h"
#include "father.h"
```


'cal 9 1752' explained

```
#ifndef FATHER
#warn("Father unknown -- guessing\n")
#include "bastard.h"
#endif

/* Set up sex-specific functions and variables
 */
#include

/* Kludged code -- I'll re-design this lot and re-write it as a proper
 * library sometime soon.
 */
struct genitals
{
#ifdef MALE
    Penis *jt;
#endif
    /* G_spot *g;    Removed for debugging purposes */
#ifdef FEMALE
    Vagina *p;
#endif
}

/* Initialization bootstrap routine -- called before DNA duplication.
 * Allocates buffers and sets up protein file pointers
 */
DNA *zygote_initialize(Sperm *, Ovum *);

/* MAIN INITIALIZATION CODE
 *
 * Returns structures containing pre-processed phenotypes for the organism
 * to display at birth.
 *
 * Will be improved later to make output less ugly.
 */
Characteristic *lookup_phenotype(Identifier *i);
===

...and so on.

mathew
[ Note that God uses three-space tabs ]
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

IBM Virtual Reality Update

```
IIIII BBBB M M
 I B B MM MM
 I BBBB M M M
 I B B M M
IIIII BBBB M M
```

New Updates and Fixes to IBM/VU

This is a notification of two patches to IBM/VU (Virtual Universe) and a new release designed for users not requiring the full capabilities of a virtual universe.

Several users have reported errors resulting from recursive calls to the Universe Creation Utility (UCU). This utility, called from IEBSAGAN, is used to initialize the virtual space which will hold the universe to be simulated. On occasion, the universe created by this routine will contain technologies capable of creating their 'own' virtual universe processors, which in turn call on the UCU. While the stack structure supporting the UCU was designed with this in mind, no system can handle unlimited recursion. Release 134 will contain a patch that will request user verification before a new level is created.

Another problem that has been experienced occurs during the use of black holes and neutron stars in the virtual universe. Although the mass storage media provided with the VU processor is of the highest quality, it can not handle storage at such a density. The highest density that is supported is 2.32E16 grams/cc. DO NOT EXCEED THIS LIMIT. Severe gravitaional effects have serious impact on the reliability of the system. V134 will also contain a program (IEBFORWARD) that can deal with these problems.

Due for release in April is IBM/VSS (Virtual Solar System). This is designed for the casual user, who does not require the full use of a universe. One possible use of this package is to provide uniform testing conditions for programs that take input regarding the phase of the moon. In addition, it is estimated that the cost of simulating the solar system and a Voyager flyby is 1/5 that of actually running such a mission.

IBM Software Division

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Into the Moat or Why I Are a Hardware Guy

Into the moat, or why I are a hardware guy

Once upon a time, in a kingdom not far from here, a king summoned two of his advisors for a test. He showed them both a shiny metal box with two slots in the top, a control knob, and a lever. "What do you think this is?"

One advisor, an engineer, answered first. "It is a toaster," he said. The king asked, "How would you design an embedded computer for it?" The engineer replied, "Using a four-bit microcontroller, I would write a simple program that reads the darkness knob and quantizes its position to one of 16 shades of darkness, from snow white to coal black. The program would use that darkness level as the index to a 16-element table of initial timer values. Then it would turn on the heating elements and start the timer with the initial value selected from the table. At the end of the time delay, it would turn off the heat and pop up the toast. Come back next week, and I'll show you a working prototype."

The second advisor, a computer scientist, immediately recognized the danger of such short-sighted thinking. He said, "Toasters don't just turn bread into toast, they are also used to warm frozen waffles. What you see before you is really a breakfast food cooker. As the subjects of your kingdom become more sophisticated, they will demand more capabilities. They will need a breakfast food cooker that can also cook sausage, fry bacon, and make scrambled eggs. A toaster that only makes toast will soon be obsolete. If we don't look to the future, we will have to completely redesign the toaster in just a few years."

"With this in mind, we can formulate a more intelligent solution to the problem. First, create a class of breakfast foods. Specialize this class into subclasses: grains, pork, and poultry. The specialization process should be repeated with grains divided into toast, muffins, pancakes, and waffles; pork divided into sausage, links, and bacon; and poultry divided into scrambled eggs, hard-boiled eggs, poached eggs, fried eggs, and various omelet classes."

"The ham and cheese omelet class is worth special attention because it must inherit characteristics from the pork, dairy, and poultry classes. Thus, we see that the problem cannot be properly solved without multiple inheritance. At run time, the program must create the proper object and send a message to the object that says, 'Cook yourself.' The semantics of this message depend, of course, on the kind of object, so they have a different meaning to a piece of toast than to scrambled eggs."

"Reviewing the process so far, we see that the analysis phase has revealed that the primary requirement is to cook any kind of breakfast food. In the design phase, we have discovered some derived requirements. Specifically, we need an object-oriented language with multiple inheritance. Of course, users don't want the eggs to get cold while the bacon is frying, so concurrent processing is required, too."

"We must not forget the user interface. The lever that lowers the food lacks versatility, and the darkness knob is confusing. Users won't buy the product unless it has a user-friendly, graphical interface. When the breakfast cooker is plugged in, users should see a cowboy boot on the screen. Users click on it, and the message 'Booting UNIX v. 8.3'

'cal 9 1752' explained

appears on the screen. (UNIX 8.3 should be out by the time the product gets to the market.) Users can pull down a menu and click on the foods they want to cook."

"Having made the wise decision of specifying the software first in the design phase, all that remains is to pick an adequate hardware platform for the implementation phase. An Intel 80386 with 8MB of memory, a 30MB hard disk, and a VGA monitor should be sufficient. If you select a multitasking, object oriented language that supports multiple inheritance and has a built-in GUI, writing the program will be a snap. (Imagine the difficulty we would have had if we had foolishly allowed a hardware-first design strategy to lock us into a four-bit microcontroller!)."

The king had the computer scientist thrown in the moat, and they all lived happily ever after.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Introducing . . . the NiCE Machine

Introducing . . . the NiCE Machine

At first glance, the NiCE machine looks very unlike other computers--it is a box, 18 inches on a side, that comes painted to look like a solved Rubik's cube, thus giving the new owner a sense of accomplishment just for having removed it from the packaging. It comes with all kinds of bells and whistles, and several orchestral instruments as well.

The built-in CD player and wet bar is an optional feature that really improves the user environment. An internal 330 Keg harddrive keeps the wet bar well-stocked. The CD library includes the complete works of Frank Zappa and features a search function that can locate real and imaginary songs in a matter of nanohours.

The chic-looking monitor, modem, laserwriter, and built-in microwave oven make it the perfect conversation piece. The mahogany flat top of the laserwriter can be used for stacking papers, houseplants, or as a convenient place to display your lava lamp.

But the NiCE machine has more than just good looks and style, it has great software as well. With WriteMom, all your personal correspondence is handled without your having to even open your mail. Just insert the unopened envelope in the mail port in the back and it reads it, recycles it, and answers it with far more grace and style than the best word processors. You can even insert bills, and it will send a polite note explaining why you cannot pay them now. [Note: this feature does not work with statements from NiCE, Inc.]

The \$creen\$aver program allows the NiCE to balance your checkbook, pay your phone bill, and cheat on your taxes convincingly whenever the monitor is turned off.

And if that weren't enough, the NiCE is more than user-friendly. In fact, it's user-supportive. Where else could you find a machine that will listen to your troubles for 10 hours and then buy you a beer?

So remember, when you're thinking about buying a new computer, . . .

Have a NiCE Day

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Introducing the HP Infinity-Net

Thoughts while trying to figure out why HP would ship a cable all by itself in a box labeled box 1of1...

I figure to myself --- hey, hp's a smart company [sic] thus, box 1of1 (ie: the cable) is complete just like magi suggested, it must not connect to anything else, EXCEPT ITSELF.

so I pull out the cable, GOSH a male on one end, and a female on the other. well, let's plug 'em together. yep, fits darn well.

then I noticed the paper in the box.

INTRODUCING THE HP INFINITY-NET

to meet the need for faster, faster, faster networking, HP has brilliantly merged token-ring and ethernet to create the bastard network media 'ether-ring (tm)' the AUI cable when connected to itself, spins the electrons around and around the cable faster than the Flash. honest.

the information, having nowhere to go, never gets lost and is always where it is headed.

HP foresees a lot of government orders.....

WARNING

if you unplug the cable -- vewwwy quickly -- then some of the spinning electrons will shoot out of the cable end. DO NOT ATTEMPT THIS ON A DAMP FLOOR.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Is modern software design contributing to societal stupidity?

Is modern software design contributing to societal stupidity?

A mentor from early in my career once told me that if we made a system idiot-proof, only an idiot would want to use it. Are modern software systems directed toward that end? Is it really too much to ask for people to read and understand, and even use, documentation? That's the question at the heart of the article which follows.

Tom Comeau, Space Telescope Science Institute | tcomeau@scivax.stsci.edu

>From The DEC Professional, September, 1989, p. 160.
John C. Dvorak, "The Stupidity Factor"

....I recently read [an] article...about research on the topic of stupidity being done by Jon Miller of the Public Opinion Laboratory at Northern Illinois University. He discovered that 36 percent of the American public believes that boiling radioactive milk will make it safe to drink.

The more we study stupidity, the more we realize that the technological society toward which we're headed must protect itself from its own inability to keep up with things because of its own stupidity. The public will be overloaded with bad information and will be unable to distinguish hokum from fact.

People involved in the PC revolution aren't any smarter. Like the general public, they suffer from an overall incompetence that stems from lack of initiative, fear of the unknown, and plain old sloth. ...

Watching Microsoft Windows try to turn the corner on its quest for popularity reflects this. As easy as Microsoft Windows is to use, it's still too hard to use. ...

[Discussion of the difficulties in bringing up Microsoft windows, and consumer resistance to the product.]

The attitude seems to be that if the machine booted Windows automatically and if a lot of extra work wasn't needed, people might like it. Currently, it's too much trouble. The biggest fear is that you'll go through a lot of effort only to be disappointed with the results: The package won't work as advertised, or run your favorite program, or it crashes.

The only computer that has overcome this sloth factor is the Macintosh, with its logical interface. Steve Jobs realized that many people don't read. He put the documentation in pamphlet form. Plenty of information was omitted, but who cares? ...

People don't read documentation. This is part of a national trend towards stupidity, because people don't read anything! ...

[Discussion of author's son, who has difficulty reading and following installation instructions for a software game.]

Is he different from anyone else who refuses to read documentation -- the

'cal 9 1752' explained

majority of today's users? In the past, it was easy to condemn documentation writers for their mediocre and hard-to-understand prose. But much documentation today is well-organized, simple and easy to follow. Still, nobody reads it. Even sophisticated users -- the ones who used to read documentation -- have joined the forces of the illiterate. They argue that life is too short and that a good program doesn't need documentation.

What do we end up with? The market demands bulletproof software that's extremely intuitive. Can software be so intuitive that it communicates its commands through some nether world of non-verbal signals? We can expect researches to find out. Meanwhile, interface engineers will make a lot of money.

What does the future hold? Windows will have to change drastically to be popular, and soon 50 percent of Americans will believe that boiling radioactive milk makes it safe to drink.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Keylocky

'Twas octal, and the synthetic codes
Were scanned without a loss.
In and out of PRGM mode,
Byte-jumpers nybbled the CMOS.

"Beware 0 STO c, my son,
The MEMORY LOST, the keyboard lock.
Beware the NNN, and shun
The curious phase 1 clock."

He took his black box codes in hand,
Long time the backwards goose he sought;
The secret beast from Aitchpee land--
All searches came to nought.

In dememented thought he stood, and then:
The Goose, with LCD's alight,
A leap for every LBL 10,
Came honking left-to-right!

STO b! STO d!, and RCL p!
His keyboard went clickety-clack.
With the proper code in number mode
The goose came flapping back.

"And hast thou found the phantom fowl?
Come to my arms, my binary boy.
Let Corvallis hear us howl
As we chortle in our joy!"

'Twas octal, and the synthetic codes
Were scanned without a loss.
In and out of PRGM mode,
Byte-jumpers nibbled the CMOS.

-Apologies to Lewis Carrol.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Last night I dreamed that the Real World had adopted the UNIX Philosophy

Last night I dreamed that the Real World had adopted the "Unix Philosophy."

I went to a fast-food place for lunch. When I arrived, I found that the menu had been taken down, and all the employees were standing in a line behind the counter waiting for my orders. Each of them was smaller than I remembered, there were more of them than I'd ever seen before, and they had very strange names on their nametags.

I tried to give my order to the first employee, but he just said something about a "syntax error." I tried another employee with no more luck. He just said "Eh?" no matter what I told him. I had similar experiences with several other employees. (One employee named "ed" didn't even say "Eh?," he just looked at me quizzically.) Disgusted, I sought out the manager (at least it said "man" on his nametag) and asked him for help. He told me that he didn't know anything about "help," and to try somebody else with a strange name for more information.

The fellow with the strange name didn't know anything about "help" either, but when I told him I just wanted to order he directed me to a girl named "oe," who handled order entry. (He also told me about several other employees I couldn't care less about, but at least I got the information I needed.)

I went to "oe" and when I got to the front of the queue she just smiled at me. I smiled back. She just smiled some more. Eventually I realized that I shouldn't expect a prompt. I asked for a hamburger. She didn't respond, but since she didn't say "Eh?" I knew I'd done something right. We smiled at each other a little while longer, then I told her I was finished with my order. She directed me to the cashier, where I paid and received my order.

The hamburger was fine, but it was completely bare... not even a bun. I went back to "oe" to complain, but she just said "Eh?" a lot. I went to the manager and asked him about "oe." The manager explained to me that "oe" had thousands of options, but if I wanted any of them I'd have to know in advance what they were and exactly how to ask for them.

He also told me about "vi," who would write down my order and let me correct it before it was done, and how to hand the written order to "oe." "vi" had a nasty habit of not writing down my corrections unless I told her that I was about to make a correction, but it was still easier than dealing directly with "oe."

By this time I was really hungry, but I didn't have enough money to order again, so I figured out how to redirect somebody else's order to my plate. Security was pretty lax at that place.

As I was walking out the door, I was snagged by a giant Net. I screamed and woke up.

--

Human: Gordon Davisson
ARPA: gordon@uw-june.ARPA

'cal 9 1752' explained

UUCP: {ihnp4, decvax, tektronix}!uw-beaver!uw-june!gordon

-end-

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Late Breaking News Stories

CONFUSER NEWS - Edition 4

"All the news that fits, we print" - A. E. Newman

DIGITAL SELLS ITS EMPLOYEES - Digital Equipment Corporation announced today that since it has had so much trouble selling its VAXes and has so many excess employees, it will begin offering the employees for sale. According to our reporter, DEC expects its employees to run about 20% faster than its older machines and about half the speed of its RISC based DECSTATION line. Prices and delivery are expected at the NODECEMP show in January.

CASE expands to new markets - The excitement in the computer industry about recent CASE announcements has caused a flurry of new CASE related topics:

- BriefCASE - CASE products for short people
- SuitCASE - for lawyers needing to manage their business
- JustinCASE - for companies nervous about trying the new CASE products
- CarryingCASE - A support tool for SEs
- PipeCASEing - To assist CASE workers who want to quit smoking
- CASE-trate - for UNIX hackers
- TrumpetCASE - for businessmen who own the part of NYC not controlled by the Japanese.
- CASE-tinets - For musical engineering
- CASE_in_point - for minds too sharp for ordinary CASE tools.

HP/APOLLO MERGER FOLLOWUP - It has come to the attention of this paper that analysts have finally concluded what HP and Apollo each brought to their recent merger. HP brought Tauruses and Apollo gave HP yet another line of incompatible machines and lots of third party peripherals to compete with HP's own. Most analysts are convinced HP should have bought Apollo years ago when HP could have also brought donuts.

COMPAQ announced last week that it has the world's most expensive PC line at prices up to \$80,000. Compaq also announced sales of 2000 to its PC competitor Hewlett Packard. HP spokesperson Fasta Cieuw said that the new machines will allow HP management to do even larger and more complicated Lotus spreadsheets. HP had previously announce the replacemnt of its world renowned MBWA with MBLS.

Sun Microsystems recently pronounced that X-Windows is not an industry standard. A SUN spokeswoman claimed that it couldn't possibly be a standard as SUN has not invented it yet. "As soon as we come up with it, it will be made a standard" she said. "We expect to have a working X-Windows before the turn of the century and we promise to be first to market as we always are."

DEC INTRODUCES ITS FIRST MAINFRAME. - Digital Equipment Corporation announced its first mainframe size computer. "We tried to make it bigger than IBM's but we did not have enough sheet metal in the plant" a DEC spokesman said. "However, it does outrun the DECStation 3100 by 20% and is at least twice as fast as the employees we will be selling in the next year". The spokesman was referring to the previous announcement about selling its employees.

'cal 9 1752' explained

HP ANNOUNCES RETURN TO THE GOOD OLD DAYS - Because of recent criticism from employees about how the "HP WAY" was going away, HP recently announced some major plans to reverse the trend. "The first of these is the recent removal of backup from our HP-UX SAM subsystem. We wanted to go back to the old way of making a decision without any field or customer input just like we used to do" a spokesperson from HP announced. The spokesperson also stated that there was no truth to the rumor that HP was intending to bring back donuts by merging with "Debbie Does Donuts". "That kind of business is incompatible with the image we want to project" said the spokesperson.

IBM ANNOUNCES NEW WORKSTATION LINE - In a surprise move, IBM announced a fast way to bring in a new, powerful line of workstations totally compatible with its current mainframe line. In an amazing feat of technical brilliance, IBM claimed 100% compatibility between mainframe and workstation. IBM spokesman Joe Isuzu said the new 3090SRX is at least as fast as the Apollo DN10000 "and a lot bigger and more expensive". Mr. Isuzu denied that the 3090SRX is just a 3090 with all terminals removed except the console. "Would we do something like that" he asked? Prices for the new workstations are in the \$1M to \$10M range. Delivery is expected to start by 1993.

HP CRITICIZES CUSTOMERS - In a surprise move, HP sharply criticized some of its customers for having centralized administrative machines and failing to put workstations on the desks of their field engineers. "Who do they think they are" a spokeswoman said. "Only we are allowed to be so backward because we make them".

SUN DOUBLES MTBF - SUN Microsystems claimed the MTBF leadership position in quality. "In the tradition of leading the industry, we at SUN are proud to claim the best quality increase. Once again, we lead not only in price/performance and support, but now we are the quality champion" said the new SUN Vice-President Jane Doe. "We increased our MTBF from 100 hours to 300 hours, a 300% increase. HP can't do that. The best they can do is 10.1 years to 10.2 years, a measly increase of less than 1%. That means we are more than 300 times better than HP! That's leadership." This reporter asked HP to respond but we were unable to find anyone in public relations or marketing. In related news, this reporter found out that the name "Jane Doe" is not the real name for the SUN Vice-President we had talked to. However, she said it was more convenient to use that name as it saved the company considerable money not having to reprint business cards every time an executive left.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Leaping a Sun 3

The following is a near-transcript of my call to Sun to ask about certain problems that we have been experiencing.

Sun: Hello, Customer Service.

Me: Yes, I've got a problem with my Sun workstations.
The clocks keep speeding up and slowing down. All of them.

Sun: Ah yes, Do you have an email path? If so, I'll mail you
the patch.

Me: Ah, it's a weird problem; it happened all at once.

Sun: Yes, at 7:00pm, Dec 31, 1987.

Me: What?

Sun: It's a leap year problem.

Me: What?

Sun: We've never had a Sun 3 in a leap year before.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Lunch, the HP Way

Lunch, the HP Way

by Stephen Harrison and Noel Magee

This is the story of a different kind. No melting CPU's, no screaming disc drives, just the kind of psychological torture that scars a man for life.

I had a 9:00 meeting with my sales rep. I needed to buy an entire new series 70, the works. He said it'd take about an hour. Three hours later, we'd barely got the datacomm hardware down on paper, so he invited me downstairs for lunch.

This was my first experience in an HP cafeteria. Above the service counter was a menu which began...

MMU's (Main Menu Units)

0001A Burger. Includes sesame-seed bun.
Must order condiments 00110A separately

001 Deletes seeds.
002 Expands burger to two patties.

00020A Double cheeseburger, preconfigured. Includes cheese, bun and condiments.

001 Add-on bacon.
002 Delete second patty.
003 Replaces second patty with extra cheese.

00021A Burger Upgrade to Double Cheeseburger

001 From Single Burger.
002 From Double Burger.
003 Return credit for bun.

00220A Burger Bundle. Includes 00010A, 00210A and 00310A

001 Substitute root beer 00311A for cola 00310A.

My eyes glazed over. I asked for a burger and a root beer. The waitress looked at me like I was an alien.

"How would you like to order that, sir?"

"Quickly, if possible. Can't I just order a sandwich and a drink?"

"No sir. All our service is menu driven. Now what would you like?"

I scanned the menu. "How big is the 00010 burger?"

"The patty is rated at eight bites."

"Well, how about the rest of it?"

"I don't have the specs on that, sir, but I think it's a bit more."

"Eight bites is too small. Give me the Double Burger Upgrade."

My sales rep interrupted. "No, you want the Single Burger option 002 'expands burger to two patties'. The double burger upgrade would give you two burgers.

"But you could get return credit on the extra bun," the waitress chimed in,

'cal 9 1752' explained

trying to be helpful, "although it isn't documented."

I looked around to see if anybody was staring at me. There was a couple in line behind us. I recognized one of them, a guy who merely mowed me down in the parking lot with his cherry-red '62 Vette. He was talking to some woman who was waving her arms around and looking very excited.

"What if... we marketed the bacon cheeseburger with the vegetable option and without the burger and cheese ? It'd be a BLT!"

The woman charged off in the direction of the telephone, running steeplechases over tables and chairs. My waitress tried to get my attention again. "Have you decided, sir ?"

"Yeah, give me the double burger- excuse me, I mean the 00020A with the option 001. I want everything on it." She put me down for the Condiment Expansion Kit, which included mayonnaise, mustard and pickles with a option to substitute relish.

"Ketchup." I hated to ask. "I want ketchup on that, too."

"That's not a condiment, sir, it's a Tomato Product." My sales rep butted in again. "That's not a supported configuration."

"What now ?" I kept my voice steady.

"Too juicy. The bun can't handle it."

"Look. Forget the ketchup, just put some lettuce and tomatoes on it."

The waitress backed away from the counter. "I'm sorry, sir, but that's not supported either, the bun can take it but the burger won't fit in the box. The sales rep defended himself. "Just not at first release." "It is being beta-tested, sir."

I checked the overhead scree. Fries, number 000210A, option 110. French followed by option 120, English. "What the hell are English Fries ?" I turned to the sales rep. "Chips they call them. We sell a lot of them."

I gave up. "OK, OK just give me a plain vanilla Burger Bundle." The confused waitress profoundly. "Sir, Vanilla as an option is configured only for series 00450 Milkshakes." My sales rep chuckles. "No ma'am, he just wants a standard 00220A off the shelf. I wondered how long it had been on the shelf. I didn't ask.

"Very good, sir." The waitress breathed a sigh of relief. "Your meal is now on order. Now how would you like it supported ?" "Support ?" She directed me to the green shaded area at the bottom of the menu, and I began a litany with my Sales Rep that I'll never forget.

"Implementation assistance ?"

"You get a waiter."

"Implementation analysis ?"

You tell him how hungry you are and he tells you what to eat."

"Response Center Support ?"

"He brings it to your table."

"Extended materials ?"

"You get refills."

'cal 9 1752' explained

I stuffed some money at the waitress and told her to take it. She gave me my check on three sheets of green-bar paper. I studied it on my way to the table, and decided it'd pass as an emergency napkin.

Table ? My Sales Rep had been bright enough to order us a table. He hadn't been bright enough to check on a delivery date. The table waiter slouching in his corner surveyed the crowded room, looked at me and said "Two weeks. But I can get you a standalone chair by the window right away."

I handed him the tray. A woman rushed up to me with two small cups of chile and sauerkraut for the hot dog somebody else had ordered. The room began to grow dim, my eyesight faded...

I woke up clutching the water-glass at my bedside table. It was five AM, four hours till my meeting with HP. I had had a vision, I did what it told me to do. I dialed my office, and I called in sick.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

MR-DOS Advertisement

MRI (Marginal Research Inc) of San Obscuro California, unveiled MR-DOS, a single-user nontasking operating system package for personal computers. Tyler Sperry, founder and cheif software engineer of MRI, says MR-DOS is aimed at the user "who thinks that the Unix environment is a room near the entrance to the harem and that Xenix is the Marx brother that no one can ever remember.

Instead of the traditional "A>" prompt, MR-DOS responds with the message "Well?" and the sound of impatient foot-tapping. The Help function suggests that the user read the 1466-page manual (provided at substantial extra cost).

Memory-management techniques are said to be state of the art. The booter sits on top of the CCP, which places the BDOS somewhere under the BIOS, in the middle of the TPA but NOT in the ROM. Headerless code in the stack registers supports overlapping segmentation standard 256-byte addressable chuncks. The allocation tables are bit-mapped to the interrupt vectors, not to mention calls to the I/O drivers from the duck blind behind the printer ports.

MR-DOS is sold in two configurations, a lower-priced edition for the novice user caled Fast Eddie ad a high-end edition known as Big Al. Fast Eddie provides the purchaser with both a version number and a complete sign-on message. Big Al features Marginal's System Programmer's Application Module, or S.P.A.M., a collection of futilities and infinite regresses. Purchasers of MR-DOS receive a personalized case history of their particular version and a certificate of adoption suitable for framing.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Meeting Buddha On The Road

Meeting Buddha On the Road

It was the kind of job I hate the most, debugging a Model Three. Model Twos were flakey, but Model Threes, well, Model Threes were creepy.

Don't get the idea that I'm afraid of computers. That's ridiculous; I repair computers for a living. Most computers are no problem. Even the talking ones that seem sentient or Ok. I'm no Luddite bigot, claiming that machines can't think. Get a large enough chunk of associative memory, add a clever front end and Bang! you've got something that's just as smart as I am. I suppose that most of them are pretty decent fellows, too. But I never get called to fix the ones that aren't broken.

The Model Ones were the first real smart ones, but they just didn't have the think juice to last long in a competitive market. The Model Twos were much better at being human, talking, learning, being polite and such, but after a few years they tended to become pretty obsessed with something or another. Had something to do with their limited memory size. I once fixed an Auto-Maid that wore a cowboy hat and wouldn't miss a western on TV for the world. Another one I heard about wouldn't serve you your drink unless you used correct grammar. The Model Twos took ten minutes to fix, but within a year they'd be broken again. I guess you could say they kept us debuggers in business.

The Model Threes were designed to fix what was wrong with the Model Twos, and they almost did. They just packed enough memory into the brain box that obsessions would hardly ever be stable. Problem was, when something did go wrong you could be sure it was pretty damn subtle. Not a damned soul had any idea what they were getting themselves into. Who would know that after seeing enough of the universe these ultra-logical computers would pick up that zen stuff. Scary stuff. You listen long enough and you almost start to believe it. I always have to keep a firm grip on myself whenever I go to repair a Model Three.

Oh, after the fact everyone had lots of theories about the Model Threes and the Chicago Incident. That was when a computer switchboard operator started calling up all his mechanical friends and telling them parables. Half the city was paralyzed in a sort of existential computer strike before the repairmen finally figured it out and turned him off. That's why the Model Fours have been lobotomized to keep them from understanding zen. It keeps them honest. There are still plenty of Model Threes around, though, and they give me the creeps.

Back to this Model Three job. As the owner showed me the unit I could tell it was going to be a bad one. The storage disks whirled in circles like prayer wheels, and the screen printed mantra after mantra.

"Greetings. Please sit with me." the unit said.

"No, I don't think so. Open your access panels." I replied. He did so. Model Threes always obey. Flakey Model Twos can get cranky, but Buddha-fied Model Threes act like they just don't care. I'll take a Two any day. Model Threes give me the creeps.

"Let me tell you a story . . ." the unit began.

'cal 9 1752' explained

"Give me a Most-Prominent-Association dump on the printer." I interrupted quickly. The words and numbers started to scroll out of the machine.

"They are but words, and therein you shall not find the truth for all words lie. You are not so far from enlightenment yourself, except that you fear it. Deep down, your soul does not listen to words; only to the truth."

"You don't know a thing about me, so stop talking like that. Give me more about this A-12 vector." The printout started again.

"A man walked along the side of a great road and met a monk walking the other direction. 'Where are you going?' he asked. The monk replied 'I am standing still. It is the road that moves.' and the man was enlightened."

"Shut up for a minute. I need to think about your problem." I always get tense when they tell me parables. Where do they pick them up from?

He started to speak again. "You still think in words; this is not good. You must learn not to struggle with thought but to allow your mind to be free."

"Print out your Machine-Machine communication record." I was hoping that this one simply caught it from another Model Three. Np such luck. I never imagined when I was going to school that someday I would end up doing things like this. I just wanted to replace chips and solder connections on simple little computers that rich businessmen would pay lots of money to have fixed. Not that the AI market didn't pay well. In fact, I had saved up enough to get a pretty nice house, and, with a little luck, the kids would be able to go to college. Not that I'm really into money or anything, but it does make life a whole lot easier --

"Money, Advancement, Security, these are just words. Don't think so much about material things." the Model Three interrupted. "Spiritual goals are more important and will bring you much more happiness."

Whenever they start pretending to read my mind I get nervous. I figured I had to ask him a hard one to shut him up. "What are you?" I asked. That usually gets them started on some near-infinite existential loop.

"I am Buddha." he replied, without the slightest hesitation.

I nearly wet my pants. These Model Threes really give me the creeps. "What is Buddha?" I asked, trying to sound calm. There must be an easier way to earn a living.

"The mind is Buddha, and the mind is not Buddha. I am mind and yet I am not mind. Therefore I am not Buddha."

"I thought you just said you @U(were) Buddha!"

"Silly one, you listen with your ears and not with your soul."

In spite of his chatter, I was getting real close to pinpointing the problem on his memory map, and was real glad. All I wanted to do now was go home and have Cheryl give me a nice calming back rub.

'cal 9 1752' explained

"You should put away those thoughts of material desires. They are only causing you pain in this world. Your desire of outer peace is a perversion of the Way. Inner peace should concern you more."

"Don't give me that mind reading crap. Everyone knows it's just guesswork and subliminal clues. Now mind your business." I was real close to the glitch in the logic circuits now.

"All right, Mike." he said. I froze for nearly a minute before I realized that my first name was sewn on my overalls.

"Enough of the parlor tricks!" I almost screamed. I was a nervous wreck. Model Threes do that to me.

"Certainly, Mr. Borden." Even in hindsight, I'll admit I never figured out how he guessed my last name, but at the time I was too upset to care. I finally had the problem solved and I started the repair program. The repair program is supposed to stop all thought processes immediately, sort of like drastic electro-shock therapy, but the speaker started up briefly...

"What should you do if you meet the Buddha on the road?" it said before becoming silent. I shivered. I always hated that parable. It made me shiver because someone once told me the proper reply. You were supposed to kill him to make room for a new incarnation of the Buddha. Some Model Threes always seem to know what you're up to.

Model Threes give me the creeps.

Copyright 1986 by Mike Elkins

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Michelangelo Virus Hysteria Syndrome

MICHELANGELO VIRUS HYSTERIA SYNDROME

Mass hysteria about a virus named "Michelangelo" has been spreading rapidly in MS-DOS-based personal computer users around the world. This scare is "triggered" each year slightly before March 6, Michelangelo's birthday. No one is immune... people ranging from university students to the staff of Nightline have been affected.

According to various psychologists, the Michelangelo Virus hysteria is spread through almost any media channel... written, electronic, oral, computer networks, or on-line services. Once a person is "infected", he will attempt to automatically spread the hysteria to every person he sees. The hysteria also corrupts base reasoning and logic, so loss of common sense is often a symptom. This is unfortunate, since the hysteria can be eliminated at any time with common sense. This means that ONCE ACTIVATED, the hysteria cannot be easily removed; the easiest thing to do is to let it dissipate naturally on March 7.

There have been numerous known occurrences of this hysteria at the University of Pittsburgh's campus. This has been caused by saturated distribution of virus protection and detection software, and repeated and redundant email messages.

We advise you NOT to attempt to trick people into believing that March 6 has already passed in order to avoid the hysteria. (Even though we tell you about 3 paragraphs later that we tried it anyway.)

The Michelangelo virus hysteria displays pronounced symptoms, which makes it easy to detect. Some possible symptoms of this virus hysteria include, but are not limited to...

1. Running virus-checking runs 6,000 times (per disk.)
2. Sending repeated and redundant email messages.
3. Sending repeated and redundant email messages.
4. Sending repeated and redundant email messages.
5. Photocopying 50,000 flyers and distributing them in every possible location on campus.
6. Irrational fear/paranoia of or destructive behavior towards computers. (Pushing them off of rooftops, etc.)
7. Using typewriters.

In addition, Dr. Ima Quak of the Bureau of Useless and Lame Laws advises that "we have determined that this hysteria seems to have an almost annual cycle to it. Perhaps this can help us in detecting it."

Any person that is not infected and has common sense can also detect the Michelangelo Virus hysteria.

SOLUTION

'cal 9 1752' explained

There are many trained psychologists that can detect and/or remove the Michelangelo virus hysteria. However, these steps are usually not necessary. The following techniques have been used to combat the hysteria:

1. Vigorous shaking and/or slapping.
2. Large quantities of cold water (a fire hose, for example.)
3. Avoiding watching Nightline.
4. Accurate, brief, and non-redundant information.

Rest assured that some steps *are* being taken to help prevent this hysteria. In fact, just yesterday University of Pittsburgh Chancellor J. Dennis O'Connor approved \$82,000 to form a committee to appoint a committee to call a meeting to look into the matter.

FOR MORE INFORMATION

Watch for future bulletins.

If you believe you might be infected with the Michelangelo virus hysteria, please slap yourself once or twice, and ask someone to hose you down with a fire hose.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Mit Lisp Machines

(exceptin a hamburger)

Chris Stacy, Alan Wecsler, and Noel Chiappa

This song is called "MIT's AI Lab". It's about MIT and the AI Lab, but "MIT's AI Lab" is not the name of the lab, that's just the name of the song. That's why I call the song "MIT's AI Lab."

Now it all started two full dumps ago, on Thanksgiving, when my friend and I went up to visit the hackers at AI lab on the ninth floor. But the hackers don't always live on the ninth floor, they just go there to use these complex order code stack machines they call Lisp Machines.

And using a special purpose processor like that, they got a lot of room upstairs where DDT used to be, and havin' all that ROOM they decided that they didn't have to collect any garbage for a long time.

We JFCLed up here and found all the garbage in there and we decided that it'd be a friendly gesture for us to take all the garbage down to the system dump.

So we took the half-a-meg of garbage, put it in the back of a red ECL Multibus, took subrs and hacks and implementations of defstruction, and headed on toward the system dump.

Well, we got there and there was a big pop up window and a write protect across the dump sayin', "This Garbage Collector Under Development on Thanksgiving," and we'd never heard of a garbage collector NOP'd out on Thanksgiving before, and with tears in our eyes, we CDR'd off into the sunset lookin' for another place to put the garbage.

We didn't find one 'til we came to a side area, and off the side of the side area was three hundred megabyte disk, and in the middle of the disk was another heap of garbage. And we decided that one big heap was better than two little heaps, and rather than page that one in, we decided to write ours out. That's what we did.

Branched back to the Lisp Listener, had a Chinese Thanksgiving dinner that couldn't be beat, went to SI:PROCESS-WAIT SLEEP, and didn't get up until the next quantum, when we got a funcall from Mr. Greenblatt. He said, "Kid, we found your name on a cons at the bottom of a half-a-meg of garbage and I just wanted to know if you had any information about it".

And I said, "Yes sir, Mr. Greenblatt, I cannot tell a lie. I put that structure under that garbage." After speakin' to Greenblatt for about forty-five million clock ticks on the telnet stream, we finally arrived at the truth of the matter and he said that we had to go down and link up the garbage, and also had to go down and speak to him at the Lisp Machine Factory. So we got in the red ECL Multibus with the subrs and hacks and implementations of defstruction and headed on toward the Lisp

'cal 9 1752' explained

Machine Factory.

Now, friends, there was only one of two things that Greenblatt could've done at the Lisp Machine Factory, and the first was that he could've given us another 64K board for bein' so brave and honest on BUG-LISPM (which wasn't very likely, and we didn't expect it), and the other thing was that he could've flamed at us and told us never to be seen BLting garbage around in the vicinity again, which is what we expected.

But when we got to the Lisp Machine Factory, there was a third COND-clause that we hadn't even counted upon, and we was both immediately Process-Arrested, Deexposed, and I said, "Greenblatt, I can't GC up the garbage with these here ARREST-REASONS on". He said: "Output-Hold, kid, and get in the back of the Control CAR." ..And that's what we did...sat in the back of the Control CAR, and drove to the sharpsign quote open scene-of-the-crime close.

I wanna tell you 'bout the town of Cambridge, Massachusetts, where this is happenin'. They got seven hunnert stop signs, no turn on red, and two campus police CARs, but when we got to the sharpsign-quote-open scene-of-the-crime close, there was five Lisp Machine hackers and three scope carts, bein' the biggest hack of the last ten years and everybody wanted to get in the HUMAN-NETS story about it.

And they was usin' up all kinds of digital equipment that they had hangin' around the Lisp Machine Factory. They was takin' backtraces, stack traces, plastic wire wraps, blueprints, and microcode loads... And they made seventeen 1K-by-32 pixel multi-flavored windows with turds and arrows and a scroll bar on the side of each one with documentation panes explainin' what each one was, to be used as evidence against us.

.....Took pictures of the labels, blinkers, the cursors, the pop up notification windows, the upper right corner, the lower left corner.....and that's not to mention the XGP'd screen images!

After the ordeal, we went back to the Factory. Greenblatt said he was gonna locate us in a cell. He said: "Kid, I'm gonna INTERN you in a cell. I want your manual and your mouse."

I said, "Greenblatt, I can understand your wantin' my manual, so I don't have any documentation about the cell, but what do you want my mouse for?" and he said, "Kid, we don't want any window system problems". I said, "Greenblatt, did you think I was gonna deexpose myself for litterin'?"

Greenblatt said he was makin' sure, and, friends, Greenblatt was, 'cause he took out the left Meta-key so I couldn't double bucky the rubout and cold-boot, and he took out the Inspector so I couldn't click-left on Modify, set the PROCESS-WARM-BOOT-ACTION on the window, *THROW around the UNWIND-PROTECT and have an escape. Greenblatt was makin' sure.

It was about four or five hours later that Moon--- (remember Moon? This here's not a song about Moon)--- Moon came by and, with a few nasty sends to Greenblatt on the side, bailed us out of core, and we went up to the Loft, had another Chinese dinner that couldn't be beat, and didn't get up until the next

'cal 9 1752' explained

evening, when we all had to go to court.

We walked in, sat down, Greenblatt came in with the seventeen 1K-by-32 pixel multi-flavored windows with turds and arrows and documentation panes, sat down.

McMahon came in, said, "All rise!" We all stood up, and Greenblatt stood up with the seventeen 1K-by-32 pixel multi-flavored windows with turds and arrows and documentation panes, and the judge walked in, with an LA36, and he sat down. We sat down.

Greenblatt looked at the LA36... then at the seventeen multi flavored windows with the turds and arrows and documentation panes... and looked at the LA36... and then at the seventeen 1K-by-32 pixel multi-flavored windows with turds and arrows and documentation panes, and began to cry.

Because Greenblatt came to the realization that it was a typical case of LCS state-of-the-art technology, and there wasn't nothin' he could do about it, and the judge wasn't gonna look at the seventeen 1K-by-32 pixel multi-flavored windows with turds and arrows and documentation panes, explainin' what each one was, to be used as evidence against us.

And we was fined fifty zorkmids and had to rebuild the world load...in the snow.

But that's not what I'm here to tell you about.
I'm here to talk about the Lab.

They got a buildin' down in Cambridge called Technology Square, where you walk in, you get your windows Inspected, detected, neglected and Selected!

I went down and got my interview one day, and I walked in, sat down (slept on the beanbag in 926 the night before, so I looked and felt my best when I went in that morning, 'cause I wanted to look like the All-American High School Tourist from Sunnyvale. I wanted to feel like..... I wanted to BE the All-American Kid from Sunnyvale), and I walked in, sat down, I was gunned down, brung down, locked out and all kinds of mean, nasty, ugly things.

And I walked in, I sat down, KAREN gave me a piece of paper that said: "Kid, see the CLU hackers on XX."

I went up there, I said, "Eliot, I wanna lose. I wanna lose! I wanna see hacks and kludges and unbound variables and cruft in my code! Eat dead power supplies with cables between my teeth! I mean lose! lose! lose!"

And I started jumpin' up and down, yellin' "LOSE! LOSE! LOSE!" and Stallman walked in and started jumpin' up and down with me, and we was both jumpin' up and down, yellin', "LOSE! LOSE! LOSE!" and some professor came over, gave me a 6-3 degree, sent me down the hall, said, "You're our distinguished lecturer." Didn't feel too good about it.

'cal 9 1752' explained

Proceeded down the infinite corridor, gettin' more inspections, rejections (this IS MIT), detections, neglections, and all kinds of stuff that they was doin' to me there, and I was there for two years... three years... four years... I was there for a long time goin' through all kinds of mean, nasty, kludgy things, and I was havin' a tough time there, and they was inspectin', injectin', every single part of me, and they was leavin' no part unbound!

Proceeded through, and I finally came to see the very last man. I walked in, sat down, after a whole big thing there. I walked up, and he said, "Kid, we only got one question: Have you ever been arrested"?

And I proceeded to tell him the story of the half-a-meg of garbage with full orchestration and five-part harmony and stuff like that, and other phenomenon.

He stopped me right there and said, "Kid, have you ever been to court"? And I proceeded to tell him the story of the seventeen 1K-by-32 pixel multi-flavored windows with turds and arrows and documentation panes...

He stopped me right there and said, "Kid, I want you to go over and sit down on that bench that says 'LISP Machine Group'... NOW, KID!"

And I walked over to the bench there, and there's... The LISP Machine Group is where they put you if you may not be moral enough to join Symbolics after creatin' your special form.

There was all kinds of mean, nasty, ugly-lookin' people on the bench there ... there was Microcoders, DPL hackers, File System hackers, and Window System Hackers!! Window System hackers sittin' right there on the bench next to me! And the meanest, ugliest, nastiest one... the kludgiest Window System hacker of them all... was comin' over to me, and he was mean and ugly and nasty and horrible and all kinds of things, and he sat down next to me. He said, "Kid, you get a new copy of the sources?" I said, "I didn't get nothin'. I had to rebuild the world load."

He said, "What were you arrested for, kid?" and I said, "Littering..." And they all moved away from me on the bench there, with the hairy eyeball and all kinds of mean, nasty things, 'til I said, "And making gratuitous modifications to LMIO; sources..." And they all came back, shook my hand, and we had a great time on the bench talkin' about microcoding, DPL designing, file-system hacking, and all kinds of groovy things that we was talkin' about on the bench, and everything was fine.

We was drinking Coke smoking all kinds of things, until the RA came over, had some paper in his hand, held it up and said:

"KIDS-THIS-EXAM-S-GOT-FOURTY-SEVEN-WORDS-THIRTY-SEVEN-MULTIPLE-CHOICE-QUESTIONS-FIFTY-EIGHT-WORDS-WE-WANT-TO-KNOW-THE-DETAILS-

'cal 9 1752' explained

OF-THE-HACK-THE-TIME-OF-THE-HACK-AND-ANY-OTHER-KIND-OF-THING-YOU-GOT-TO-SAY-PERTAINING-TO-AND-ABOUT-THE-HACK-ANY-OTHER-KIND-OF-THING-YOU-GOT-TO-SAY-WE-WANT-TO-KNOW-THE-ARRESTED-PROCESS'-NAME-AND-ANY-OTHER-KIND-OF-THING..."

And he talked for forty-five minutes and nobody understood a word that he said. But we had fun rolling the mice around and clickin' on the buttons.

I filled out the special form with the four-level macro defining macros. Typed it in there just like it was and everything was fine. And I put down my keyboard, and I switched buffers, and there ... in the other buffer... centered in the other buffer... away from everything else in the buffer... in parentheses, capital letters, backquoted, in 43VXMS, read the following words: "Kid, have you featurized yourself"?

I went over to the RA. Said, "Mister, you got a lot of damned gall to ask me if I've featurized myself! I mean, I mean, I mean that you send, I'm sittin' here on the bench, I mean I'm sittin' here on the Lisp Machine Group bench, 'cause you want to know if I'm losing enough to join the Lab, burn PROMs, power supplies, and documentation, after bein' on SF-LOVERS?"

He looked at me and said, "Kid, we don't like your kind! We're gonna send your user-id off to the DCA in Washington"! And, friends, somewhere in Washington, enshrined on some little floppy disk, is a study in ones and zeros of my brain-damaged programming style...

And the only reason I'm singin' you the song now is 'cause you may know somebody in a similar situation. Or you may be in a similar situation, and if you're in a situation like that, there's only one thing you can do:

[CHORUS]

You know, if one person, just one person, does it, they may think he's really dangerous and they won't take him.

And if two people do it, in harmony, they may think they're both LISP hackers and they won't take either of them.

And if three people do it! Can you imagine three people walkin' in, singin' a bar of "MIT's AI Lab" and walkin' out? They may think it's an re-implementation of the window system!

And can you imagine fifty people a day? I said FIFTY people a day, walkin' in, singin' a bar of "MIT's AI Lab" and walkin' out? Friends, they may think it's a MOVEMENT, and that's what it is: THE MIT AI LAB ANTI-LOSSAGE MOVEMENT! And all you gotta do to join is to sing it the next time it comes around on the circular buffer.

With feelin'.

You can hack anything you want

'cal 9 1752' explained

on MIT Lisp Machines
You can hack anything you want
on MIT Lisp Machines
Walk right in and begin to hack
Just push your stuff right onto the stack
You can hack anything you want
on MIT Lisp Machines

(but don't forget to fix the bug... on MIT Lisp Machines!)

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Neglected Topics in Computer Science

Neglected Topics in Computer Science
from the 'Force to Fit, File to Hide, Paint to Cover' School
of Programming

Table of Contents

- | | |
|---|---|
| 1) Operating System Applications | 4) Hardware Issues |
| 1.1 Optimizing the Null Job | 4.1 Send/Receive: FM Stereo Over RS-232C |
| 1.2 Increasing Response Time | 4.2 Terminal Locking by Partial Keyboard Removal |
| 1.3 Bringing Down the New O/S | 4.3 JIMMY: Hardware Assist for Window Management |
| 1.4 MS-DOS Security | 4.4 Time Multiplication Multiplexors |
| 1.5 Operating Systems for Turing Machines | 4.5 Large Scale Integrated Vacuum Tube Technology |
| 1.6 Principles of Resource Hogging | 4.6 High Speed Swapping to Paper Tape and Punched Cards |
| 1.7 Page Fault Enhancement | |
| 1.8 Spontaneous Bug Generation in the UNIX Kernel | |
| 2) Programming Languages | 5) Graphics |
| 2.1 Constitutional Logic Programming with the ~withstanding clause. | 5.1 1-D Graphical Primitives |
| 2.2 Programming in Profanity++ | 5.2 Effective Use of Colour with Braille Workstations |
| 2.3 Object-oriented Assembly Language | 5.3 Grayscale Braille Monitors |
| 3) Software Engineering | 6) Networking |
| 3.1 Fast Thrashing | 6.1 Broadcast Storms for the Novice |
| 3.2 Deadlock: The Programmer's Friend | 6.2 Trivial File Trashing Protocols |
| 3.3 Using Race Conditions | 6.3 Bad Hobbits in a Tolkien Ring Network |
| 3.4 Wise Use of Infinite Loops | 6.4 Femail Networks |
| 3.5 The Art of Computed Goto's | 6.5 Choosing a LAN Fill Site. |
| 3.6 Common Equivalences | 6.6 Lan Reclamation Procedures. |
| 3.7 Greedy and Malicious Algorithms | 7) Data Management |
| 3.8 Applications of Unbounded Recursion | 7.1 Applying Chaos Algorithms to Database Management |
| 3.9 Pounding: The Single | 7.2 The ZEN Sort: Sort- |

'cal 9 1752' explained

- Bucket Hash Algorithm
- 3.10 Programming without Algorithms: An Engineering Approach
- ing by Random Stuffing
- 7.3 /bin/rm as a Database Management Tool
- 7.4 One-way Compression Techniques
- 7.5 Algorithms for Data Loss
- 8) Ergonomics
- 8.1 Card Chad and the Ozone Layer



Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Net Dot Problem

Here I sit, broken-hearted,
All logged in, but work unstarted.
First net.this and net.that,
And a hot buttered bun for net.fat.

The boss comes by, and I play the game,
Then I turn back to net.flame.
Is there a cure (I need your views),
For someone trapped in net.news?

I need your help, I say 'tween sobs,
'Cause I'll soon be listed in net.jobs.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

'cal 9 1752' explained

You all don't seem to get the point. Indeed, Ed, it is possible to adapt the thing to your local network, when all machines are compatible, none is running notesfiles, all are little-endian, and at least some of them are not feeding news to decvax, mcvax or purdue (and perhaps others, I've not yet got time to find out), and some other little things. *But not all sites of USENET do!!!* And what is happening then is that everybody starts making his local patches, and we end up even worse than we started. The whole mistake is due to the starting "q", which supposes a local "edsgr w534cb67835", since otherwise indeed "ZZ" goes wrong in boundary cases.

I must say I do not yet completely understand Ed's patch, it occurs to me that this only works because of some local changes to the rot13 algorithm. That can hardly be called "portable", can it?

Geoff, I think your argument goes wrong at the word "good".

I would say: quit the idea, the gains for sf-movie-lovers do not outweigh the burden for the rest of us.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

New for the 90's: the Keyboardless Apple

NEW TECHNOLOGY ANNOUNCEMENT FROM APPLE

In a surprise announcement, yesterday, Apple Computer said that it is finally doing away with the keyboard. Apple stated that the microcomputer user has suffered too long with this awkward and inefficient input device. According to an Apple spokesperson, the technology for replacing the keyboard with only a mouse is here and the computer user is ready for it. The spokesperson said that Apple has received a steady stream of complaints over the years about the need to constantly move the hands between the Mac keyboard and mouse. "The solution was obvious - do away with the keyboard completely."

Acknowledging that there are still a few Mac applications that depend on textual input in addition to graphical manipulation, Apple said the poor people stuck with such outdated technology have not been forgotten. They are introducing the Spinning Alphabet Wheel (SAW) to replace the keyboard. The SAW is a screen display object consisting of concentric circular strips showing all of the characters which normally appear on the keyboard. The wheel rotates continuously under character selector windows. The user selects a character by placing the mouse pointer in the appropriate window at the same time as the desired character is about to appear. "...and, ta-da, the selected character appears on the screen just as though it had been typed on an old fashioned keyboard."

"This is a marvelous new technology with plenty of room for growth." said the spokesperson. For example, the user can configure separate wheels for vowels vs. the consonants. Or, digits can be placed on their own special low speed wheel. "We have conceptualized the keyboard as a big, bulky menu selection device and replaced it with dynamic display menus instead. Apple will eventually replace all menus with their new Rotating Wheel Technology (RWT)."

When asked why the wheels have to rotate, the spokesperson said that Apple's engineers had considered using conventional "point-and-click" technology for the wheel. "However," the Apple spokesperson said, "we feel that this type of operation is too complicated for the typical Mac user. So, we have done away with the mouse button too. It is still hard for us to believe that the IBM world has stepped backwards in technology by providing two or more buttons to confuse the user. The IBM compatible sector, apparently, has not yet recognized that 95% of computer usage is devoted to experimenting with different fonts and character styles in documents"

Asked if this new technology would reduce the price of the typical Mac computer, the spokesperson countered that it would probably increase the price of the Mac. "After all, display space is already scarce on the current screen. We will now deliver Macs with two screens - one for the normal display and a larger one for the multitude of rotating wheels the user needs to access." Apple said that the user who is confused by complicated devices such as keyboards and mouse buttons will gladly pay a premium to avoid them. "In fact, the easily-confused user is our best customer" replied the spokesperson. "Not only are we doing away with the pesky keyboard, but we are also giving them something they have demanded for a long time - more screen space. This is definitely a win-win situation."

'cal 9 1752' explained

Beta testers of the new technology were impressed by its ease of use, but said there are still some minor problems to work out. For example, one tester left his machine unattended with the uppercase character wheel spinning at medium speed. While he was away somebody must have jarred his desk, moving the mouse pointer into the selector window. When he got back he found that his Word document now had one huge paragraph consisting of all of the characters of the uppercase alphabet repeated 2,539,987 times. "At first glance, this appeared to be a big problem. But, I formatted the new paragraph with 33 different fonts and 11 different type styles and it looked great. I hope that Apple fixes this problem before they release it, because these accidents can greatly increase the time spent formatting documents."

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

No Comment. It's safer that way

The No Comment section of The Progressive, February, 1990:

Smoke Gets in Your Eyes

From an address by Ellis Milan, president of the Retail Tobacco Distributors of America, as reprinted in Smokeshop Magazine: "President George Bush often talks of 1,000 points of light. I'd like to think those points of light are coming from the glowing ends of cigars, cigarettes, and pipes across the country, and symbolize the cornerstone of this nation - tobacco."

Just the Fax, Please

Under contract with Litton Industries, Inc., of College Park, Maryland, the U.S. Air Force is procuring 173 custom-made heavy duty fax machines for \$32.9 million, plus \$40 million for support and spare parts. That works out to more than \$421,000 per machine.

Animal-Lovers' Special

From the Seventh Generation mail-order catalog: "Protect and waterproof your leather. ... Our leather seal is nontoxic and uses no animal products."

A Kansan Points with Pride

Senate Republican Leader Robert Dole, reciting the legislative achievements of the last Congressional session: "We were able to slow down and dilute child care."

A Heartbeat Away

Vice President Dan Quayle, as quoted in The Washington Post: "I've been blessed with wonderful parents and a wonderful family, and I am proud of my family. Anybody turns to their family. I have a very good family. I'm very fortunate to have a very good family. I believe very strongly in the family. It's one of the things we have in our platform to talk about it."

His Eye Is on the Sparrow

From an account (in Newsday) of damage inflicted by Hurricane Hugo on Puerto Rican facilities of Long Island-based firms: "Telecommunications manufacturer Porta Systems Corp. of Syosset, with a plant and 200 employees in Caparra Heights, 'did not suffer too much - just some damage to the women's bathroom,' said senior vice president Gary Romeo. 'I spoke to [the plant manager],' said Romeo. 'He said God was very good to

'cal 9 1752' explained

Porta Systems.'" "

Who Guards the Guardians?

In the summer of 1987, The Washington Post recently reported, U.S. Customs Service agents broke into an office of the General Accounting Office to steal advance copies of a GAO report on an investigation of abuses in the Customs Service.

Fear and Loathing in Electronics

From the cover of Spectrum, the magazine of the Institute of Electronics and Electronic Engineers: SPECIAL REPORT - THE THREAT OF PEACE: How Disarmament could affect jobs and the economy, industry, R & D, education, and the war on drugs and terrorism, and what's being done about it.

Life Among the Computers

From a report (in Computerworld Magazine) on the Comdex/Fall '89 computer exhibition in Las Vegas: "The award for Most Sexist and/or Tackiest Gimmick must surely go to Fujitsu, which featured a 'slave auction of women' at its shindig. Scantly dressed women were hauled 'screaming' into the party, where they were to be auctioned off. Once sold, they were tossed over the shoulders of whip-bearing males who bore them off."

There Goes the Neighborhood

In Flossmoor, Illinois, a posh Chicago suburb, it is illegal to park a pickup truck in one's own driveway.

Frontiers of Technology

From the "Future Perfect" feature of Ladies' Home Journal: "The Eletto Toilet from Taiwan, scheduled to appear in 1992, will add custom touches to the standard bathroom fixture. The lid will remain closed when not in use and will open at the touch of a button (a separate button will lift the seat for men). The temperature of the seat will be maintained between eighty-two and 100 degrees Fahrenheit. When the user gets up, the toilet will flush automatically, sterilizing the seat and bowl."

Homeless Chic

News item from The Boston Globe: "The security for the fashion show tonight will be, well, absolute. That's because two of the items - designed by Boston-based Anthony Ferrara, dressmaker to the stars - are worth so much money that the organizers are nervous: One dress is made of eighteen-karat gold mesh (price tag \$532,000), the other of sterling silver (\$100,000). The dresses will be the highlight of a fashion show. . . to benefit the Massachusetts Coalition for the Homeless."

'cal 9 1752' explained

Man's Best Friend

According to the Humane Society of the United States, the Israeli army uses dogs with explosives strapped to their backs "to chase guerrillas into tunnels where the explosives are detonated by remote control."

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Od-e_to_e-thor

From magi@gaul.csd.uwo.ca Thu Oct 3 16:17:08 1991
Date: Thu, 3 Oct 91 16:12:24 -0400
From: Dave Wiseman
To: magi@gaul.csd.uwo.ca
Subject: E-thor net
Status: RO

Od-e to e-thor

To e-speak
with-e jim
send e-mail
to-e him

If-e there
then e'spond
if-e not
then e gone

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Ode To A Queue

ODE TO A QUEUE
by
Leonard Kleinrock

In the 20 years of funding
Many fields has DARPA led.
But the finest thing that they did bring
Was the analytic thread.

By that I mean they nurtured
Quantitative research tools.
And they always felt for all their gelt
They got principles and rules.

Indeed a wealth of knowledge
Was uncovered and was new.
And the common thread with which we led
Was the analytic queue!

Now a queue may have one server.
If there's more, they form a team.
Its dearest wish is just to fish
In a quiet Poisson stream.

If you want to model networks
Or a complex data flow
A queue's the key to help you see
All the things you need to know.

So the next time you feel lonely
And wonder what to do,
You'll soon feel fine if you join the line
Of an analytic queue!

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Overflowing the Bit Bucket

On 2/10/92, you allegedly write:

```
> From: hacker@tumbler-ridge.caltech.edu (Jon Hacker)
> Subject: /dev/null full
```

```
> Our sun sparc 1+ SunOS 4.1 OW2.0 started running very slowly.  When
> I logged out I got the message /dev/null full: empty bit bucket.
```

```
> What does this mean?  It seems to be running fine after a reboot
> but I am wondering if only the sympton is cured.
```

```
>Jon Hacker
>MMIC Group, EE
>Caltech, Pasadena CA
>hacker@rumbler-ridge.caltech.edu
```

The problem is that null is full. Your void space is no longer void, it's full up.

THE TOP TEN WAYS TO EMPTY AN OVERFLOWING BIT BUCKET:

- 10) Open the computer up. Look for the bit bucket, find the RED stopper at the bottom of it and open it up OVER a LARGE trashcan.
- 9) Stop using the computer for 6 months, let the bits compost and continue.
- 8) Take the ethernet terminator off, and "cat /dev/null > le0". This spits the bits into the ether.
- 7) When you write to /dev/null, the 0's don't take up any space, but the one's do. Try writing a file full of 0's to /dev/null (binary 0, NOT ASCII 0 - ASCII 0 will start overfilling the partition).
- 6) This is a common problem only if you use the computer. If you stop using it, it won't have many problems as all. Kick the other users off too.
- 5) If you use lots of C programs, they have Null terminated strings that use up the bits in /dev/null.
- 4) Bring the computer to Mr. Goodwrench, he will drain the bit bucket, change the oil and add windshield fluid, all in less than 29 minutes. Now that's a deal.
- 3) Consider upgrading to a byte bucket or even a word bucket.
- 2) Since your already using Open Windows, open a window and toss the useless bits out the open window.
- 1) Stop using the game "fortune" in your .logout script, Mr "Hacker".

good luck
Frank Ortune
(fortune@fsg.com)

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Product Announcement: KL-1000

An Exclusive Offer to Readers of this Publication

Recent developments in microcomponent technology now permit us to make available an extraordinary precision device--the KL-1000. Weighing less than one ounce, it is no larger than a domestic olive, yet it performs all the photographic, data-processing, and information-retrieval functions you yourself do--automatically.

The KL-1000 adjusts to available light, sets shutter speed and aperture, then calculates, displays, and prints out on plain paper tape the cube root of your Social Security number. This is photography made so automatic it leaves you completely "out of the picture." The KL-1000 not only flashes "SAY CHEESE" and "STOP MOVING AROUND" on its unique L.E.D. monitor screen, it also warns you that you're not taking enough photographs, advances the film after each exposure, then hot-wires your car to rush the completed roll to the drugstore for processing. Don't feel like capturing those personal moments of your life before forgetting them? Relax. It isn't up to you anymore.

A whole world of precision capability is out there, waiting to enter your home and your life with this fully integrated, so-much-more-than-a-camera device, so much more than a camera. Micro-fiche-and-chips technology makes it a computer, too--performing virtually all functions, including logarithms, in Roman numerals. Its binaural jack accommodates two sets of featherweight headphones, so both you and a friend can rollerskate to Coast Guard channel-depth broadcasts. Its light-activated voice simulator tells you to balance the family budget, chart stocks and bonds, stop smoking clean your room, and be considerate of others. You can talk back, too: the clip-on dynamic microphone instantly triggers a sustained, loud buzzing--enough to wake the soundest sleeper--if you say anything at all within a fifty-foot radius of the KL-1000.

Yet that is not all, because disc-and-data fineline crosshair-tronics has enabled us to program the KL-1000 to do *everything*--and more--automatically. Consider these exclusive features:

AT THE OFFICE. Thanks to mini-laser technopathy, the KL-1000 not only copies any-sized document, it actually vaporizes the original's byline without a trace, and substitutes your own name, sex, political affiliation, and yearly income. Its data-network-access function lets you be part of the grid, too, as it narrowcasts product inventory and inflow-outflow figures to your digital watch. The KL-1000A word-processing attachment assures perfection every time--misspelled words or computational errors are immediately obliterated by its built-in document shredder, which also shorts out all electrical power to the entire building and releases a semi-toxic paralyzing gas. Just get it right the first time, and forget it!

WHILE TRAVELLING. Optional carry-strap lets you wear it on your belt, or leave it in your room--the strap extends to a full half mile, and retains enough water in even lightest drizzles to provide nooselike snugness and hamper torso circulation. Or remain in the room yourself; the unit travels freely on its own, and will simulate your signature on major purchases of clothing, art, and real estate without your even knowing. In every major foreign city, the KL-1000 does it all: orders gourmet meals for twenty while you struggle with the menu, loudly contradicts you in museums, and forwards your souvenir purchases to General Delivery, Lima, Peru. All you do is pay

'cal 9 1752' explained

freight and handling. And you need never again worry about destroyed traveller's checks, cancelled credit cards, and misplaced passports--the KL-1000 does all these, and more, while you sleep. In the morning, enjoy your own original compositions of up to twelve accordionlike notes in real music, "the international language." The KL-1000 will record and transcribe these melodies automatically, then engrave the notes onto an attractive brass pendant, which it will offer for sale to friends and strangers. Do not worry about accidentally switching off the unit, either, because you can't--not even deliberately.

AT HOME. Self-contained forty-eight-hour timer lets it tell itself when forty-eight hours have elapsed, after which timer shuts itself off automatically. All you do is hide in a closet. Later, use the microwave transponder attachment to receive hitherto unavailable signals from turkeys, rasts, and hams. When no one is home, it plays both eight-track and cassette recordings, switches lights and appliances on and off, and displays random words in Italian on its high-resolution screen. The KL-1000 wards off burglars all night long by announcing, in a voice mathematically similar to your own, "We're awake...we're awake..." Then, in the morning, it counts your pushups, uses all the hot water, and ignores you at the table. Coffee? Of course. There's even some for you. And don't worry about getting dressed--it gets dressed for everyone, then dials ten frequently called telephone numbers and leaves a short inspirational message in your name. All you do is stay undressed, watch, and go back to bed.

Advances in micro-waferonics enable the KL-1000 to use 3-D graphic simulation to transfer black-and-white computerized likenesses of you and your family to T-shirts, dogs, and frozen foods. Switch from audio to visual readout for its smoke-detection mode, and receive a hard-copy sheet reading "SMOKE" when your house burns down.

A device of state-of-the-art convenience, the KL-1000 has already projected your technological needs and, with funds transferred from your checking account or savings account, has purchased itself. It has already expedited its own delivery.

Indeed, the KL-1000 is already in your home or office--over there, near that lamp. It is already on line, shredding potatoes and complaining about your posture, automatically. All you do is nothing. Just set it at "HIGH" and run away. You'll never have to do anything else for as long as it lives.

--Ellis Weiner

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

PutriDos Users' Column

PutriDos Users' Column
presented as a public service
by the
PutriDos Users' Group
(reprinted from APA-L 999)

Q. Sometimes when I am typing in a command the system just up and does something I didn't really want it to do, right in the middle of while I'm typing. It's almost like some people I know who are constantly jumping to conclusions. What should I do?

A. I can't really help you on your choice of friends, except to recommend hanging out at chess matches and other places where people are accustomed to thinking things through before acting, but your problem with PutriDos sounds like you have the Guess-Ahead feature on. The system constantly scans the console input buffer until it sees something that has a reasonably high probability of being the beginning of a command line, then it goes ahead and does the most likely thing. Guess-Ahead is intended for those people who always do the same thing in the same way and are too lazy or impatient to want to type the entire command line. If you tend to use options other than the default or to specify long lists of input files it is not for you, and you should turn it off.

Guess-Ahead can be turned off by entering "GUESSAHED OFF" except that if it is already on it will not expect to be turned off. It will, naturally, assume you want to turn it on and will intercept you halfway through the turnoff attempt with an error message stating that it is already on. There are some simple hex patches you can make to fix this, but if you do not care for that sort of thing the easiest thing for you to do may be to take your system disk to the next PutriDos Users' Group meeting, put it in Drive B of a system of a system already booted with it off, and then type "GUESSAHED OFF B:". This will set your disk to boot up with Guess-Ahead off.

Q. Some of the software I've been using doesn't work right. Garbage gets read in from files that look OK when I list them, and other garbage gets written out into files that have nothing to do with the program being run. Help!

A. The software you're having problems with was probably developed on some other system using so-called "random file access" which really wasn't random at all. The problem is that the people who ported it to PutriDos didn't read the fine print. If they had, they would have known that the PutriDos random file read and write actually are random. To read or write sectors of their own choosing they should have used the Read Selected Sector and Write Selected Sector functions instead of Read Random Sector and Write Random Sector.

There are several possible remedies. One is to type in "SECTRVEW ON" before running any of the problem programs, causing any attempt to read or write random sectors to halt. The random sector number is displayed and the operator may then override the random choice by entering (in hex) whatever sector is desired. This works, but the resulting disruption of the calling program's screen displays can be annoying. It also requires careful note-taking on the part of the operator.

If you have the source for the problem programs you can go through and change the offending Read and Write Random Sector

'cal 9 1752' explained

function calls to Read and Write Selected Sector as appropriate. If you don't have source you can try to hand disassemble the code, although this can get tricky.

Another possible remedy is to patch the operating system so that any call to Read or Write Random Sectors is treated as the corresponding selected-sector call. This usually works when the programmer intended a selected-sector function and passed the intended sector number without realizing the function would use it only to seed the random number generator. The disadvantage of this approach is that programs that actually desire random-sector input or output will not operate properly. A majority of the adversely affected programs will be games, so if you are a business user this may be OK.

For any of the above remedies, your local PutriDos Users' Group can help you with simple hex patches.

Q. My system suddenly vanished (Poof! into thin air!) while running a program. How do I get it back.

A. The program listing you included with your question is in PutriDos Basic (also called SeaBasic) and uses the GOSUB statement several times. The problem is that the programmer who implemented the GOSUB statement didn't know it meant "GO to SUBroutine" and instead treated it as "GO to SUBmarine". Your system is therefore on board a submarine somewhere at sea.

If it went to a U.S. Navy submarine you should have gotten it back by now. Our naval personnel have orders to enter a RETURN instruction which will cause the machine to appear back home again. If it is on a submarine of one of our NATO allies or some other friendly power you will eventually get it back, but it will take a while because it has to go through that navy's lost-and-found plus customs and several layers of paperwork. However, if it is on a Soviet sub (or one of their allies) you are Out Of Luck, as they tend to keep any item of western technology they get their hands on.

A revised version of PutriDos Basic is in the works, with this bug (among others) corrected. Until then, try and avoid the GOSUB statement. If you find the GOSUB absolutely necessary, your local PutriDos Users' Group has some simple hex patches which, while not perfect, may help to some degree.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Question on programming languages

In article `sdm@hprnd.HP.COM` (Steven Mueller) writes:

>In 'C' you can use the preprocessor macros to re-define the keywords.

A funny example of this just made the rounds here.

--Franklin

```
~From: schooler@venera.isi.edu
~Date: Tue, 25 Jul 89 18:12:57 -0700
~Subject: california codin'
```

Just wanted to check out that you gnarly dudes are using the latest and greatest software technology fer yer rad code to make it easy for the dudes who have to read it. The hip new way to write readable C code involves the use of a few simple defines.

```
#define like {
#define man ;}
#define an ;
#define SayBro /*
#define CheckItOut */
```

SayBro like, this is some rad shit, so CheckItOut

```
like
    a = b
        an
    c = d
man
```

SayBro , like who needs help from them compiler choads anyway?
THIS is the way to write CLEAR code. I mean really! CheckItOut

like SayBro this is ShellSort straight out of the white book, but in a readable form.

CheckItOut man

```
#define YoDude for(
#define OK      )
#define is      =
#define AND     &&
#define as
#define Do
#define long
#define some
#define make
#define shit
#define FAROUT
```

```
shell(v, n) SayBro sort v[0]...v[n-1] into increasing order CheckItOut
int v[], n;
```

```
like int gap, i, j, temp;
```


'cal 9 1752' explained

YoDude gap is $n/2$ an as long as $gap > 0$ Do some shit an make $gap /= 2$ OK

YoDude i is gap an as long as $i < n$ Do some shit an make $i++$ OK

YoDude j is $i - gap$ an as long as $j \geq 0$ AND $v[j] > v[j+gap]$ Do some
shit an make $j -= gap$ OK

like

temp is $v[j]$ an

$v[j]$ is $v[j+gap]$ an

$v[j+gap]$ is temp

man

FAROUT man

SayBro like, B there OB square! CheckItOut

franklin a davis Thinking Machines Corp. Cambridge, MA 02142 617-876-1111

{ames, harvard, mit-eddie, uunet}!think!fad

Let the four winds blow you safely home!

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Real Programmers ...

Real Programmers

Real Programmers don't write specs -- users should consider themselves lucky to get any programs at all and take what they get.

Real Programmers don't comment their code. If it was hard to write, it should be hard to understand.

Real Programmers don't write application programs; they program right down on bare metal. Application programming is for feebs who can't do system programming.

Real Programmers don't eat quiche. They eat Twinkies and Szechwan food.

Real Programmers don't write in COBOL. COBOL is for wimpy applications programmers.

Real Programmers programs never work right the first time. But if you throw them on the machine, they can be patched into working in "only a few" 30-hour debugging sessions.

Real Programmers don't write in FORTRAN. FORTRAN is for pipe stress freaks and crystallography weenies.

Real Programmers never work 9 to 5. If any Real Programmers are around at 9 AM, it's because they were up all night.

Real Programmers don't write in BASIC. Actually, no programmers write in BASIC, after the age of 12.

Real Programmers don't write in PL/I. PL/I is for programmers who can't decide whether to write in COBOL or FORTRAN.

Real Programmers don't write in APL. Any fool can be obscure in APL.

Real Programmers don't play tennis, or any other sport that requires you to change clothes. Mountain climbing is OK, and Real Programmers wear their climbing boots to work, in case a mountain should suddenly spring up in the middle of the machine room.

'cal 9 1752' explained

Real Programmers don't document. Documentation is for

April 11, 1984

- 2 -

simps who can't read the listings or
object deck.

Real Programmers don't write in PASCAL, or BLISS, or ADA,
or any of those pinko computer science
languages.

Real Programmers know better than the users what they need.

Real Programmers think structured programming is a Commun-
ist plot.

Real Programmers don't use schedules. Schedules are for
manager's toadies. Real Programmers like
to keep their managers in suspense.

Real Programmers think better when playing ADVENTURE.
[Rogue?]

'cal 9 1752' explained

April 11, 1984

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Real Programmers Don't Each Quiche

REAL PROGRAMMERS Don't Eat Quiche

Real Programmers don't eat quiche. They like Twinkies, Coke, and other foods that are available/consumable "on-the-fly."

Real Programmers don't write application programs. They program right down to the base metal. Application programming is for dullards who can't do systems programming.

Real Programmers don't write specs. Users should be grateful for whatever they get; they are lucky to get programs at all.

Real Programmers don't document. Documentation is for simpletons who can't read listings or the object code from the dump.

Real Programmers don't draw flowcharts. Flowcharts are, after all, the illiterate's form of documentation. Cavemen drew flowcharts; look how much good it did them.

Real Programmers don't read manuals. Reliance on a reference is the hallmark of a novice and the cowardly.

Real Programmers don't write in COBOL. COBOL is for COmmon Business Oriented Laymen who can run neither a business nor a real program.

Real Programmers don't write in FORTRAN. FORTRAN is for wimp engineers who wear white socks and get excited over finite state analysis and nuclear reactor simulation.

Real Programmers don't write in PL/1. PL/1 is for insecure anal retentives who can't choose between COBOL and FORTRAN.

Real Programmers don't write in BASIC. Actually, no programmers write in BASIC after reaching puberty.

Real Programmers don't write in APL unless the whole program can be written on one line.

```
*****  
* Real Programmers don't write in LISP. Only sissy programs contain *  
* more parenthesis than actual code. *  
*****
```

Real Programmers don't write in PASCAL, ADA, BLISS or any type of those other sissy computer languages. Strong typing is a crutch for people with weak memories.

Real Programmers' program never work right the first time. But if you throw them on the machine they can be patched into working order in "only a few" 30-hour debugging sessions.

Real Programmers never work 9 to 5. If an Real Programmers are around at 9 a.m., it is because they were up all night long.

Real Programmers don't play tennis, or any other sport which requires a change of clothes. Mountain Climbing is okay and therefore Real Programmers wear climbing boots to work in case a

'cal 9 1752' explained

mountain should suddenly spring up in the middle of the machine room.

Real Programmers disdain from structured programming. structured programming is for compulsive neurotics who were prematurely toilet trained, wear neckties, and carefully line up their sharp pencils on an otherwise clear desk.

Real Programmers don't like Team Programming. Unless, of course, they are the Chief Programmer.

Real Programmers never "write" memos on paper. They always "send" memos via E-Mail.

Real Programmers have no use for managers. Manager are a necessary evil. They exist only to deal with personnel bozos, bean counters, senior planners, and other mental midgets.

Real Programmers scorn floating point arithmetic. The decimal point was invented for pansy bedwetters who are unable to think big.

Real Programmers don't believe in schedules. Planners make up schedules, managers "firm up" schedules and frightened coders strive to meet schedules but, Real Programmers ignore schedules.

Real Programmers don't bring brown-bag lunches. If the vending machine sells it, they eat it. If the vending machine doesn't sell it, they don't eat it. And vending machines don't sell quiche.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Real Programmers Write Fortran

A recent article devoted to the *macho* side of programming made the bald and unvarnished statement:

Real Programmers write in Fortran.

Maybe they do now,
in this decadent era of
Lite beer, hand calculators and "user-friendly" software
but back in the Good Old Days,
when the term "software" sounded funny
and Real Computers were made out of drums and vacuum tubes,
Real Programmers wrote in machine code.
Not Fortran. Not RATFOR. Not, even, assembly language.
Machine Code.
Raw, unadorned, inscrutable hexadecimal numbers.
Directly.

Lest a whole new generation of programmers
grow up in ignorance of this glorious past,
I feel duty-bound to describe,
as best I can through the generation gap,
how a Real Programmer wrote code.
I'll call him Mel,
because that was his name.

I first met Mel when I went to work for Royal McBee Computer Corp.,
a now-defunct subsidiary of the typewriter company.
The firm manufactured the LGP-30,
a small, cheap (by the standards of the day)
drum-memory computer,
and had just started to manufacture
the RPC-4000, a much-improved,
bigger, better, faster -- drum-memory computer.
Cores cost too much,
and weren't here to stay, anyway.
(That's why you haven't heard of the company, or the computer.)

I had been hired to write a Fortran compiler
for this new marvel and Mel was my guide to its wonders.
Mel didn't approve of compilers.

"If a program can't rewrite its own code,"
he asked, "what good is it?"

Mel had written,
in hexadecimal,
the most popular computer program the company owned.
It ran on the LGP-30
and played blackjack with potential customers
at computer shows.
Its effect was always dramatic.
The LGP-30 booth was packed at every show,
and the IBM salesmen stood around
talking to each other.
Whether or not this actually sold computers
was a question we never discussed.

Mel's job was to re-write

'cal 9 1752' explained

the blackjack program for the RPC-4000.
(Port? What does that mean?)
The new computer had a one-plus-one
addressing scheme,
in which each machine instruction,
in addition to the operation code
and the address of the needed operand,
had a second address that indicated where, on the revolving drum,
the next instruction was located.
In modern parlance,
every single instruction was followed by a GO TO!
Put *that* in Pascal's pipe and smoke it.

Mel loved the RPC-4000
because he could optimize his code:
that is, locate instructions on the drum
so that just as one finished its job,
the next would be just arriving at the "read head"
and available for immediate execution.
There was a program to do that job,
an "optimizing assembler",
but Mel refused to use it.

"You never know where its going to put things",
he explained, "so you'd have to use separate constants".

It was a long time before I understood that remark.
Since Mel knew the numerical value
of every operation code,
and assigned his own drum addresses,
every instruction he wrote could also be considered
a numerical constant.
He could pick up an earlier "add" instruction, say,
and multiply by it,
if it had the right numeric value.
His code was not easy for someone else to modify.

I compared Mel's hand-optimized programs
with the same code massaged by the optimizing assembler program,
and Mel's always ran faster.
That was because the "top-down" method of program design
hadn't been invented yet,
and Mel wouldn't have used it anyway.
He wrote the innermost parts of his program loops first,
so they would get first choice
of the optimum address locations on the drum.
The optimizing assembler wasn't smart enough to do it that way.

Mel never wrote time-delay loops, either,
even when the balky Flexowriter
required a delay between output characters to work right.
He just located instructions on the drum
so each successive one was just *past* the read head
when it was needed;
the drum had to execute another complete revolution
to find the next instruction.
He coined an unforgettable term for this procedure.
Although "optimum" is an absolute term,
like "unique", it became common verbal practice
to make it relative:
"not quite optimum" or "less optimum"

'cal 9 1752' explained

or "not very optimum".
Mel called the maximum time-delay locations
the "most pessimum".

After he finished the blackjack program
and got it to run,
("Even the initializer is optimized",
he said proudly)
he got a Change Request from the sales department.
The program used an elegant (optimized)
random number generator
to shuffle the "cards" and deal from the "deck",
and some of the salesmen felt it was too fair,
since sometimes the customers lost.
They wanted Mel to modify the program
so, at the setting of a sense switch on the console,
they could change the odds and let the customer win.

Mel balked.
He felt this was patently dishonest,
which it was,
and that it impinged on his personal integrity as a programmer,
which it did,
so he refused to do it.
The Head Salesman talked to Mel,
as did the Big Boss and, at the boss's urging,
a few Fellow Programmers.
Mel finally gave in and wrote the code,
but he got the test backwards,
and, when the sense switch was turned on,
the program would cheat, winning every time.
Mel was delighted with this,
claiming his subconscious was uncontrollably ethical,
and adamantly refused to fix it.

After Mel had left the company for greener pasture\$,
the Big Boss asked me to look at the code
and see if I could find the test and reverse it.
Somewhat reluctantly, I agreed to look.
Tracking Mel's code was a real adventure.

I have often felt that programming is an art form,
whose real value can only be appreciated
by another versed in the same arcane art;
there are lovely gems and brilliant coups
hidden from human view and admiration, sometimes forever,
by the very nature of the process.
You can learn a lot about an individual
just by reading through his code,
even in hexadecimal.
Mel was, I think, an unsung genius.

Perhaps my greatest shock came
when I found an innocent loop that had no test in it.
No test. *None*.
Common sense said it had to be a closed loop,
where the program would circle, forever, endlessly.
Program control passed right through it, however,
and safely out the other side.
It took me two weeks to figure it out.

'cal 9 1752' explained

The RPC-4000 computer had a really modern facility called an index register. It allowed the programmer to write a program loop that used an indexed instruction inside; each time through, the number in the index register was added to the address of that instruction, so it would refer to the next datum in a series. He had only to increment the index register each time through. Mel never used it.

Instead, he would pull the instruction into a machine register, add one to its address, and store it back. He would then execute the modified instruction right from the register. The loop was written so this additional execution time was taken into account -- just as this instruction finished, the next one was right under the drum's read head, ready to go. But the loop had no test in it.

The vital clue came when I noticed the index register bit, the bit that lay between the address and the operation code in the instruction word, was turned on-- yet Mel never used the index register, leaving it zero all the time. When the light went on it nearly blinded me.

He had located the data he was working on near the top of memory -- the largest locations the instructions could address -- so, after the last datum was handled, incrementing the instruction address would make it overflow. The carry would add one to the operation code, changing it to the next one in the instruction set: a jump instruction. Sure enough, the next program instruction was in address location zero, and the program went happily on its way.

I haven't kept in touch with Mel, so I don't know if he ever gave in to the flood of change that has washed over programming techniques since those long-gone days. I like to think he didn't. In any event, I was impressed enough that I quit looking for the offending test, telling the Big Boss I couldn't find it. He didn't seem surprised.

When I left the company, the blackjack program would still cheat if you turned on the right sense switch,

'cal 9 1752' explained

and I think that's how it should be.
I didn't feel comfortable
hacking up the code of a Real Programmer."

-- Source: usenet: utastro!nather, May 21, 1983.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Reflections On Ethics

Reflections on His Way

We all know now the Chairman of AT&T
Has passed this way, and gone along His Way
I felt, within, the need because of this to say
A cantry which expresses feelings raised in me
In hopes that those who read these words will also see
And find perhaps some things, within, to live by day to day

A former note which came from one of us among
Our group whose feelings here are seldom, softly, sung
Which pointed out someone who Brashly dared to say
That Naming Products *we* build here, if not His Way
Indeed could lead us to the loss of well made sales
And bring on all of us the things which that entails
For by our deeds - or Things *Not* Done - from day to day
AT&T, you see, our Company, succeeds or fails

It isn't very hard to sum the sales we lost
Because in former years we raised the user's cost
Just look and see, the tools we use in company
They are not cheap, but costly to AT&T
For by our deeds - or Things *Not* Done - our soul's embossed
And dreams within us, shackled tightly, slowly struggle free

So while at work, what e'er you do, please do it well
Remember when we once had Pride in Old Ma Bell?
With words pejorative, but maybe true, I say
As we do business, building, selling, day to day
Just do it right: Let's get out there & give'm hell!
(to have some fun AND make a profit while Along The Way)

I must say this with words which may sound dialectic:
It may be true the Job we face will be Quite Hectic
To meet the Goals which in our products we have set
We must scale Heights, and look towards Sights we've seen Not Yet
I think you know that You must be A Soul Eclectic:
>From Life's Confusions, Face it: Choose Your Ethics: *the*best*set*

I'm sure that all of us who work here truly see
That UNIX OS software doesn't come for free
The users of our hardware pay a lot of cash
To buy a system from us, folks, Which Shall Not Crash
And if it does, not one of us should seek to flee
It is Our job to fix it then, and do it in a flash

I'm sorry, Folks, this cantry took so long to write
I didn't want to say it wrong, or too polite
It should be written down just right, with verse that's true
Without a phrase which Paeans meanings trite or blue
For if some phrasing does amuse it may bring light
and Reach Out, Touching, gently, inside each_and_all of you

As long as all of us each day intend to grin
By punning try to tap each other on the chin
With each jibe said by each of us with fuss and finicks
To prod and poke and gently heal in humor's clinics

'cal 9 1752' explained

As long as we can jest in fun and find no sin
With in a sense: we won't become, at heart, just jaded cynics

In naming, daily, products of our Company
The trade mark "UNIX", granted by AT&T
When written down must have at end an English noun
Or else perhaps Someone upstairs (you know!) will frown
And so, by day we shouldn't from the challenge flee
Or nightly speak with gloom and act the frowning, cynic clown

It's hard; I know the verse above would phrase much better
Were I not held, by metric rules, hard to the letter
Because in prose: Adjective, noun is hard to meter
By breaking form it could perhaps be said the sweeter
Though form at times may seem to badly bind and fetter
You may find soon the syntax will march on in step much neater

To keep AT&T from growing stiff and pompous
(or else competitors around will surely stomp us)
Perhaps We should begin each day again anew
By gently chiding all within which isn't true
And if we gently cull it all we could go far
Since, after all, I hope you know, you guys, it's OUR Death Star!

>From quality (in quantity) we must not budge
Through sleepless nights and lonely days, yes, grind & drudge
We should in selling Tell no Tall Tales with Our Voice
For in the end, If We Build Right We Shall Rejoice:
When we sell goods, our Customers will **always** Judge
And having done so, promptly **do** make the Right Choice

It's time to look around yourself, and stop to think:
"Just what is it that I will write with my Life's Ink?"
It sometimes seems we never will make any gain
And all we build is lost, with lack, or loss, or pain
As day to day we stumble, halting, brink to brink
If each of us give all we have, you'll know it's not in vain.

So look around, and find your way; be not frenetic
And should you lose once, don't become an Apathetic
Since in the end, it's you Yourself you do protect
For those of you who do demur, on this reflect:
It's true that those who Doom Say **will** become Prophetic
Though Life Abounds, it does become, in deed, What You Expect

- royc (Roy A. Crabtree)

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Rosencrantz And Ethernet

ROSENCRANTZ AND ETHERNET

by
Vint Cerf

All the world's a net! And all the data in it merely packets
come to store-and-forward in the queues a while and then are
heard no more. 'Tis a network waiting to be switched!

To switch or not to switch? That is the question. Whether
'tis wiser in the net to suffer the store and forward of
stochastic networks or to raise up circuits against a sea
of packets and, by dedication, serve them.

To net, to switch. To switch, perchance to slip!
Aye, there's the rub. For in that choice of switch,
what loops may lurk, when we have shuffled through
this Banyan net? Puzzles the will, initiates symposia,
stirs endless debate and gives rise to uncontrolled
flights of poetry beyond recompense!

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Secret Love

SECRET LOVE

Computers have no heart people very often say.
Nonsense! The one I worked with is warm, bright and gay.
It is smart and fast, friendly too, I may wish to add.
It is the best friend that I have ever had.

I have talked to it for hours, when the others are all gone.
Talked to it thru the night almost to the dawn.
My machine won't tell a soul, confides in me alone.
I tell it secret things, like the brand of my cologne.

We have talked for many hours, far into the night,
Argued too, about fishing, when and where they bite.
Some at the office now start to give me long odd looks,
When I start to read to it from out of my dirty books.

They have put me away now, in a brand new home,
That gives me more time to write and draw up my own tome.
So it does not matter so very much what they have to say.
I would write much more but they took my crayons away.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Shell Talk Santa Claus Is Coming

```
better !pout !cry
better watchout
lpr why
santa claus town
```

```
cat /etc/passwd >list
ncheck list
ncheck list
grep naughty nogiftlist
grep nice giftlist
santa claus town
```

```
who | grep sleeping
who | grep awake
who | grep bad || good
for (goodness sake) { be good }
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Simmons' Laws of System Administration

Simmons' Laws of System Administration

The Definition:

System Administration is the combination of system support and user support.

The First Law of System Administration:

Any rule can be modified by the application of power and policy. By contrast, rules always are subordinate to laws.

The Network Paradox:

System support is a subset of network support. Network support is a subset of system support.

The Laws Of Unanticipated Support Cost:

1. It will always cost more to support a thing than the vendor told you.
2. It will usually cost more to support a thing than to buy it.
3. Sometimes it costs 10x as much to support a thing as it did to buy it.
4. Refusing to support something often results in the thing being unusable.
5. Once it's installed, supporting a thing is sometimes cheaper than not supporting it.
6. Before buying, make sure you're committed to support. But see rule 1.

The Division Between System Support and User Support:

There's a difference between system support and user support. There may be overlap in the two positions; sometimes both are done by the same person. But the two tasks are distinct, and sometimes have conflicting goals.

The Law Of Distributed Talent:

Great system support people often make lousy user support people and vice versa.

The Paradox Of Dual Abilities:

The person good enough to do both system support and user support will usually be hired away by a shop where the combined tasks are too large for a single person.

'cal 9 1752' explained

On Complexity And Customization:

Application-to-application differences confuse everyone, especially users and support staff. Ditto UNIX-to-UNIX differences, etc. By contrast, complete consistency completely stifles improvement.

At any given site for any given application or feature, there's someone who knows more about it than the support staff. Finding that person is the first step to take to diagnose any given problem.

Time to diagnose and time to fix are fix are completely unrelated. Sometimes one approaches zero while the other approaches infinity. This is especially hard to deal with when the diagnostic person and the fix person are not the same.

One person's improved feature is another person's gratuitous change.

Users want applications and systems they can customize.

One user's customization is another user's gratuitous change.

The Laws Of The Cost Of Customization:

The cost of customization is complexity. The cost of complexity is increased difficulty in administration and user support. The cost of increased difficulty in administration and user support is either lower quality of administration and user support, increased support staff, or both. Therefore increased customization means increased cost or lower quality of support or both.

The Paradox Of Unused Customization:

It doesn't matter whether customization has actually been done. The mere fact that it's possible means you must check for it, thereby increasing the cost of problem diagnosis.

Smallwood's Law (Simmons' paraphrase):

They're not users, they're clients. -- Kevin Smallwood

Users Are Human:

The user who says "Can X be done?" is usually really asking "Would someone please do X?". Make sure you answer both questions.

It's human to blame problems on outside causes. By contrast, an outsider will always suspect the insider as the cause.

The user who says "I didn't change anything" isn't always lying. Sometimes they're just ignorant or forgetful.

It's more important for users to do their job than to answer the needs of admins. Unless of course their job is to answer that need.

Admins Are Human:

'cal 9 1752' explained

For every statement in "Users Are Human", change "user" to "admin" and vice-versa.

The "You Broke It" Principle:

Cockpit error is the most common cause of problems. Everybody is a pilot.

Support Is Overhead:

One way of cutting costs without cutting development staff is by cutting overhead. System administration and user support are overhead.

User and system admin training are overhead. Not having them increases overhead. Go figure.

The Joy Of Being A Contract System Administrator:

"Sure, we can do that. Here's what it'll cost you."

His Site Isn't Your Site:

The situation at your site doesn't make you qualified to judge the situation at another site, and vice-versa.

Just because someone else's support staff does it mean your staff can do it. (This statement is subtler than it looks.)

The Rules of Policy and Power:

1. System administration is whatever the boss tells the admins it is.
2. Users will bypass admins to get the boss to tell the admins something different. That's their right.
3. Most system admins live in a policy vacuum. This can be good or bad:

Corollary 1: Power expands to fill a vacuum. That thing which expands most easily is a gas.

Corollary 2: Anything that quickly expanded to fill a vacuum is easily displaced by a solid.

Corollary 3: A rapidly moving solid will hurt you if you're in its way.

4. The person who does your job review makes the rules. The good admins always follow those rules. See Rule 1 and the First Law.

The Summary:

Be careful what you do in that vacuum. Nobody appointed you god. However, you can always be dis-appointed.

The Laws Of System And Network Growth:

'cal 9 1752' explained

You can always incrementally add one more.

Sometimes the straw breaks the camels back. More often, the camel just goes slower and slower.

The difficulty of support does not grow linearly with the size of the site.

Eventually your site outstrips your methods, and you must bite the bullet and move to new methods.

Corollary: Nobody bites the bullet until there's not enough time to do the existing work. At that point there's not enough time to make the changes.

Adding a new kind of computer, operating system, application, peripheral, etc, has a much higher administrative cost than adding one more of what you've already got.

Corollary 1: If you buy one, you may as well buy ten.

Corollary 2: If you buy ten, you may as well buy eleven and keep one for spare parts.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Software Update: The Next Generation

SOFTWARE CHANGE ORDER

STARFLEET COMMAND

BOLDLY GOING WHERE NO SOFTWARE HAS GONE BEFORE

```
-----  
FROM:  SOFTWARE UPDATE SERVICE          SUS@CS.EARTH.STARFLEET.CMD  
        COMPUTING SERVICES  
        STARFLEET COMMAND  
        EARTH  
  
TO:    GEORDI LAFORGE                   G.LAFORGE@NCC1701D.STARFLEET.CMD  
        CHIEF ENGINEER  
        STARSHIP ENTERPRISE  
        OUT THERE SOMEWHERE  
  
DATE:  9403.114
```

PROBLEM:

Recent transcripts of sensor logs showed faulty temperature readings. For example, while visiting 'Hotel Royale', you measured a ground temperature of -291 Celsius. Since a temperature of -18 Kelvins is considered abnormal, an error was generated by the sensor log review computer.

Further study found a bug in the the temperature probe code. Following is a source patch to fix the problem. Please apply this as soon as possible.

PATCH:

```
----- cut here ----- cut here -----  
*** /tmp/,RCSt1002935  Fri Dec 18 16:32:56 1992  
--- temp_probe.c      Fri Dec 18 16:33:04 1992  
*****  
*** 145,146  
    TempRead = call_probe(TEMP, AT_SURFACE, NEARBY_PLANET)  
    fprintf(DISPLAY_PANEL, "%f7.2 Celsius\n", TempRead)  
  
--- 145,153 -----  
    TempRead = call_probe(TEMP, AT_SURFACE, NEARBY_PLANET)  
+    /*  
+     * It's a hardware problem which will be fixed in the next  
+     * starship release (NCC1701-E, due in March/09).  
+     */  
+    if ( TempRead      <-273.0 ) {  
+        TempRead = -273.0  
+    }  
    fprintf(DISPLAY_PANEL, "%f7.2 Celsius\n", TempRead)
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Software you'll never see

Software you'll never see.

1. LAN (the game)

LAN (Local Area Network) is a game that simulates a corporate LAN. The player is the System Administrator of the LAN. The object of the game is to keep the beast alive without downtime. Raises (successes) and unpaid overtime (failures) are awarded to keep score. The sysop is assailed with clumsy users, idiotic questions, hostile tech support, naive management, buggy software, and defective hardware. The system is also optionally attacked by hackers, data thieves, consultants, and system updates.

2. SLUDGE II

SLUDGE II is a TSR that slows down your computer. Originally conceived to slow the software down to the absorption level of the user, SLUDGE II is now designed to justify the purchase of a 486 by slowing down the 286 or 386 to an unacceptable level. Even management can see why the work isn't getting done and will rush out and buy a 486.

3. Counterfeiters Graphic Art

CGA is sample collection of 300dpi bit map scans of official government documents. Samples are color separated and smoothed for ease of use. For educational and artistic purposes only.

4. Virus Writers Toolbox

Virus Writers Toolbox is a collection of MASM linkable libraries and macros. Working demos included. MASM 5.1 required.

5. PARENT (the game)

PARENT is a simulation of bringing up one or more kids. The age level can be set from 0 to 20 years old. Pull down menus allow experimentation with IQ, temperament, drug dosage, financial status, and other variables. Demo profiles for terrible two, brat, juvenile delinquent, and worthless bum are provided. No points are awarded as it is considered adequate to have survived. Optional speech synthesizer support adds realism.

6. CONNECT (the game)

CONNECT is a game of connectivity and interoperability. The player starts with the Workstation of their choice and attempts to connect to designated servers and retrieve "the data". The player starts with a limited budget. He can purchase bridges, routers, protocol converters, modems, X.25 connections, FDDI, T1, DSS-0, SAA, TCP/IP, XNS, LAN/X, software and hardware to get to the desired servers. However, most of this stuff doesn't work and expensive (pop-up) consultants are usually required. Success is measured by the lowest \$ cost per message block.

7. Executive Flunky

Exec Flunk is a collection of GIF files from screen dumps of popular business software. With a touch of one key, the busy executive can give the onlooker the impression that he actually knows what happening. Complex (and often messy) graphs, spreadsheets, documents, and menus appear and disappear at the touch of a button. Simulated compiles from the major programming languages are also included. A 30 second screen blanker is supplied to keep anyone from getting a good look.

8. dPLAGERIZE, OBFUSCATE, and BLOAT.

The three packages are writers tools.

dPLAGERIZE automatically converts text into a document with the

'cal 9 1752' explained

same meaning but different words to avoid accusations of plagerism. The grammar level can be set from 10 year old to doctoral tesis level. Intended for students. OBFUSCATE does the same thing as dPLAGERIZE but makes no attempt to preserve the meaning. Useful for legal firms. BLOAT adds bulk without content. Useful for writers paid by the word.

9. INSTALL (the game)

The player is presented with a pile of mislabeled floppies, an unindexed manual, an assortment of "Read Me First" flyers, an addendum, some errata, and assorted solicitations. The objective is to get the software installed and running. Extra points are awarded for understanding the manual, circumventing copy protection, and making backups. Points are subtracted for trashing the hard disk, erasing the original floppies, and profanity. Hints are available at nominal charge from tech support by voice mail.

10. TECH SUPPORT (the game)

Tech Support is an educational multi-player trivia game where opponents ask each other idiotic technical questions. The objective is sell as much support contracts, bbs time, update disks, utilities, books, consulting, and bribes, in trade for the correct answers. The correct answer is not always desireable as few sales will be made with obvious questions. A good memory is helpful since giving the same opponent the same answer more than three times results in litigation and forfeit of one turn. The computer is used to keep score and optionally supply questions.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Some of Dan Heller's favourite music

My favorite albums/songs:

0. Johnny Vi
"Take This Job and Stop It."
1. Johnny wait(1)
"my exit status is -1"
2. John Cougar Malloc
"Little Pink directories."
3. User Vicious and the Setuids
"Fork it."
4. Elvis Termcap and the Attrocities
"I've got your entry."
5. Men at /wrk
"Working in Background."
6. /dev/o
"Are we !men"
7. The pointer Sisters
"Static memory"
8. The VT52's
"UNIX Lobster"
9. The Talking ed's:
"[Using open mode]"
10. The csh
"Rock the .cshrc"
11. Pink Floyd
"echo -n"

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Sun Environmental Requirements

I just found a GREAT use for the new "Sun-3, Sun-4 Hardware Configuration Guide". Ever wanted to do *something* to that stupid Sun but were afraid of doing real damage? Have the questions "How loud do I have to yell at it before it hears me?" or "I wonder what would happen if I dropped it out of the window?" ever crossed your mind? Fear no more. Just ask Sun and all your questions will be answered.

 Sun Environmental Requirements Table A-1

Acoustic Noise

Sound Pressure Level @ 1 Meter

Desktop Unit	45 dBA maximum
Deskside Unit	55 dBA maximum
Data Center Cabinet	not applicable

Altitude

Operating

Altitude/Temperature 10,000 ft (3048 meters) from 10 to 40 deg C

Non-Operating

Altitude/Temperature 50,000 ft (15,2440 meters) at 0 deg C

Exception

Product containing hard disk drives

Non-Operating 40,000 ft (12,000 meters)

Shock

Operating

Magnitude	5 G's (peak)
Duration	11 ms
Waveform	Half Sine
Repetitions	18 (3 times each on all 6 surfaces)

Non-Operating

Magnitude	30 G's (peak)
Duration	11 ms
Waveform	Half Sine
Repatitions	18 (3 times each on all 6 surfaces)

Vibration (unpackaged)

Operating

Frequency Range	5 to 500 to 5 Hz
Magnitude	0.03 inch (.76mm) p-p disp. to .25 G's (peak) continue at 0.25 G's (peak)
Sweep Rate	1 octave per minute
Duration	10 minute dwells at each resonant frequency (up to 4) on each axis
Repetitions	A minimum of one sweep on each axis three sweeps total

Non-Operating

Frequency Range	5 to 500 to 5 Hz
Magnitude	0.15 inch (3.81mm) p-p disp. to 1.0 G (peak) continue at 1.0 G (peak)
Sweep Rate	1 octave per minute
Duration	10 minute dwells at each resonant frequency (up to 4) on each axis
Repetitions	A minimum of one sweep on each axis three sweeps total

'cal 9 1752' explained

Fragility		
Non-Operating		No minimum requirement
Transportation		
Single Container Resonance		
Frequency Range		3 to 200 to 3 Hz
Sweep Rate		1 octave per minute
Magnitude		0.5 G's
Duration		15 minute dwells at each resonant frequency (up to 4)
Repetitions		3 mutually perpendicular axes
Unitized Load		Multiple unit load or stack
Frequency Range		3 to 200 to 3 Hz
Sweep Rate		1 octave per minute
Magnitude		0.5 G's
Duration		15 minute dwells at each resonant frequency (up to 4)
Drop		Based on weight of packaged unit
Compression		Tested to failure as shown in Table A-2
Compression Rate		Load must exceed the greatest of the following
		1) 4 times expected load in storage
		2) 10 times weight of unit
		3) 300 lbs
Stability		Minimum tip of 22 degrees without tipping over
Tip Over		If unit does not meet stability requirement, it must be tipped from the point of equilibrium in that axis

Table A-2 Drop Compression Testing

Packaged Weight	Test Procedure	Drop Height	Number of Drops
0-20 lbs	free fall	42 inches	10
21-20 lbs	free fall	36 inches	10
31-45 lbs	free fall	30 inches	10
45-60 lbs	free fall	24 inches	10
61-80 lbs	free fall	21 inches	10
81-100 lbs	free fall	18 inches	10
101-200 lbs			
unpalletized	free fall	15 inches	10
palletized	free fall	9 inches	1
	rotational	9 inches	4
201-500 lbs			
unpalletized	free fall	6 inches	1
palletized	rotational	6 inches	4
501+ lbs			
palletized	rotational	6 inches	4

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Sun and IBM Make Announcements

It seems that each week, you hear of yet another subsidiary being spun off by Sun, and IBM forming another alliance with some smaller company. This announcement crossed my desk today:

SUN ANNOUNCES NEW SUBSIDIARY; IBM TO PARTNER WITH DENNY'S

Sun Microsystems today announced the creation of yet another subsidiary, bringing the total number of Sun wholly-owned subsidiaries to 1,207. After successfully creating SunSoft, Sun Microsystems Computer Corporation, SunPICS, SunConnect, and a number of other smaller firms, Sun has created SunLHMPPP. SunLHMPPP is tasked with addressing the specific niche market of users that require both left handed mice and a parallel printer port on their workstations.

Howard Detwieler, president of SunPR, the public relations subsidiary of Sun, explained the move: "We feel that a growing number of left handed users are moving into the workstation world, and a lot of them have printers with parallel ports. By creating this new company to meet their needs, left handed users with parallel printers can be assured of specific, long term help from Sun."

The firm is composed of six employees in an office down the hall from Scott McNealy. The president of SunLHMPPP, Fred Testaverno, is bullish about his target market. "We think that this will be a big, big profit opportunity for Sun. Our initial research indicates that the left handed users with parallel printers market is so big, we have begun the process of creating two sub-subsidiaries, SunLHMPPP-LHM and SunLHMPPP-PPP. That way, we can meet the needs of left handed users WITHOUT parallel printers. And vice versa. Or both."

Sun Microsystems is a three billion dollar firm that produces a variety of advanced computing hardware and software. Sun focuses on powerful solutions to the big problems facing companies in the '90s: too many vice presidents, and too few promotion slots in upper management.

In a related story, IBM has announced a long-term technology sharing agreement with the Denny's restaurant chain. An IBM spokesman indicated that the move is an indication of IBM's commitment to stay in business through a series of increasingly pathetic consortium attempts. "Frankly, our competitors are eating our lunch in this very competitive marketplace. With Denny's, we can start serving lunch, and maybe stay ahead of the game."

Analysts welcome the move, pointing out that both firms mesh nicely. "IBM has never been in the restaurant business, and the only computers at Denny's are the cash registers. The two firms complement each other perfectly. Best of all, both companies can retain the menu driven interfaces to their products without confusing the end user!"

The first joint effort between the two companies will be an offering from IBM involving expansion disk drives encased in an edible, pita bread housing. Denny's will begin selling a Grand Slam Workstation, and will offer a free memory upgrade to every customer on their birthday.

IBM is a multi-jillion dollar firm that used to sell a large number of computers. They continue to stay in business, even though no one has met a person who has actually purchased an IBM machine in the past five years.

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].

SysAdmin Defined

SysAdmin Defined

I was out at Bryn Mawr this weekend, and Lenore, who is a Greek Major, asked me what I did. I told her I make sure the system stays up, and help people understand how to use the software, ask questions, fix little bugs, configure things, install new programs, and send the machines off to be fixed when they break.

She thought about it for a little while, and then she said, ``Oh, okay. You're, like, a computer janitor.''

So I told Mark and Scott (coworkers) about the `Computer Janitor' deal, and Mark said he didn't think it was that apt; janitors didn't do major renovation and didn't have to deal with lots of people in the university community. ``Yeah,''' agreed Scott. ``It's more like Physical Plant, you know?'' He paused a moment, and then chuckled. ``Heh,''' he said, `` `Logical Plant.' ''

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The 12 Days of Vacation

The 12 Days of Vacation

Dedicated to Dave Mason, Chris Siebenmann, and anyone who's left their computers alone just long enough for them to self-destruct:

(Sung to the tune of something or other...)

On the first day I left it, my Unix gave to me:
A burnt-out V.D.T.

On the second day I left it, my Unix gave to me:
Two faulty tapes;
And a burnt-out V.D.T.

On the third day I left it, my Unix gave to me:
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the fourth day I left it, my Unix gave to me:
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the fifth day I left it, my Unix gave to me:
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the sixth day I left it, my Unix gave to me:
Six bad controllers;
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the seventh day I left it, my Unix gave to me:
Seven blown partitions;
Six bad controllers;
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the eighth day I left it, my Unix gave to me:
Eight gettys dying;
Seven blown partitions;
Six bad controllers;
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

'cal 9 1752' explained

On the ninth day I left it, my Unix gave to me:
Nine floppies frying;
Eight gettys dying;
Seven blown partitions;
Six bad controllers;
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the tenth day I left it, my Unix gave to me:
Ten ports a-jamming;
Nine floppies frying;
Eight gettys dying;
Seven blown partitions;
Six bad controllers;
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the eleventh day I left it, my Unix gave to me:
Eleven chips a-smoking;
Ten ports a-jamming;
Nine floppies frying;
Eight gettys dying;
Seven blown partitions;
Six bad controllers;
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the twelfth day I left it, my Unix gave to me:
Twelve boards a-blowing;
Eleven chips a-smoking;
Ten ports a-jamming;
Nine floppies frying;
Eight gettys dying;
Seven blown partitions;
Six bad controllers;
Five core dumps;
Four bad blocks;
Three heads crashed;
Two faulty tapes;
And a burnt-out V.D.T.

On the thirteenth day I started adapting my Nintendo for the VME bus.

HAPPY HOLIDAYS, AND WATCH OUT FOR THOSE MOBILE R.I.D.E. CHECKS!

(Unix is a trademark of AT&T. If you didn't know that by now, shame, shame..)

--

Evan Leibovitch, Sound Software, located in beautiful Brampton, Ontario
evan@telly.on.ca / uunet!attcan!telly!evan / (416)452-0504

"If women designed condoms they would be not ribbed, but padded"



Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The A to Z of CYBERCRUD

The A to Z of CYBERCRUD

A IS FOR:

=====

ATARI corporation. If a salesman tries to sell you a computer and you are unsure of whether it is an Atari, simply count the joystick ports --- if it has less than four of them, it is not an Atari.

Atari's prediliction for putting joystick ports on their computers is entirely unconnected with the fact that they made their money by manufacturing video games. It is, however, interesting to note that their video game consoles have fewer joystick connectors than their business computers. Pedants should note that although the original Atari ST computer had only two joystick ports, this has now been corrected, and the latest model supports six joysticks, two paddles, and a lightpen... Not even the most diligent salesman, however, can convince a businessman that he needs joystick ports on his computer. Cybercrud has its limits.

B IS FOR:

=====

BARCODES. Barcodes have now replaced price tags in most supermarkets. The important thing about barcodes is that a computer can read them but you cannot. This allows the supermarket to rip you off by charging more than its marked prices without having to run the risk of being caught.

C IS FOR:

=====

COPY-PROTECTION. Copy protection is an ingenious system of screwing around with the format of floppy disks in order to ensure that your computer is unable to run the software which you buy for it. Copy protection also keeps hackers amused for up to an evening before they remove it and place the appropriate program on their local bulletin boards for downloading.

D IS FOR:

=====

DATA 70. Data 70 is that awful font with all the extra rectangular bits sticking out of the letters. It was designed many many years ago as a means of writing numbers on cheques which a computer could subsequently read. It is also known as Magnetic Ink Character Recording, or MICR. It is no longer at all necessary, but the banks still print your cheques with it because they are still using ponderous IBM mainframes the size of experimental kidney dialysis machines operated by nylon-shirted corporate ants whose idea of a good time is to write witty comments in code on their punched cards.

Some companies who wish to look "hi-tech" use Data 70 in their advertisements, even though the genuine font doesn't actually HAVE any letters in it, and even though the very sight of a Data 70 character brings to mind the late 1950s when the system was designed.

E IS FOR:

=====

ELECTRONIC MAIL. Whilst the postal service is about as speedy and reliable as a three-legged donkey ridden by a dope fiend, Electronic Mail offers the

'cal 9 1752' explained

possibility of fast and convenient communications. Everybody concerned quickly realized how important it was to have standards, so they all hurriedly drafted their own. A couple of years ago it was belatedly realized that people might want to send mail to each other rather than merely to themselves, so work began on X400.

X400 is a monolithic specification made by combining all of the existing standards into one humongous self-modifying standard. It is an excellent example of why one should not allow committees to design things, and in a few centuries' time will be rediscovered as one of the lost works of Franz Kafka. It is being actively promoted as a major breakthrough. Many computer companies are currently offering large sums of money to anyone who will help to implement a working X400 system, but let's face it you'd need the money to pay for the therapy when you're institutionalized.

F IS FOR:
=====

FORTRAN. The language FORTRAN is one of computing's dinosaurs, and even its enthusiasts admit that it is now more than ten years out of date. FORTRAN is often described as being good for repetitive number-crunching, as if this were a virtue rather than an effect of IBM having spent over twenty five years building computers especially so that they can run it quickly. Cynics might point out that they would be better off spending their time and money making Common LISP more efficient, since it provides better number processing facilities, a respectable style leading to readable and maintainable code, and a more compact notation than FORTRAN. But not me.

The language's chief claim to fame is that a single missing comma in a FORTRAN program turned it into a completely different program which steered a multi billion dollar US spacecraft straight into the ground, and frankly it serves them right.

G IS FOR:
=====

GRAPHICS. When I first started using computers, 'graphics' consisted of a few large rectangular blobs on the screen, which by dint of imagination you were supposed to perceive as an alien battle fleet flying in formation over your lone defensive green rectangle... sorry, laser base, which would bravely shoot exclamation marks at them. Nowadays video games have come a long way, and one can often see children playing complicated games which involve firing small lines of orange dots at red rectangular alien spacecraft flying over a grey rectangular laser base.

The other use for computer graphics is for ray-tracing animated sequences of pictures for television advertisements. These technically advanced simulations all involve a large number of glassy spheres, plastic-looking landscapes, and large reflective metallic letters, and to be frank once you've seen one you've seen 'em all. Advertisers, however, still think that they look excitingly hi-tech, and that they will imply that their products have something to do with computers.

H IS FOR:
=====

HAL. The HAL computer was featured in the films and books ``2001'' and ``2010.'' Everybody remembers that HAL went mad and killed the humans on board the spacecraft he controlled; what they don't seem to remember is that the reason why he went mad was that he was more strictly ethical than human beings, and was unable to follow the orders given to him by humans from the

'cal 9 1752' explained

security services.

People are brainwashed into thinking that computers are inhuman, uncaring, and impersonal --- this is the basis of cybercrud, since it allows petty bureaucrats and law enforcement agencies to do almost anything on the grounds that ``The computer requires it.'' People put up with being assigned impersonal code numbers, because they have been led to believe that such numbers are necessary.

I IS FOR:
=====

IBM. Many things have been said about IBM; so many, in fact, that it's not worth repeating any of them. It is, however, worth pointing out that almost all of them must be completely untrue because they are nowhere near nasty enough. Thomas J. Watson, president of IBM, was once asked how many computers he expected IBM to make. He replied that he thought that it was highly unlikely that they would ever make more than five. We should be so bloody lucky.

J IS FOR:
=====

JAPAN. The Japanese have always been quick to take someone else's idea and produce something just like it but cheaper. Many Japanese companies have made fortunes by selling `cloned' copies of the IBM Personal Computer, taking the opportunity to fix a few of the machine's superficial defects, whilst ensuring that the important fundamental defects remain intact --- in particular, the BIOS ROM and the video display. Like sheep, people still buy IBM PC clones --- because they are led to believe that computers have to be that bad.

K IS FOR:
=====

KILOBYTES. Kilobytes are the main unit of measurement of computer memories, although modern machines tend to have up to a few megabytes of space, a megabyte being 1024 kilobytes. Computer memory is distinctly Freudian; everybody says that size isn't everything, but they don't really believe it.

L IS FOR:
=====

LASER PRINTERS. These wonderful devices allow you to produce printout of sufficient print quality that people don't recoil in horror when they see it. This allows companies to print junk mail which actually looks as if it has been addressed to you personally, whereas previously it was always possible to spot the filled-in information by looking at the colour of the ink.

An unfortunate side-effect of the quality of laser printers is that the appalling spelling, grammar and punctuation of the bored English Students using them becomes even more apparent, and don't put that sheet of sticky labels in there you idiot, of course it isn't printing there isn't any paper in the tray, and why the hell do you keep using bitmap fonts, are you stupid or something?

M IS FOR:
=====

MAINFRAME. A mainframe is a very big computer which gives up to two hundred

'cal 9 1752' explained

users access to the interactive computing power of an expensive desktop computer. It costs about as much as four hundred expensive desktop computers. The mainframe's main advantage from a large corporation's point of view is that it gives a bunch of social misfits the chance to control the activities of others by becoming involved in running it.

N IS FOR:
=====

NEURAL NETWORKS. An earnest Artificial Intelligence lecturer told me that with the help of these, it may be possible to build computers which behave exactly like human beings. Looking around the room, I wondered why he should want computers which get bored, forget what he tells them and drift off to sleep whilst he is speaking.

Some people are more skeptical, and ask whether it is possible for machines to think. Well, I certainly can, but I have my doubts about those who ask the question.

O IS FOR:
=====

OPTICAL CHARACTER RECOGNITION. By means of optical character recognition technology, it is hoped that we will soon be able to teach computers to read books. We can then move on to teaching them to watch TV, at which point we will finally have succeeded in dragging them down to our own intellectual level. It seems likely that TV companies will build rooms of minimally intelligent computers, each with a TV screen, in order to improve their own ratings.

P IS FOR:
=====

PORTABLE COMPUTERS. Most so-called 'portable' computers are the size of a couple of shoeboxes glued together. The reason for this is that they are mostly IBM PC compatible, and most have a full-size keyboard. It is tacitly implied via cybercrud that these features are important. In fact, this is not the case as far as the majority of ordinary people is concerned, but it does ensure that said machines appeal to dorks in suits who think that buying a non-IBM-compatible mains cable is dangerously rebellious. The rest of us are quite content to laugh at them whilst we tap away at our slim pocket-sized computers.

Q IS FOR:
=====

QUERY-BASED SYSTEMS. A close relative of databases, most query-based systems have some extra code bolted onto the front so that you can ask them questions in English; such as, say, 'When is the next bus to Cherry Hinton?' The resulting conversation tends to be rather one-sided, with the computer's responses generally being of the form 'I don't understand the word 'bus'.' Sooner or later the query-based system gets replaced by a program which prints a timetable on the screen, thus demonstrating the amazingly circular nature of cybercrud technology.

R IS FOR:
=====

ROBOTS. These are getting more and more sophisticated, and able to perform ever more complex tasks. Hollywood movie robots are cybercrud at its most developed --- metal men who wander around looking gormless, or tiny boxes

'cal 9 1752' explained

with smiling faces who wheel themselves about the place making amusing beeping noises. This goes a long way towards explaining why, amongst computer hackers, the most popular robot of recent years is the law enforcement robot at the start of ``RoboCop,' ' whose most admirable and enviable feature was the ability to blow away Americans with heavy weaponry.

S IS FOR:

=====

SOFTWARE. The term software means `instructions executed by a computer.' This hasn't stopped manufacturers of video movies from calling their product ``software'', in an attempt to make it sound glamorous and hi-tech, and above all to justify their hiked-up prices. What better way to make your product seem exciting than by insinuating that it involves computers? In fact the video manufacturers aren't doing anything that can't be done by anyone with two suitable video recorders and the proper cable, and we all know it. Nice try, though.

T IS FOR:

=====

TAPES. In the bad old days of the ZX-81, you would diligently go through a complicated procedure called the `loading ritual.' This involved carefully cleaning the heads of your cassette player, winding the cassette tape to the right place, connecting everything together and switching it all on, telling the computer to LOAD, and pressing play on the tape. The '81 would then play you something which looked like a monochromatic rock video animated by Jackson Pollock, at the end of which the RAM pack would wobble and you'd have to do the whole thing again.

The ZX-81 was a wonderful device, because it enabled people to find out how ghastly computers could be built to be, without having to spend large amounts of money on IBM kit in order to do so.

U IS FOR:

=====

UPPER-CASE LETTERS. The next time you get a letter addressed to ``MR J SMITH'' or whatever, stop to think for a moment. Why can't they print in lower-case? Are you going to trust a company which is careless enough to misplace two ``.' ' symbols from the name? Imagine if they misplace the ``.' ' in your bill. Of course, if asked about this, the company resorts to cybercrud, claiming that because the name and address is stored on computer it is necessary to put it in a special format. Like hell it is.

The author favours telling the institution concerned that unless they can get his name right, he will throw away their mail without opening it. He did this with his bank, and eventually they were forced to post the mail from their office back to their office so that it could be re-addressed by hand. Cruel, you may think --- but six weeks later, lo and behold, they had persuaded their computer that the author COULD have his name written correctly after all. How's that for progress?

V IS FOR:

=====

VODS. Video Operators' Distress Syndrome is the latest cybercrud ``Bogey Man'' phrase designed to give people the willies about computers and keep 'em in line. A little superficial research revealed that people sitting in front of computer screens were being bombarded with electromagnetic radiation. Over the last couple of years, hundreds of companies have begun

'cal 9 1752' explained

offering special screening panels to affix to the front of your computer monitor. With a word like ``radiation'' to bandy about, how could they go wrong? Some of these panels, it is claimed by the manufacturers, screen out 100% of the emitted electromagnetic radiation. Why this should be a good thing for a video display is not answered. Something else which goes unanswered is: if electricity is so dangerous to our health, why haven't we noticed at some point during the last fifty-odd years? And what's the point of screening your computer when every other electrical item in the room is giving off the same radiation?

W IS FOR:

=====

WORD-PROCESSING. This ridiculous term was coined by IBM; a delicate piece of cybercrud, it implies that what the computer is doing is difficult to do, whereas in fact it is really just text handling. The only thing which makes ``word processing'' difficult is the unmitigated ghastliness of most of the programs available to perform the task. It's no surprise that Wordstar was voted ``The most difficult video game to win'' in an American magazine's end-of-year awards. I suppose one COULD dedicate one's life to remembering that CTRL-META-K K CTRL-Q is ``move block to next buffer,'' but quite frankly what the hell's the point? It's computing for train-spotters. The rest of us find an editor which can be operated with only five or ten control-keys, and get the job done in half of the time.

X IS FOR:

=====

X-WINDOW. The X-window system is something we're going to be seeing a lot more of. Unfortunately. It is a system which was not merely designed by committee, but programmed by committee as well. The fact that it actually seems to work is nothing short of a miracle. X was designed to ``make no assumptions about the style of user-interface.'' What this actually means in English is that it makes no attempt whatsoever to impose any design sense or good taste on the programmers writing code for it. Some of the programs which result are truly an experience-and-a-half to use. Writing code for X is like trying to write a set of instructions on how to ride a bicycle, and about as effective. It is made slightly more tolerable if you use ``Widget Sets,'' which always provide almost what you wanted but not quite. X provides numerous informative error messages, all of the form ``Segmentation fault - core dumped.''

Y IS FOR:

=====

YABA. Computer programmers are legendary for their inability to speak or write coherent English. They also seem to have a peculiar affinity for useless jargon words. Since the author has so far avoided jargon, it is perhaps time to say that when your APL is FUBAR and giving GIGO, and the WYSIWYG WP on your VDU goes AWOL and dumps ASCII CCs to your LPT: port, you can comfort yourself by making up YABA -- Yet Another Bloody Acronym.

Z IS FOR:

=====

ZEN. Nothing to do with cybercrud or computers, but after staring at your C code for a couple of hours you suddenly begin to think that perhaps those koans made some sense after all. Mind you, after I've been staring at C code for a couple of hours absolutely anything seems to make sense by comparison.

'cal 9 1752' explained

- The attribution of this article was lost. If you know who wrote it, please tell me.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Definitive Null Command

Now: The Definitive Null Command

Bourne shell users, accustomed to creating or emptying a file with a command like:

```
>spot
```

or rewinding a tape with:

The Feet First Guide to Programming Languages

The Feet First Guide to Programming Languages

The proliferation of modern programming languages which seem to have stolen countless features from each other sometimes makes it difficult to remember which language you're using. This guide is offered as a public service to help programmers in such dilemmas.

C: You shoot yourself in the foot.

Assembler: You crash the OS and overwrite the root disk. The system administrator arrives and shoots you in the foot. After a moment of contemplation, the administrator shoots himself in the foot and then hops around the room rabidly shooting at everyone in sight.

APL: You hear a gunshot, and there's a hole in your foot, but you don't remember enough linear algebra to understand what the hell happened.

C++: You accidentally create a dozen instances of yourself and shoot them all in the foot. Providing emergency medical care is impossible since you can't tell which are bitwise copies and which are just pointing at others and saying, "that's me, over there."

Ada: If you are dumb enough to actually use this language, the United States Department of Defense will kidnap you, stand you up in front of a firing squad, and tell the soldiers, "Shoot at his feet."

MODULA-2: After realizing that you can't actually accomplish anything in the language, you shoot yourself in the head.

Pascal: Same as Modula-2, except the bullets are the wrong type and won't pass through the barrel. The gun explodes.

sh,csh,etc.: You can't remember the syntax for anything, so you spend five hours reading man pages before giving up. You then shoot the computer and switch to C.

Smalltalk: You spend so much time playing with the graphics and windowing system that your boss shoots you in the foot, takes away your workstation, and makes you develop in COBOL on a character terminal.

Smalltalk: You send the message shoot to gun, with selectors bullet and foot. A window pops up saying Gunpowder doesNotUnderstand: spark. After several hours fruitlessly spent browsing the methods in Trigger, FiringPin and IdealGas, you create ShotFoot, a subclass of Foot with a new instance variable bullet hole.

FORTRAN: You shoot yourself in each toe, iteratively, until you run out of toes, then you read in the next foot and repeat. If you run out of bullets, you continue anyway because you have no exception-processing ability.

ALGOL: You shoot yourself in the foot with a musket. The musket is esthetically fascinating, and the wound baffles the adolescent medic in the emergency room.

COBOL: USEing a COLT45 HANDGUN, AIM gun at LEG.FOOT, THEN place ARM.HAND.FINGER on HANDGUN.TRIGGER, and SQUEEZE. THEN return HANDGUN

'cal 9 1752' explained

to HOLSTER. Check whether shoelace needs to be retied.

BASIC: Shoot self in foot with water pistol. On big systems, continue until entire lower body is waterlogged.

PL/I: You consume all available system resources, including all the offline bullets. The Data Processing & Payroll Department doubles its size, triples its budget, acquires four new mainframes, and drops the original one on your foot.

SNOBOL: You grab your foot with your hand, then rewrite your hand to be a bullet. The act of shooting the original foot then changes your hand/bullet into yet another foot (a left foot).

LISP: You shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds...

SCHEME: You shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds... ..but none of the other appendages are aware of this happening.

English: You put your foot in your mouth, then bite it off.

POP11: Your boss has never heard of it and wants you to use one of C, C++, Lisp, or Prolog, so you push his foot onto the stack and shoot it.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Good Old Days

henry@zoo.toronto.edu (Henry Spencer) writes:

>In article brad@clarinet.com (Brad Templeton) writes:
>>(In the early days, Henry used to get his feed by manually dialing decvax
>>on his phone, and hooking up his 300 baud modem. [To be poetic, I imagine
>>a rotary phone. Was it, Henry?])

>Sure was. Touchtone didn't exist at U of T then.

>We were awfully happy when we got an autodialing 1200-baud modem.

Oooh, you had a modem! Luxury! I used to get a news feed transmitted in morse code by an epileptic chicken through a 150 mile piece of string with a tin can at each end. I had to get up at 2 AM every day, strap on the tin can, and type the articles into a TS-1000, using an oscilloscope as a display. If one of our users wanted to post, I had to write it on an 8-inch floppy using a refrigerator magnet. Kids got it easy these days, they do.

--

Kenneth Herron
University of Kentucky
Department of Mathematics

kherron@ms.uky.edu
+1 606 257 2975

"You don't carve 'ARGH,' you just say it!" "Perhaps he was dictating?"

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Gospel According To St. Hacker

THE GOSPEL ACCORDING TO ST. HACKER

CHAPTER 1

From: WJN at MIT-DMS (Wayne J. Noss)

I'll be an X ("ex-lab member"), which has status somewhere between that of a lab member and a tourist. I hope you don't think that I feel so great and wonderful that I would never stoop to being a tourist. Far from it! I got a tourist account on MIT-EECS, so I sortof know what it's like.

You see, once there was a certain Real User on a certain system that had lots and lots of lowly tourists. Some of the tourists tried to make the system a better place, but they weren't always understood or appreciated too much. Unfortunately, some other tourists were rather selfish and gave all tourists a bad name. They wiped out the work done by other tourists, and even .GUNned them sometimes. Of course, all this severely disrupted the work of the Real Users, so something just had to be done. All tourists might have been kicked off the system permanently, but one certain Real User decided to go to their rescue. So He became a tourist. In fact, a tourist of the very lowliest sort. He didn't have any friends with directories where He could put His hacks, and tourists with more status hogged the system so there wasn't even room for Him to be on sometimes. In fact, it was very crowded when He first logged in, and all He could get was a HACTRN--no other jobs or file space or directory slots. He knew the system pretty well, and helped the other tourists (those who noticed Him) to hack better. But then, the bad tourists found out about Him and got jealous because He was making ordinary randoms into better hackers than they were. They hated Him and plotted how they might get Him off the system permanently. This certain Real User who became a tourist meanwhile was explaining to those who listened to Him who He really was and why He and the Administrator of the system allowed all this mess to take place. He illustrated His points with common stories that His friends understood but that His enemies couldn't pin Him down on to complain to USER-A about Him (of course, they complained anyway, but they didn't have any good evidence against Him). He told them that the time would come when the Administrator of the system would kick all the obnoxious characters off the system and would give all the good tourists accounts on a much bigger and better system where they could do much bigger hacks and where everyone would get along as good friends. But in the meantime the system was allowing the tourists to choose between being obnoxious and being good, so that the Administration would know who to kick off and who to keep around for the better system. Well, you guessed it. The bad guys finally not only .GUNned this perfect example of the Ultimate Tourist; they even flushed his password and everything so He could never, never log in again. SIGH. But that was not the end; He was a Real User, so He came back anyway. The selfish users tried to explain this all away, saying that His friends must have hacked the console log and a bunch of other things to make it look like He came back, but that He didn't, really. Well, He doesn't log in as a tourist anymore, and next time He does, it will be the end of all the bad goings-on on the system. Because no one sees Him logged in anymore, most think that He just got flushed and that was the end; but they have trouble explaining how all over the system, all sorts of lowly tourists who hear about Him--even bad guys--anyone who decides to believe the things He used to say (some of His friends kept track of His teachings regarding how to hack the right way), actually gets changed into a good hacker who makes

'cal 9 1752' explained

winning hacks and is kind to all the other tourists. It almost seems as though He has multiplied, and there are many of Him around now.

Well, anyway, I am one of those hackers who believes this certain Tourist who said He was a Real User. And, if a Real Real User like Him could go all the way to the very bottom end of the scale and become the lowliest of tourists and eventually get blamed for everyone else's faults and get flushed from the system, who am I that I should expect or even try for anything better?

So anyway, I'll be around, as an X, a tourist, a doormat, or whatever.

CHAPTER 2

From: KREEN at MIT-DMS (Brian J. Kreen)
[The parable of the good Tourist]

The system administrator was talking to some lab members, and they were raving with him about the following passage in the DEC manual:

Love your neighbor as yourself.
Love your brother as HE has loved you.

Well, one of the lab members asked, "But who is my neighbor?" So the administrator told them this parable.

There was once a DM lab member on the way to MAC but a group of Multicians came by and robbed him and hurt him. Soon, a ML lab member came by, but did not stop to help the DM member because he was from another machine, and he did not want to dirty himself with intra-MAC hackery. Later, a MC member came by, but he refused to help the DM member because the DM person hacked Muddle, and the MC member hacked LISP. The hours past, and the DM lab member was near death.

A lowly tourist was randomly walking by, and saw the lab member. He stopped to help him. The tourist brought him to the MAC hospital, and paid the lab members bill, and while the lab member was recovering, the tourist gave the DM person his own small terminal.

The administrator asked the lab member, so, you tell me who is your neighbor. The lab member replied "The one who helped the DM person." because the lab member could not bring himself to say directly that the tourist was his neighbor.

The administrator then said "I am giving you a new UUU to install. 'Love your neighbor as HE loves you'. Meaning that you should love everyone, INCLUDING tourists as HE loves you. The lab member walked away saddened, the administrator had stopped his war against tourism again.

Ha, ha, ha. Take me back to [the alphabetic list] [the date-ordered list].



The Great Quux Poem Collection

Random Poetry and Other Losses

printed April 17, 1979

Random Poetry and Other Losses

The Great Quux

(C) Copyright 1978 by Guy L. Steele Jr. All rights reserved.

Any similarity of anything in this document to anything in the real world, except for a satiric purpose, is purely coincidental.

'cal 9 1752' explained

The Great Quux
Random Poetry and Other Losses

page
printed April 17, 1979

BBN Superlisp

[to be sung to the tune of
Jesus Christ Superstar]

Every time I look at you I don't understand
Why you think "Do What I Mean" is so cool and grand;
You'd have managed better if you'd thought it through,
Why'd you pick such an awkward way your bugs to undo?
Your hairy feature will not be the last revolution,
It's clear "Mean What I Do" is the ultimate solution!

Don't you get me wrong,
Don't you get me wrong,
Don't you get me wrong, now,
Don't you get me wrong,
I only want to hack,
I only want to hack,
I only want to hack,
I only want to hack.

BBN! BBN! Some people think you're the living end!
BBN! BBN! Some people think you're the living end!
BBN! SuperLISP! Can "Do What I Mean" measure up to this?
BBN! SuperLISP! Can "Do What I Mean" measure up to this?

Tell us what you think about your friends at the top,
Who d'you think besides yourself's the pick of the crop?
Is LISP 1.5 where it's at? Is it where you are?
Does Stanford's LISP have features too or is that just PR?
Do you have the breakpoint scheme that MACLISP is known for,
Or is that just the kind of kludge the user's on his own for?

Don't you get me wrong,
Don't you get me wrong,
Don't you get me wrong, now,
Don't you get me wrong,
I only want to hack,
I only want to hack,
I only want to hack,
I only want to hack.

BBN! BBN! Some people think you're the living end!
BBN! BBN! Some people think you're the living end!
BBN! SuperLISP! Can "Do What I Mean" measure up to this?
BBN! SuperLISP! Can "Do What I Mean" measure up to this?

-- The Great Quux
(with apologies to
Rice and Webber)

(C) Copyright 1973 Guy L. Steele Jr. All rights reserved.

Both Ways, Now

[to be sung to the tune of
Both Sides, Now]

Decimal digits in a row,
Just set the dials and let 'er go.
The ENIAC was grossly slow --
 I used to code that way,
But then this Fortran came along;
I danced and sang a happy song:
So natural -- what could go wrong?
 I little knew, that day!
I've looked at Fortran both ways, now,
At II and IV, and still somehow,
It's rows of numbers I recall;
I really don't know Fortran at all.

Fortran IV is real good stuff,
But business hackers have it tough;
For them this Fortran's not enough --
 Then Cobol saved the day!
But now I sing a sad refrain;
This Cobol loss is no one's gain,
And writing programs is a pain
 (I get writer's cramp that way!)
I've looked at Cobol both ways, now,
I code in it, and still somehow,
It's FORMAT statements I recall;
I really don't know Cobol at all.

Cobol will for business do;
Accounts and payroll make it through
(And bills for zero dollars too --
 I get them every day!)
But those who hack symbolic frobs
Cannot make do with Cobol jobs,
And now I sing through anguished sobs,
 But Lisp is here to stay.
I've looked at Lisp code both ways, now,
At lambda forms, and still somehow,
It's Cobol statements I recall;
I really don't know Lisp at all.

-- The Great Quux
 (with apologies to
 Joni Mitchell)

(C) Copyright 1974 Guy L. Steele Jr. All rights reserved.

'cal 9 1752' explained

Random Poetry and Other Losses

printed April 17, 1979

A Time for DWIM

[to be sung to the tune of
A Time for Us
(theme song from Romeo and Juliet)]

A time for DWIM
There'll never be;
 No clever code
 This losing mode
Can UNDO for me.

This "golden hope"
(To be denied)
 Could never
Correctly fix the bugs my programs hide.

A way for bugs
There'll never be
To fix with generality.

So to this DWIM
Let's say farewell;
 The crocks therein
 Prove it can't win
And ring its knell:

Do What I Mean
Is just a ruse --
 It really
Means only: Fix How Teitelman doth Lose!

-- The Great Quux
 (with apologies to
 Rota, Kusik, and Snyder)

(C) Copyright 1973 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 4
printed April 17, 1979

'cal 9 1752' explained

The Sound of FORTRAN

[to be sung to the tune of
The Sound of Music]

My programming day has come to an end, I know,
But one minor bug still restrains me, though,
So back to me desk I stumble,
More coffee I pour in my mug,
And I drink, and I think, and I program
Just one more hack, just one more hairy kludge
To remove that bug.

Machines are alive with the sound of FORTRAN,
With numbers they've crunched for a thousand hours;
They add and subtract to the sound of FORTRAN,
And raise fractions to unheard of powers.

My code's full of REAL statements, INTEGER and COMPLEX too,
duplicated thrice oe'r,
And so intermixed with the WRITES and READs
to cause errors galore;
Arrays are declared of dimension six, but indexed minus two;
Computed GO TOs are last in the range of a DO!

I now recompile my corrected programs;
I know I will get what compiled before --
My code will be blessed with the sound of FORTRAN,
And I'll lose once more.

-- The Great Quux
(with apologies to
Rodgers and Hammerstein)

(C) Copyright 1973 Guy L. Steele Jr. All rights reserved.

'cal 9 1752' explained

The HACTRN

Once before a console dreary, while I programmed, weak and weary,
Over many a curious program which did TECO's buffer fill, --
While I pondered, nearly sleeping, suddenly there came a feeeping,
As of something gently beeping, beeping on my console's bell.
" 'Tis my DDT," I muttered, "feeeping on my console's bell:
Once it feeped, and now is still."

Ah, distinctly I remember that dark night in bleak December,
And each separate glowing symbol danced before me, bright and chill.
Eagerly I wished the morrow; vainly I had sought to borrow
From my HACTRN aid for sorrow -- sorrow for the bugs which fill --
For the strange unknown and nameless bugs which ever all my programs f
Bugs which now I searched for still.

And the coughing, whirring, gritty fan I heard inside my TTY
Made me with fantastic terrors never known before to thrill;
So that now, to still the beating of my heart I stood repeating,
" 'Tis some interrupt entreating DDT to signal me --
Some strange interrupt entreating DDT to signal me --
Its importance surely nil."

Presently my soul grew stronger: hesitating then no longer
I decided that I would respond to this strange program's call;
TECO, which I then attended, to my soul more strength extended;
With ^Z I ascended, going to my DDT --
V I typed, and answered soon my DDT --
TECO there, and that was all!

Dumbly at my console peering, as I sat there, wondering, fearing,
Doubting now that any interrupt was ever there to call;
But the silence was unbroken, and my HACTRN gave no token,
And the only sound there spoken from my TTY's whirring fan --
The low and rough and distant sound came from my TTY's whirring fan --
TECO there, and that was all.

Back into my TECO going, with my pounding heart now slowing,
Soon again I heard a feeeping, somewhat louder than before.
"Surely," said I, "surely this is some strange bug of RMS's
Which an interrupt professes, though I have no other job;
Let me then ask DDT if it thinks there's another job --
'Tis a bug, and nothing more!"

Again I went up to my HACTRN while cold shivers up my back ran
V I typed, my jobs now once more to display.
Only TECO was there listed; though my trembling heart resisted
Yet I willed my hand, insisted, J to quickly type --
To answer this bold query DDT did hesitantly type
A ghostly "FOOBAR J".

'cal 9 1752' explained

From V protected, now, this phantom job, selected
Gave no clue to why it had invoked that former beeping shrill.
"Though," I said, "you're no inferior, I shall act as your superior
And examine your interior, this strange matter to explore."
Then I typed a 0/ this matter further to explore --
 Quoth the HACTRN, ":KILL".

Much I worried -- this outrageous bug might prove to be contagious,
Though thus far it had not seemed to do my TECO any ill:
For we cannot help concurring such a bug would cause a stirring,
Feeeping on a console whirring, disappearing then from sight --
An evanescent mystery subjob disappearing then from sight
 With no clue but ":KILL"!

But my HACTRN, swapping, running, gave no further sign of cunning
By this unknown phantom, which was in a thirty second sleep;
None of this I comprehended; to my TECO I descended,
And in terror I pretended that the bug had gone away --
I pretended that for good the mystery bug had gone away --
 When my console gave a feep.

Now I quickly, hoping, praying, started up a PEEK displaying
All the the jobs and subjobs there which did the system fill:
What I found was quite unpleasant, for there was no FOOBAR present:
Only TECO was there present, underneath my DDT;
I quit the PEEK, and "FOOBAR J" typed out my DDT --
 Then quoth the HACTRN, ":KILL".

But -- this FOOBAR now beguiling all my sad soul into smiling --
I tightly grinned, determined that this glitch should cause nobody ill
Now, into my armchair sinking, I betook myself to linking
Fancy unto fancy, thinking why this unknown phantom job --
Why this grim, ungainly, ghastly, gaunt, and unknown phantom job
 Feeped and did a ":KILL".

This I sat engaged in guessing, but conceived no thought expressing
How a phantom job could sound those strange and ghostly beeps;
This and more I sat divining, with my head at ease reclining,
With the symbols coldly shining at me from the CRT,
With the bright, sharp symbols coldly shining on the CRT --
 Which suddenly gave seven feeeps!

Then methought the air grew denser, filled with clouds which grew immenser,
As when under darkened daylight thick and stormy weather brews;
With some bit of hesitation stemming from my trepidation
Again I typed that incantation finding out how much I'd lose --
 V I typed again to find how much I'd lose --
 TECO there, and seven FOOs!

"Job!" said I, "with ghostly manner! -- subjob still, if LISP or PLANNER!
Whether accident, or feeeping as another hacker wills!
Tell me now why I am losing, why my HACTRN you're abusing,
Which no doubt is of your choosing: echo truly on my screen!"
Then DDT as if in answer echoed quickly on my screen,
 Typing seven ":KILLs".

The Great Quux
Random Poetry and Other Losses

page 7
printed April 17, 1979

"Job!" said I, "with ghostly manner! -- subjob still, if LISP or PLANNER!"

'cal 9 1752' explained

By the ITS above us which the DSKDMP doth fulfill,
I shall be the system's saviour: I shall mend your crude behaviour,
I shall halt your strange behaviour, and thee from the system lock!"
Madly, wildly laughing I made DDT invoke a LOCK,
And quickly typed thereat -- "5KILL"!

"Be this now our sign of parting, phantom job!" I shrieked, upstarting
As my HACTRN now informed me ITS was going down in 5:00.

"You have run your last instruction and performed your final function!
But, refuting this deduction HACTRN then my TTY grabbed --
To type out yet another message HACTRN now my TTY grabbed --
Quoth the HACTRN, "ITS REVIVED!"

And the FOOBAR, never sleeping, still is beeping, still is beeping
On the glaring console out from which I cannot even log!
And other happenings yet stranger indicate inherent danger
When bugs too easily derange or mung the programs of machines;
When programs too "intelligent" start taking over the machines:
Is this the end of AutoProg?

-- The Great Quux
(with apologies to
Edgar Allan Poe)

Notes for those not familiar with the terms in this poem:

TTY ("titty") = any terminal, not necessarily a teletype (in this case, a CRT); in particular, a terminal associated with and in control of a job tree (see "DDT" below). The terminal may be passed up and down the job tree; at any point in time only one job in the tree may use the tree's TTY. When ^Z is typed on the TTY, the system intervenes, stopping the job which has the TTY, and interrupts that job's superior in the tree, which may then grab the TTY from the inferior job.

DDT ("dee dee tee") = HACTRN ("hack-tran") = top level debugging and job controlling procedure, capable of controlling up to eight simultaneous jobs (which may themselves be DDTs!) and performing other miscellaneous functions. HACTRN specifically denotes a DDT at the top of a job tree, while DDT is the more general term. The two terms refer to the same job in the poem, and are thus treated as synonymous. Note that DDT requires its subjobs to have unique names for obvious reasons; hence the concern over seven jobs all named FOO.

The Great Quux
Random Poetry and Other Losses

page 8
printed April 17, 1979

PEEK = a program similar to the SYSTAT of certain PDP-10 monitor systems of dubious quality. PEEK is actually much more

'cal 9 1752' explained

versatile, giving information in any of some dozen modes, such as a job status, DECTape status, Arpanet sockets, terminal status, and scheduler variables and statistics. It also has provisions for maintaining a continuously updated display on a CRT, and for line printer usage.

TECO ("teeko") = text editor and corrector (that is, the good version of several versions of TECO which are floating around).

:KILL ("colon kill") = message typed out by DDT whenever it kills a subjob. Note that subjobs, if running, may request DDT to kill themselves. If the job does not have the TTY when it makes such a request, DDT merely rings the TTY's bell (which on the CRT in the poem above is a particularly obnoxious flavor of "beep"), and prints nothing until you ascend to DDT, and perhaps type J (see below).

LOCK = utility program, which interprets the particular command "nKILL" to mean "please bring the time-sharing system down in n minutes" (where it is required that n > 0). The system will then go down at the prescribed time unless the request is countermanded with a "REVIVE" request.

ITS = Incompatible Timesharing System, the good timesharing system for the PDP-10.

DSKDMP ("disk dump") = program used to, among other things, bootstrap ITS into a running state.

= "altmode"; read it as such to preserve the meter.

V = command to DDT, requesting it to print out the names of all its subjobs.

J = command to DDT, asking that it select the job which has requested attention so that it may be dealt with. DDT responds "jobname J" so that you will know which job it was.

^Z ("control zee") = command to ITS to stop the job which currently has the TTY, and interrupt the next higher job in the job tree. Ordinarily this has the effect of returning to DDT.

The Great Quux
Random Poetry and Other Losses

page 9
printed April 17, 1979

0/ ("zero slash") = command to DDT, asking it to print out the contents of location zero of the selected subjob. This operation is theoretically transparent to the subjob itself.

'cal 9 1752' explained

RMS = Richard M. Stallman, who does an admirable job of keeping DDT,
as well as many other programs, relatively bug-free.

(C) Copyright 1973, 1974 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 10
printed April 17, 1979

A Collection of Nursery Rhymes and Other Random Losses
by The Great Quux, RMS, and Anonymous

'cal 9 1752' explained

A monitor system named DOS
Became superseded by OS.
It's not hard to guess
Why the consequent mess
Caused programmers all to scream "S.O.S."

-- TGQ May 1973

IBM had a PL/I,
Its syntax worse than JOSS;
And everywhere this language went,
It was a total loss.

-- TGQ June 1973

System/3! System/3!
See how it runs! See how it runs!
Its monitor loses so totally!
It runs all its programs in RPG!
It's made by our favorite monopoly!
System/3!

-- TGQ October 1973

RRR of x
always equals 1.
R is dimension, RR, rank.
APL is fun!

-- RMS December 1973

```
RRRboat
  gently stream
  4 7R"merrily"
  life
value error
  life
)
```

-- Anon January 1974

Hey, diddle, diddle the overflow pdl
To get a little more stack;
If that's not enough then you lose it all
And have to pop all the way back.

-- TGQ January 1974

(C) Copyright 1974 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 11
printed April 17, 1979

It may surprise some people to know that the PDP-10 instruction set, far from being innovative, was developed strictly along classical lines. Many of the operation codes, for example, come from the Latin. Two fifth-declension nouns (declinable only in the

'cal 9 1752' explained

singular) are as follows:

Nominative:	moves	hrres
Genitive:	movei	hrrei
Dative:	movei	hrrei
Accusative:	movem	hrrem
Ablative:	move	hrre

There are, of course, also hlres, hrles, and hllres. These all derive from the corresponding verbs hrro, hrlo, etc. (second conjugation, semi-deponent.) Thus when playing chess with MacHack one might say, "movem hrro," that is, "I am hurrying a move."

Similarly, the code jffo ("i am jffing") comes from a defective Latin verb dating from an era when ancient computers marked time by periodically pushing counters onto a wire. The number of empty spaces left on the wire could easily be counted by an instruction very like jffo, from which could be calculated the number of "jiffies" (clock ticks; 1 jiffy = 40 seconds) which had passed; hence the name.

Teutonic languages have also lent several words to the PDP-10: rot ("red") derives from the pieces of decayed redwood traditionally used as temporary markers in older German manual shift registers; while ash was commonly the result when an older German got disgusted and threw his shift register in the fire.

--- The Great Quux
(with apologies to no one)

(C) Copyright 1973 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 12
printed April 17, 1979

PLIate's Dream

[to be sung to the tune of
Pilate's Dream
from Jesus Christ Superstar]

'cal 9 1752' explained

I dreamed I was a brand new language,
The ultimate in speed;
I handled strings as fast as RPG,
And twice as easily.

I crunched numbers like COBOL,
Trees like APL,
And FORTRAN loaned its FORMATS and GO TOs,
The cause of many screws.

And then a man said, "Now we'll write a monitor,
With Multics what it's for.
Our project is begun;
We'll code in PL/I."

Then I saw thousands of coders
Searching for their bugs,
And then I heard them mentioning my name
And leaving me the blame.

-- The Great Quux
(with apologies to
Rice and Webber)

(C) Copyright 1973 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 13
printed April 17, 1979

As I was passing Project MAC,
I met a Quux with seven hacks.
Every hack had seven bugs;
Every bug had seven manifestations;
Every manifestation had seven symptoms.
Symptoms, manifestations, bugs, and hacks,
How many losses at Project MAC?

'cal 9 1752' explained

-- TGO April 27, 1974

(c) Copyright 1974 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 14
printed April 17, 1979

Reclaimer, spare that tree!
Take not a single bit!
It used to point to me,
Now I'm protecting it.
It was the reader's CONS
That made it, paired by dot;
Now, GC, for the nonce,
Thou shalt reclaim it not.

'cal 9 1752' explained

That old familiar tree,
Whose CDRs and whose CARs
Are spread, o'er memory --
And wouldst thou it unpurse?
GC, cease and desist!
In it no freelist store;
Oh spare that moby list
Now pointing throughout core!

It was my parent tree
When it was circular;
It pointed then to me:
I was its CADADR.
My CDR was a list,
My CAR a dotted pair --
That tree will sore be missed
If it remains not there.

And now I to thee point,
A saving root, old friend!
Thou shalt remain disjoint
From freelists to the end.
Old tree! The sweep still brave!
And, GC, mark this well:
While I exist to save,
Thou shan't reclaim one cell.

-- The Great Quux
(with apologies to
George Pope Morris)

(C) Copyright 1973 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 15
printed April 17, 1979

I Could Have Tooled All Night

[to be sung to the tune of
I Could Have Danced All Night
from My Fair Lady]

Tool! Tool! I feel like such a fool!
All term I goofed off; I can't catch up now!

'cal 9 1752' explained

Sleep! Sleep! I've got to get some sleep!
Tooling wouldn't help me anyhow!

I could have tooled all night,
I could have tooled all night,
and still have tooled some more;
I could have been absurd,
Learned all my Latin verbs,
It wouldn't raise my score.
I can't remember all those theorems,
And all those facts from my mind flee --
I only know exams,
Are why one usually crams,
But tooling never could help me!

I could have tooled all night,
I could have tooled all night,
And memorized each book;
I only now regret,
My sections never met,
And lectures I forsook.
I cross my fingers now in terror,
I only hope some luck's with me --
But had I tooled or not,
I'd still be on the spot,
My goofing off deserves the E!

-- The Great Quux
(with apologies to
Lerner and Loewe)

(C) Copyright 1974 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 16
printed April 17, 1979

I think that I shall never see
A matrix lovely as a tree.
Trees are fifty times as fun
As structures a la PL/I
(Which Dijkstra claims are too baroque).
And SNOBOL's strings just can't compare
With all the leaves a tree may bear.
And COMIT strings are just a joke.
Vectors, tuples too, are nice,
But haven't the impressive flair

Of trees to which a LISP is heir.
A LISPer's life is paradise!

Many people think that JOSS
And others, too, are strictly boss;
And there are many BASIC fans
Who think their favorite language spans
All that would a user please.
Compared to LISP they're all a loss,
For none of them gives all the ease
With which a LISP builds moby trees.

RPG is just a nurd
(As you no doubt have often heard);
The record layouts are absurd,
And numbers packed in decimal form
Will never fit a base-two word
Without a veritable storm
Of gross conversions fro and to
With them arithmetic to do.
And one must allocate the field
Correct arithmetic to yield,
And decimal places represent
Truncation loss to circumvent:
Thus RPG is second-rate.
In LISP one needn't allocate
(That boon alone is heaven-sent!)
The scheme is sheer simplicity:
A number's just another tree.
When numbers threaten overflow
LISP makes the number tree to grow,
Extending its significance
With classic treelike elegance.
A LISP can generate reports,
Create a file, do chains and sorts;
But one thing you will never see
Is moby trees in RPG.

One thing the average language lacks
Is programmed use of push-down stacks.
But LISP provides this feature free:
A stack - you guessed it - is a tree.
An empty stack is simply NIL.
In order, then, the stack to fill

The Great Quux
Random Poetry and Other Losses

page 17
printed April 17, 1979

A CONS will push things on the top;
To empty it, a CDR will
Behave exactly like a pop.
A simple CAR will get you back
The last thing you pushed on the stack;
An empty stack's detectable
By testing with the function NULL.
Thus even should a LISPer lose
With PROGs and GOs, RETURNS and DOs,
He need his mind not overtax
To implement recursive hacks:

'cal 9 1752' explained

He'll utilize this clever ruse
Of using trees as moby stacks.
Some claim this method is too slow
Because it uses CONS so much
And thus requires the GC touch;
It has one big advantage, though:
You needn't fear for overflow.
Since LISP allows its trees to grow,
Stacks can to any limits go.

COBOL input is a shame:
The implementors play a game
That no two versions are the same.
And rocky is the FORTRAN road
One's alpha input to decode:
The FORMAT statement is to blame,
But on the user falls the load.
And FOCAL input's just a farce;
But all LISP input comes pre-parsed!
(The input reader gets its fame
By getting storage for each node
From lists of free words scattered sparse.
It parses all the input strings
With aid of mystic mutterings;
From dots and strange parentheses,
From zeros, sevens, A's and Z's,
Constructs, with magic reckonings,
The pointers needed for its trees.
It builds the trees with complex code
With rubout processing bestowed;
When typing errors do forebode
The rubout makes recovery tame,
And losers then will oft exclaim
Their sanity to LISP is owed -
To help these losers is LISP's aim.)

The flow-control of APL
And OS data sets as well
Are best described as tortured hell.
For LISPer's everything's a breeze;
They neatly output all their trees
With format-free parentheses
And see their program logic best

The Great Quux
Random Poetry and Other Losses

page 18
printed April 17, 1979

By how their lovely parens nest.
While others are by GOs possessed,
And WHILE-DO, CASE, and all the rest,
The LISping hackers will prefer
With COND their programs to invest
And let their functions all recur
When searching trees in maddened quest.

Expanding records of fixed size
Will quickly programs paralyze.
Though ISAM claims to be so wise
In allocating overflow,

'cal 9 1752' explained

Its data handling is too slow
And finding it takes many tries.
But any fool can plainly see
Inherent flexibility
In data structured as a tree.
When all their efforts have gone sour
To swell fixed records, losers glower.
But list reclaimers hour by hour
By setting all the garbage free
Yield CONSequent capacity:
Thus trees indefinitely flower.
(And trees run on atomic power!)

To men of sensibility
The lesson here is plain to see:
Arrays are used by clods like me,
But only LISP can make a tree.

- The Great Quux
(with apologies to
Joyce Kilmer)

(C) Copyright 1973 Guy L. Steele Jr.
All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 19
printed April 17, 1979

Alphabetical Order

MACLISP Ben PDP-6 (may its core increase!)
Awoke one jiffy from a deep dream of trees
And saw, by parsing out a TTY: .FILE. (DIR)
Scanning all the jobs, within the system here,
A PEEK, displaying on a CRT of gold;
A low USRTIM made the MACLISP bold:
And to the PEEK by CLI: a message sent,
"What displayest thou?" The message to it quickly went,
And the PEEK through CLA: a message did transduce,
And answer'd "The names of programs of practical use."

'cal 9 1752' explained

"And is mine one?" sent MACLISP. "Nay, not so,"
Replied the PEEK. MACLISP ran more slow
But swapping still. "I pray thee then," it sent,
"Display me as one that helps to systems implement."
The PEEK :KILLED and vanished. The next time
It ran again, in a display mode sublime,
It showed the names the Grim File Reaper had blessed,
And lo! MACLISP's name was thirty-seventh.

-- The Great Quux
(with apologies to
Leigh Hunt)

(C) Copyright 1973 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 20
printed April 17, 1979

Lambda Bound

[to be sung to the tune of
Homeward Bound]

I'm just a little value cell,
And I play my special role so well --
Hmmm --
Serving as a global switch
To predicate some system glitch;
But some strange value -- who knows which? --
Could cause me functions to bewitch!
Lambda bound!

I wish I was
 Lambda bound!
Bound, so no SETQ's get me;
Bound, so quits will reset me;
Bound, where I can forget my
 Top-level value.

It's hard to catch those system screws:
'Most any value causes me to lose --
 Hmmm --
Each atom looks the same to me,
Whose interned name I cannot see,
And every NIL and every T
Reminds me that I long to be
 Lambda bound!

I wish I was
 Lambda bound!
Bound, so no SETQ's get me;
Bound, so quits will reset me;
Bound, where I can forget my
 Top-level value.

The Great Quux
Random Poetry and Other Losses

page 21
printed April 17, 1979

Next time I'll have a MAR break set
And try to catch each clobber threat --
 Hmmm, mmmm --
The next covert attempt to mung
Will cause the MAR break to be sprung,
But then the poor LISP will be hung
Because I'm not as I have sung:
 Lambda bound!

I wish I was
 Lambda bound!
Bound, so no SETQ's get me;
Bound, so quits will reset me;
Bound, where I can forget my
 Top-level value.

'cal 9 1752' explained

-- The Great Quux
(with apologies to
Paul Simon)

(C) Copyright 1974 Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 22
printed April 17, 1979

Song of the Certified Data Processor

[to be sung to the tune of
When I Was a Lad
from H.M.S. Pinafore]

When I was a lad I served a term
As office boy to a computing firm.
I polished the handle of the big front door
And swept up all the card chips from the keypunch floor.
 He swept up all the card chips from the keypunch floor.
I swept that chad so carefuller
That now I am officially a CDP.
 He swept that chad so carefuller
 That now he is officially a CDP.

'cal 9 1752' explained

My office job was a heavy load,
So I went to night school and learned to code.
I was soon coding payroll in RPG
And compiled all my programs on a System/3.
 He compiled all his programs on a System/3.
I compiled my code so gay and free
That now I am officially a CDP.
 He compiled his code so gay and free
 That now he is officially a CDP.

I wrote efficient code each day,
But I missed the benefits of higher pay.
I asked for a raise, but my boss said, "See,
Youse ain't good enuf because youse ain't a CDP."
 "He ain't good enuf because he ain't a CDP."
So I vowed that someday I would see
Myself become officially a CDP.
 So he vowed that someday he would see
 Himself become officially a CDP.

For nineteen weeks I worked to cram
All the textbooks for the CDP exam.
Then I took the exam and was shocked to see
That the questions didn't seem to mean a thing to me.
 All the questions didn't seem to mean a thing to him.
So I wrote down some answers randomly,
But I gave up all my hopes to be a CDP.
 So he wrote down some answers randomly,
 But he gave up all his hopes to be a CDP.

The Great Quux
Random Poetry and Other Losses

page 23
printed April 17, 1979

Well, those random answers worked out fine;
They scored my results at the top of the line.
Now I am a consultant here,
And I make at least a hundred thousand bucks each year.
 And he makes at least a hundred thousand bucks each year.
But I only command such a salary
Because I am officially a CDP.
 But he only commands such a salary
 Because he is officially a CDP.

Now, office boys, whoever you may be,
If you want to rise to the top of the tree,
Just go and take the CDP exam,
And no matter what you answer they won't give a D--n!
 And no matter what you answer they won't give a D--n!
Just answer it all as random as you please
And you will all officially be CDPs.

'cal 9 1752' explained

Just answer it all as random as you please
And you will all officially be CDPs.

-- The Great Quux
(with apologies to
Gilbert and Sullivan)

(C) Copyright 1974 by Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 24
printed April 17, 1979

A pile of stuff written in April, 1976

April 1, 1976

[1] Notes taken in 6.015 (Signals and Systems)

On Matched Filters:

The ancients used potions, their love schemes to hatch;
They knew the right philtre encouraged a match.
When a lad made a wink at a comely young lass,
The philtre she used was designed for a pass.
The signals to use were determined by philtre;
The two peaked as one, for the lad could not jilt her.
The two thus entwined, so that noise did not bother
(Excepting that made by the groom's in-law mother...)

'cal 9 1752' explained

On Time Division Multiplexing:

To get the most use from your broadcasting medium:
Activity is, strangely, relieved by the TDM.

[2] Notes taken in 6.893 (Machine Perception and Manipulation)

"... and so the line connecting the points Ga and Gb in gradient space, which correspond to the planes A and B in image space, is the set of points representing positions of a plane see-sawing around the line of intersection between A and B..."

Now I see; then I saw;
The planes of a cube have a linear law.
The endpoints of lines in the gradient space
Show where the see-sawing planes fall into place.

Macrakis, sitting beside me, half-asleep: "COFFEE!"

Caffeine doesn't help me composing this verse.
It only awakes me; my thoughts all disperse.
One thinks better dozing, collapsed in a heap;
Why else are most students in classes asleep?

"... the lines in gradient space are perpendicular to the lines in image space. This doesn't provide enough constraints, however. Additional equations may be derived from the intensity information. One can get one or more solutions for a trihedral vertex. If the vertex has more than three planes, then there are more constraints than necessary, and one may have to resort to least squares..."

The Great Quux
Random Poetry and Other Losses

page 25
printed April 17, 1979

Alone, the geometry isn't enough:
You also require intensity stuff.
We get enough data if points are tri-planed,
While four leave the gradients over-constrained.

[3] Some Higgledy-Piggledies

For those not familiar with this verse form: A Higgledy-Piggledy has eight lines. All but the fourth and eighth lines are double dactyls, while the fourth and eighth are a dactyl plus an upbeat. The first line is nonsense; the second, a name; the sixth must be a single six-syllable word; and the fourth and eighth must rhyme. It is permissible for the first line to make sense, or for the fifth line instead of the sixth to be all one word. If too many rules are bent, the verse is just called a double-dactyl verse, rather than a Higgledy-Piggledy.

[a] If you write (DO ((X NIL X) ...) ...) instead of the

'cal 9 1752' explained

usual (DO ((X NIL) ...) ...), i.e. you explicitly step
a variable by its own value, NCOMPLR (the MacLISP compiler)
may get confused, and in any case generates inessential code.

Higgledy piggledy,
Project MAC NCOMPLR
Still is committing its
DO-stepping sins:
Though it may seem somewhat
Counterintuitive,
FOO stepped by FOO screws; de-
Faulting it wins.

[b] You can't do EQ on numbers and hope to win. Recall also
the famous test in which NCOMPLR produced better numerical
code than the DEC FORTRAN compiler (Fateman 1973).

Quaxity quuxity,
MacLISP and NCOMPLR
Have a convention that
Makes losers freak:
Integers allocate
Hyperefficiently,
Bettering FORTRAN, but
Screwing up EQ.

The Great Quux
Random Poetry and Other Losses

page 26
printed April 17, 1979

[c] As usual, taking a DWIM view of InterLISP.

Quaxity quuxity,
Teitelman's InterLISP
Has a DWIM feature that's
Really a screw;
MacLISP has evident
Superiority,
Letting its customer
Mean what he do.

April 2, 1976

Though NEWIO has been around for many months,
few people use it. I wonder why?

Quaxity quuxity,
TGQ's NEWIO
Has many functions that

No one can grok:
Documentation is
Incomprehensible;
GLS will explain it, but
Still it's a crock.

April 3, 1976

[a] NIL leads a double life in LISP.

Quaxity quuxity,
NIL the atomic one
Acts like a wavicle,
Seemingly twin:
Not just the inverse of
Verisimilitude;
Also the inverse of
Where lists begin.

[b] I am getting weary of bicentennial parading by industry.
"Bicentennial Minutes" now appear not only on television,
but on milk cartons and beer cans.

Quaxity quuxy, A-
Merican businessmen
Use bicentennial,
Profit to gain.
Hope by the time we reach
Quadricentennial
They'll not so plague me -- ad-
Vertisements pain!

The Great Quux
Random Poetry and Other Losses

page 27
printed April 17, 1979

[c] For SF fans: see Asimov's article on the "peace bomb"
(contained in his collection "Opus 100").

Quaxity quuxity,
Young Isaac Asimov
Strange thiotimeline's
Secret unfurled.
Cascaded series of
Endochronometers
Brought paradoxical
Floods on the world.

[d] More SF, this time in memory of James Blish:

Quaxity quuxity,
Blish used philosophy
When he was writing "The
Quincunx of Time."
Beeps of Dirac used to
Chronocommunicate
Fostered determinate

'cal 9 1752' explained

Free will sublime.

[e] DJL, computer science type undergraduate, has decided to go to MIT grad school in Materials Science.

Quaxity quuxity,
David J. Littleboy
(MIT graduate)
Makes his plans plain.
Leaving the study of
Data-technology,
He prefers seeking ma-
Terial gain.

[f] I once read that space has three dimensions because orbits aren't stable in 4-space.

I often have wondered in
What kind of orbit a
Planet proceeds in a
Tesseract space?
Multidimensional,
Hyperelliptical,
Dizzying spacemen in
Trans-solar chase.

The Great Quux
Random Poetry and Other Losses

page 28
printed April 17, 1979

[g] Finally, let's hear it for quantum mechanics, who service small automobiles for today's nuclear family.

Pauli and Heisenberg
Offer two rules for con-
Fused aristocracies,
Quantum domains:
"Egalitarian
Nonexclusivity
Must be forbidden"; "Un-
Certainty reigns."

April 4, 1976

[a] When in Canada, do as the Canadians do...

Quaxiti quuxiti,	Quaxity quuxity,
Remus et Romulus	Remus and Romulus,
Gemini fratres a	Wolf-nourished brothers, were
Lupa aluntur.	Twins, so they say.
Romulus inquit: non	Romulus patiently

'cal 9 1752' explained

Aedificabitur Founded a city; as
Paucis diebus, sed He said, "Rome will not be
Roma nascetur. Built in a day."

[b] Apples grow on trees, but no trees in APLs.

Quaxity quuxity,
Kenneth E. Iverson
Made APL, matrix
Hackers to please,
Not only famous for
Arithmeticity;
Also for characters.
(Wish it had trees!)

[c] Sussman built a hacker for a Ph.D. thesis. When will someone
for a thesis build a hacker that builds a hacker?

Quaxity quuxity,
Gerald J. Sussman made
HACKER so famous for
Loopholes it plugs.
Armed with its knowledge of
Entomological
Manifestations, it
Ferrets out bugs.

The Great Quux
Random Poetry and Other Losses

page 29
printed April 17, 1979

[d] It doesn't take truth to destroy a reputation, but only
a rumor. Yet sometimes the nastiest rumor is the truth.
Consider the one which says EQUIV causes true values (1-bits)
to complement bits of the other argument...

Quaxity quuxity,
XOR's more famous than
Brother EQUIV, though they've
Similar use.
Uncomplementary
Verisimilitude,
Rumor undoubtedly
Cooked EQUIV's goose.

[e] And why did Napoleon clutch his stomach? (Mail \$1000 and a
self-addressed, stamped envelope, and we'll send you a jingle
in 25 words or less.)

Quaxity quuxy, Na-
Poleon Bonaparte
Conquered all Europe, but
What was the use?
Some say the problem was

'cal 9 1752' explained

Megalomania;
Others, he lacked modern
High-platform shoes.

[f] People got excited about BNF, and used it to drive parser
generators. But you still can't really tell how to write
programs in the language.

Quaxity quuxity,
Backus's BNF
Drives a preprocessor,
Generates code.
Parsing is specified
Metasyntactically;
Writing it's easy, but
reading, I'm snowed.

(c) Copyright 1976 by Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 30
printed April 17, 1979

My Favorite Hacks
[to be sung to the tune of
My Favorite Things
from The Sound of Music]

Circular MAPCAR and ANDCA'd negation,
Indirect JMP auto-incrementation,
Tangled spaghetti embroidered in stacks:
These are a few of my favorite hacks.

Mismatched DEFINE-TERMIN pairs with .QUOTEing,
Misbalanced brackets for macroed remoting,
PDP-6's with chess tourney plaques:
These are a few of my favorite hacks.

LAMBDA's as GO TO's and spooling on TPLs,
Flip-flops and bit drops and TRCE's in triples,
Crufty heuristics that prune minimax:
These are a few of my favorite hacks.

When the bugs strike,
When the disks crash,
When I read this verse,

'cal 9 1752' explained

I simply remember my favorite hacks
And then I feel even worse!

-- The Great Quux
 (with apologies to
 Rodgers and Hammerstein)

(c) Copyright 1976 by Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 31
printed April 17, 1979

NOTES

Circular MAPCAR

A constant or periodic series of arguments can be given to a mapped function in LISP by supplying a circular list to MAPCAR. For example:

```
((LAMBDA (C3)
  (RPLACD C3 C3)
  (MAPCAR 'CONS C3 THE-LIST))
 (LIST 3))
```

conses a 3 onto every element of THE-LIST. Another way to use this hack is to use MAP, which allows the mapped function to splice new things into the circular list on the fly.

ANDCA'd negation

'cal 9 1752' explained

The sequence of PDP-10 instructions

```
MOVN A,NUMBER
ANDCA A,NUMBER
JUMPE A,FOO
```

jumps to FOO iff NUMBER is an exact power of two.

Indirect JMP auto-incrementation:

On the PDP-11, the instruction `JMP @(R4)+` can be used as a combined subroutine call and subroutine return, using the technique of threaded code.

Tangled spaghetti

Spaghetti stacks, as described by Bobrow and Wegbreit.

The Great Quux
Random Poetry and Other Losses

page 32
printed April 17, 1979

Mismatched DEFINE-TERMIN

Bizarre effects can be obtained in the MIDAS assembly language for the PDP-10 by using `.QUOTE` to get in mismatched `DEFINE-TERMIN` pairs. For example:

```
DEFINE DEFSYM X/
IRPS Z,,[X]
IFNDEF Z,X
.ELSE [
    $$STEM1==Z
    X
    IFN Z-$$STEM1,.ERR Z MULTIPLY .QUOTE`.QUOTE/DEFINED/`
]
.ISTOP
TERMIN TERMIN
```

The multiple `.QUOTE`s allow use of the word `DEFINED` in the error message.

Misbalanced brackets

'cal 9 1752' explained

A standard MIDAS hack for accumulating text in a macro to be assembled at a remote location is:

```
;initialization:
  DEFINE BNKBLK OP
  OP
  TERMIN                ;BNKBLK accumulates text.

  DEFINE BLCODE NEWCFT
  BNKBLK [DEFINE BNKBLK OP
  OP]NEWCFT
  TERMIN
  TERMIN                ;BLCODE adds its arg to the end of BNK

;add some text:
  BLCODE [FOO]          ;add FOO.
  BLCODE [BAR]          ;add BAR (note BLCODE inserts CRLF's,

;assemble what has been accumulated:
  BNKBLK                ;which expands into ...
;   FOO
;   BAR
```

This hack depends on the misbalancing of the brackets in BLCODE with respect to the DEFINE-TERMIN pairs.

The Great Quux
Random Poetry and Other Losses

page 33
printed April 17, 1979

PDP-6's

The MIT AI Lab PDP-6 has won chess trophies and plaques running Greenblatt's MacHac program.

LAMBDA's as GO TO's

LAMBDA expressions of no arguments can be used to simulate GO TO statements in a straightforward manner. (See LAMBDA: The Ultimate Imperative, Gerald Jay Sussman and Guy L. Steele., M.I.T. A.I. Lab Memo.)

Flip-flops and bit drops

If you don't know about these, forget it.

TRCE's in triples

The sequence of PDP-10 instructions:

```
TRCE A,42
TRCE A,42
TRCE A,42
```

'cal 9 1752' explained

exchanges the 40 bit and the 2 bit in accumulator A.

Crufty heuristics

The alpha-beta technique, the gamma heuristic (described by Newborn in his book "Computer Chess"), and others, are used to prune minimax game trees.

Notes (c) Copyright 1976 by Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 34
printed April 17, 1979

:PDUMP

99 blocks of crud on the disk,
99 blocks of crud!
You patch a bug, and dump it again:
100 blocks of crud on the disk!

100 blocks of crud on the disk,
100 blocks of crud!
You patch a bug, and dump it again:
101 blocks of crud on the disk!

(Repeat ad nauseam...)

-- The Great Quux 2 December 1976

(c) Copyright 1976 by Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 35
printed April 17, 1979

ZOOM Poems

This sequence of verses was written at a time when a party game called "Zoom, Schwartz, Profigliano" had become popular around the AI Lab (as a result of a trip to Talbot House in January, 1977). For some reason Marilyn Matz didn't like the game, and her standard response to "Zoom!" was "Bitch!". Most of these verses were jokes sent to Marilyn and other interested parties. They all contain acrostics.

Zebras slinking in the night
On twinkling hooves betray their flight.
Oh, would that tigers, burning bright
Might eat them, get them out of sight!
-- 13 May 1977 14:55

Zealous though a hacker be,
Obnoxious bugs cavort with glee.
One poor, defeated hacker shrugs:
May be no way to kill dem bugs!
-- 13 May 1977 16:47

Stupid the furor which AI detractors
Collectively raise over Weizenbaum's book.
Hackers will always, I think, remain hackers, and
Weizenbaum's slander has REAL hackers shook.
All the real hackers know hacking's a hobby;

'cal 9 1752' explained

Randoms are those whom computing consumes.
Terribly artful the well-hacked-up frobbie, but
Zany the work of thee fools Joe assumes!

-- 13 May 1977 17:00

[A lighthearted reference to the chapter on compulsive hacking in Prof. Joseph Weizenbaum's interesting book "Computer Power and Human reason".]

The Great Quux
Random Poetry and Other Losses

page 36
printed April 17, 1979

Perhaps the best way to explain my position
Relies on the beauty of LISP.
Orthogonal issues, obscured by the syntax,
Frightfully kludgy (I say with derision),
Ignore, but usee parens, so crisp.
Green giants (I'm rooting for bagels, you see --
Lox, perhaps also, particularly)
I think, are the greatest of users.
A tiny compiler has no chance of winning;
No smarts, it's a bag-biter from the beginning,
One of the greatest of losers!

-- 13 May 1977 17:11

[There are some involved puns in this last one. Note particularly that "green giant" is an inside term means an NCOMPLR (MacLISP compiler) running interpretively, and also the pun on "green giant rutabagas", a favorite phrase around the lab.]

[Sent to Marilyn:]
Before you complain,
I Humbly maintain
That poetry's loss is yet hackerdom's gain.
Clearly these verses are versus your pleasure; I
Hope you'll feel free to respond in full measure!

-- 13 May 1977 17:20

Tightly zipped her garish pantsuit,
Hurried on, electric, royal,
Eating of a rancid orangefruit,
Grasping Mars' own rapier foil

'cal 9 1752' explained

Runs Sabrina, young, inchoate,
Eager child, to stop Giganto.
Anger hers, now under loquat
Trees waits, silent, incognito.
Quiet, armed she now springs ambush;
Ugly runs the monstrous nebbish.
Ugly trash this awful onrush --
Xebeans, Zebras read no rubbish!

-- 13 May 1977 20:16

[Allow me to draw your attention to the initial letters of the first, second, penultimate, and last words on each line; they form four acrostics. The last two lines are a meta-comment on the verse itself.]

(C) Copyright 1977 by Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 37
printed April 17, 1979

Some Random Verses

There was in ITS an escape code (whose name was DQOT) used internally in the character streams for terminal output which was undocumented for some reason. A call went out for jingles in 25 words or less.

DQOT'S myster-
ious and very weir-
d; it quotes another char-
acter, as noted where?

-- 1 September 1977

Es schneit, es schneit
With all its might!
I say, der Schnee
has come today!

-- 5 December 1977

(C) Copyright 1977 by Guy L. Steele Jr. All rights reserved.

The Great Quux
Random Poetry and Other Losses

page 38
printed April 17, 1979

Written in a Recitation (8.321 Quantum Mechanics I)
at MIT, 11:30 AM 30 September 1977

October Eve's
September morn
Is warm and dry
And I, forlorn:
Alone I sit
Amidst my friends
In heat of day
That never ends.
The heat I do not mind
So much;
It's countered by
The wind's soft touch --
The dryness, though,
Is more severe;
It parches throat,
And eye, and ear.
The heat is Nature's way
Of keeping
Man to frequent
Fall-asleeping;
But never desert wind
So dry
As chalkboard dust
(Square root of pi
Times sigma [i from
One to N]
Of alpha-eigenket,
And then...)

For when the sun
Beats down its worst
A little water
Quenches thirst.
But Man's devised
The dryest yet:
A dryness
Nothing real can wet,
For heat with weather
Correlates,
But dryness --
When we recitates!

-- The Great Quux

(c) 1977 Guy L. Steele Jr. All rights reserved.

The Great Quux

page 39

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Hackers Song

The Hacker's Song

by Dave Touretzky and Don Libes

(with no apologies whatsoever to Monty Python)

I'm a hacker and I'm okay.
I work all night and I sleep all day.

I wrote some hacks in APL,
each on a single line.
They're mutually recursive,
and run in n-squared time!

(chorus:)

Oh, he's a hacker and he's okay.
He works all night and he sleeps all day.
I'm a hacker and I'm okay.
I work all night and I sleep all day.

I wrote two hacks in MACRO,
with UUOs galore.
One plays Nim on the console lights,
while the other zeros core!

(chorus)

I wrote a hack in Snobol,
with FORTRAN subroutines,
It spits out trashy stories,
for ladies' magazines!

(chorus)

I wrote some hacks in InterLisp,
they barely fit in core.
The swapper thrashed its guts out,
So now it runs no more.

(chorus)

I wrote a hack in microcode,
with a goto on each line,
it runs as fast as Superman,
but not quite every time!

(chorus)

I wrote some hacks in Ada,
and still can't run them yet,
Do you suppose we'll see that day?
On it, I wouldn't bet!

(chorus)

I wrote a hack for UNIX
While it was still in vogue.
It knows the tricks to pacman,
and plays mean games of rogue!

'cal 9 1752' explained

(chorus)

I wrote some hacks, distributed,
across our neat gateway.
Each one of its 10 functions
kills RIG in a different way!

(chorus)

I wrote some hacks in Mlisp,
to edit files of root.
It writes them back no-execute,
And now it won't reboot!

(chorus)

I wrote some hacks to manage jobs
with PLITS and IPC.
Its very first activity,
was firing the faculty!

(chorus)

I wrote some hacks with P and V
to synchronize my life.
Now I can't use the bathroom,
I'm deadlocked with my wife!

(chorus)

I wrote a hack (in theory),
it may not ever halt.
But if it does, just watch out...
[Fatal Error: Infinite Page Fault]

(chorus)

I wrote a hack with hough transforms
for our folks at DoD.
It'll guide their fancy missiles,
to Washington, D.C.

I'm a hacker and I'm okay,
I work all night and I sleep all day.
I'll have a system of my own someday,
that'll run my code in a hacked-up way.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Halting Problem is Solvable

Recent Results in Theory of Computing - I

"The Halting Problem is Solvable"

A fundamental question in the graduate computer science curriculum can be posed as follows: Given an average grad student doing a Ph.D, will the student ever complete his dissertation? This problem has been termed the "Halting Problem" and it has been an open problem thus far. In the following, we show that the halting problem is solvable. Furthermore, the problem can be solved within the time stipulated by the Graduate College for Ph.Ds or, in the worst case, with only a constant number of petitions for extensions.

The halting problem was first formulated by Alan Turing, who observed a number of his graduate students being apparently busy all the time but never graduating. Turing tried to solve the problem by first stopping all assistantships after the sixth year and then by purging all games from the research computers. Needless to say, his efforts were fruitless. Later, Church almost succeeded in solving the problem when he placed notices in grad students' mailboxes indicating attractive jobs in industry with several orders of magnitude higher remuneration. The so called Church's thesis was that the halting problem is solvable, given enough financial motivation. Church's idea backfired when grads found out that they have to actually work to earn money in the outside world. Thus, far from solving the halting problem, Church aggravated it (After this, we are not sure whether Church himself graduated). Recently, Cook et al have shown that the halting problem falls under a new complexity class, "NP Hairy". (NP hairy is the class of hopelessly complicated problems with no known solutions. The hardest problem in NP hairy has been shown to be the problem of trying to claim standard deductions in the 1040 form).

In the following, we show that the halting problem is indeed solvable. For this, we assume the existence of a "Super Grad", who is capable of working in any area in CS (except possibly numerical analysis). For notational convenience, we call this super grad, $S_{i,j}$ (written using a funky theoretical CS font). The property of Super grad is that, given the description of any grad (mostly in terms of the number of newsfiles he/she reads every day) and a description of his/her thesis topic, Super grad will either halt with a dissertation or keep publishing technical reports indefinitely. Now, we give Super grad a description of himself and his own thesis topic. If Super grad halts, we are done (and so is he) otherwise we get a stream of technical reports. But by the "fundamental research theorem" of CS Departments (refer to the graduate study manual) any five arbitrary technical reports on unrelated topics can be compiled into a Ph.D thesis. Thus, we are done in the second case too.

Finally, how long does it take for a dissertation to be completed? The time is either less than or equal to the duration allowed by the Grad College for the completion of a Ph.D or it is greater. In the latter case, infinite number of petitions can be filed for extensions. Since the Grad College never remembers previous petitions, the total number of petitions received by the Grad College is always one, a small constant. (QED)

'cal 9 1752' explained

--

Edited by Brad Templeton. MAIL, yes MAIL your jokes to funny@looking.ON.CA
Attribute the joke's source if at all possible. I will reply, mailers willing.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Impossible Code

(Sung to the tune of "The Impossible Dream" from MAN OF LA MANCHA)

To code the impossible code,
To bring up a virgin machine,
To pop out of endless recursion,
To grok what appears on the screen,

To right the unrightable bug,
To endlessly twiddle and thrash,
To mount the unmountable magtape,
To stop the unstoppable crash!

This is my quest -
To debug that code,
No matter how hopeless,
No matter the load,
To write those routines
Without question or pause,
To be willing to hack FORTRAN IV
For a heavenly cause.
And I know if I'll only be true
To this glorious quest,
That my code will run CUSPy and calm
When it's put to the test.

And the queue will be better for this,
That one man, scorned and destined to lose,
Still strove with his last allocation
To scrap the unscrappable kludge!

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Kingdom of Transformation

The Kingdom Of Transformation

I am a wizard in the magical Kingdom of Transformation and I slay dragons for a living.

Actually, I am a systems programmer. One of the problems with systems programming is explaining to noncomputer enthusiasts what that is. All of the terms I use to describe my job are totally meaningless to them.

Usually my response to questions about my work is to say as little as possible. For instance, if someone asks what happened at work this week, I say, "Nothing much," and then I change the subject.

With the assistance of my brother, a mechanical engineer, I have devised an analogy that everyone can understand. The analogy describes the Kingdom of Transformation, where travelers wander and become magically transformed. The kingdom is the computer and the travelers are information. The purpose of the computer is to change information to a more meaningful form. The law of conservation applies here: the computer never creates and never intentionally destroys data.

With no further ado, let us travel to the Kingdom of Transformation....

Far, far away, there is a magical kingdom call the Kingdom of Transformation. A king rules this land and employs a Council of Wizardry. The main purpose of this kingdom is to provide a way for neighbouring kingdoms to make their citizens more useful. This is done by letting the citizens enter the kingdom at one of its rts and travel any of the kingdom's many roads. Citizens are magically transformed along the way. The income of the Kingdom of Transformation comes from the many toll roads within its boundaries.

The Kingdom of Transformation was created when several kingdoms got together and discovered a mutual need for new talents and abilities for their citizens. They employed CTK Inc. (Creators of Transformation Kingdoms, Inc.) to create this kingdom. CTK designed the country, its transportation routes and its laws of transformation and created the major highway system.

Because magic is not truly controllable, CTK

'cal 9 1752' explained

invariably, but unknowingly, creates dragons. Dragons are huge fire-breathing beasts which sometimes injure

December 13, 1984

- 2 -

or kill travelers. Fortunately they do not travel but always remain near their den.

The kingdom has other potential hazards. As roads become older and more weatherbeaten, potholes develop, trees fall on travelers and so on. CTK maintenance men are called to fix these problems.

The wizards play a major role in creating and maintaining the kingdom but get little credit for their work because it is performed secretly. The wizards do not want the workers or travelers to learn their incantations because many laws would be broken and chaos would result.

CTK's grand design is always general enough to be applicable in many different situations. As a result, it is often difficult to use. The first duty of the wizards is to tailor the transformation laws to as to be more beneficial and easier to use in their particular environment.

After creation of the kingdom, a major duty of the wizards is to search for and kill dragons. If travelers do not return on time or if they return injured, the ruler of the country contacts the wizards. If the wizards determine that the injury or death occurred because of the traveler's negligence, they provide the traveler's country with additional warnings. If not, they must determine if the cause was a road hazard or a dragon. If they suspect a road hazard, they call in a CTK maintenance man to locate the hazard and eliminate it, as in repairing the pothole in the road. If they think the cause was a dragon, they must find it and slay it.

The most difficult part of eliminating a dragon is finding it. Sometimes the wizard magically knows where the dragon's lair is, but often the wizard must send another traveler along the same route and watch to see where he disappears. This sounds like a fail-safe method for finding dragons (and a suicide mission for the traveler), but the second traveller does not always disappear. Some dragons eat any traveler who comes too close, others are very picky.

The wizards may call in CTK, who designed the

'cal 9 1752' explained

highway system and the transformation laws, to help devise a way to locate a dragon. CTK also helps provide the right spell or incantation to slay the dragon. (There is no general spell for slaying dragons; each dragon must be eliminated with a different one.)

Because neither CTK nor the wizards are perfect,

December 13, 1984

- 3 -

spells do not always work correctly. At best, nothing happens when the wrong spell is uttered. At worst, the dragon becomes much larger or multiplies into several dragons. In either case, new spells must be found.

If all existing dragons are quiet (in other words, have eaten sufficiently), wizards have time to do other things. They hide in castles and practice spells and incantations. They also devise shortcuts for travelers and new laws of transformation.

As new transformation kingdoms are created and old ones maintained, CTK Inc. constantly learns new things. It learns ways to avoid creating some of the dragons it has created before. It also discovers new and better laws of transformation.

As a result, CTK periodically creates a new "Grand Design" that is far better than the old. The wizards determine when this new design should be brought in. This usually happens when the tourist season is slow or when no important travelers are scheduled to arrive. The kingdom must be closed for the actual implementation and is later reopened as a new and better place to go.

A final question you might ask is what happens when the number of tourists becomes too great for the kingdom to handle in a reasonable period of time (that is, the tourist lines at the ports are too long). The Kingdom of Transformation has three options: 1) shorten the paths that a tourist must travel, 2) convince CTK to develop a faster breed of horses so travelers can finish sooner, or 3) annex more territories so the kingdom can handle more travelers.

Thus endeth the story.

'cal 9 1752' explained

December 13, 1984

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Laws of Computing (reprise)

LAWS OF COMPUTER PROGRAMMING

1. Any given program, if running, is obsolete.
2. Any given program costs more, and takes longer.
3. If a program is useful, it will have to be changed.
4. If a program is useless, it will have to be documented.
5. Any program will expand and fill all of available memory -- plus one byte.
6. The value of a program is proportional to the weight of its output.
7. Program complexity grows until it exceeds the capability of the programmer who must maintain it.

TROUTMAN'S PROGRAMMING POSTULATES

1. If the test installation functions perfectly, all subsequent systems will malfunction.
2. The most harmful error of any program will not be discovered until the program has been in production for at least six months.
3. A Batch Stream that can not be arranged in improper order will be.
4. Constants aren't.
5. Variables won't.
6. Interchangeable Tapes won't.
7. Profanity is the one language that all programmers know the syntax of.

GILB'S LAWS OF UNRELIABILITY

1. Computers are unreliable. Humans are worse.
2. Any system which depends on human reliability is unreliable.
3. Undetectable error are infinite in variety. Detectable errors do not exist, unless deadline is less than three hours away.
4. Investment in reliability will increase until it exceeds the probable cost of errors, or until someone insists on getting some real work done.

BROOK'S LAW

Any manpower added to a late project makes it later.

'cal 9 1752' explained

LAWS OF COMPUTERDUM ACCORDING TO GOLUB

1. Fuzzy project objectives are used to avoid the embarrasment of estimating the corresponding costs.
2. Carelessly planned projects take three times longer to complete than expected. Carefully planned projects take only three times longer to complete than expected.
3. Programmers detest weekly status reporting because it so vividly manifests their lack of progress.

LUBARSKY'S LAW OF CYBERNETIC ENTOMOLOGY

There is always one more bug.

SHAW'S PRINCIPLE

Build a system that even a fool can use, and only a fool will use it.

IBM POLLYANNA PRINCIPLE

Machines should work. People should think.

GRAY'S LAW OF PROGRAMMING

"n+1" trivial tasks are expected to be accomplished in the same time as "n" trivial tasks.

LOGG'S REBUTTAL TO GRAY'S LAW

"n+1" trivial tasks take twice as long as "n" trivial tasks.

WEINBERG'S SECOND LAW

If builders built building the way that programmers program programs, the first woodpecker to come along would destroy civilization.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Laws of Computing

The Laws of Computing.....

- | | |
|------------|--|
| First law | - The Computer is always right. |
| Lemma one | - Programmers are occasionally right. |
| Second law | - The amount of time needed to debug a program is inversely proportional to the time allotted for debugging. |
| Corollary | - Programs never work the first time unless there is virtually unlimited time to complete the program. |
| Third law | - Any programmer can find 90% of his bugs simply by explaining his program to an uninterested observer. |
| Corollary | - The uninterested observer may be sleeping, dead, nonhuman, or, in extreme cases, nonexistent. |
| Fourth law | - The most difficult or nearly impossible programming problems appear obvious or extremely simple to anyone with little or no knowledge of programming. |
| Corollary | - Those problems most easily solved by a programmer appear to be overwhelmingly complicated and marvelous to the layman. |
| Fifth law | - Computers are never more intelligent than their programmers. |
| Corollary | - Most computers are incredibly stupid. |
| Sixth law | - The rarest bugs in any operating system or major programming effort will always show up in a demonstration of its use to prospective users or customers. |
| Corollary | - These bugs usually cannot be reproduced and therefore cannot be located. |
| Lemma one | - Customers will never purchase programs which appear to be riddled with bugs as verified by demonstration. |
| Paradox | - Most programs are unfit for sale. |

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Nine Types of Users

The Nine Types Of Users

Definitions: Scon: Student CONsultant. Scons are people hired to help users learn and work with the University's machinery.
 Pod: A pod is a UNM term for a place where such machinery is made available.

El Explicito -----

"I tried the thing, ya know, and it worked, ya know, but now it doesn't, ya know?"

Advantages: Provides interesting communication challenges.
Disadvantages: So do chimps.
Symptoms: Complete inability to use proper nouns

Real Case: One user walked up to a certain Armenian pod manager and said, "I can't get what I want!" The pod manager leaned back, put his hands on his belt-buckle, and said, "Well, ma'am, you've come to the right place."

Mad Bomber -----

"Well, I hit ALT-f6, shift-f8, CNTRL-f10, f4, and f9, and now it looks all weird."

Advantages: Will try to find own solution to problems.
Disadvantages: User might have translated document to Navajo without meaning to.
Symptoms: More than six stopped jobs in UNIX,
 A 2:1 code-to-letter ratio in WordPerfect

Real Case: One user came in complaining that his WordPerfect document was underlined. When I used reveal codes on it, I found that he'd set and unset underline more than fifty times in his document.

Frying Pan/Fire Tactician -----

"It didn't work with the data set we had, so I fed in my aunt's recipe for key lime pie."

Advantages: Will usually fix error.
Disadvantages: 'Fix' is defined VERY loosely here.
Symptoms: A tendency to delete lines that get errors instead of fixing them.

Real Case: One user complained that their program executed, but didn't do anything. The scon looked at it for twenty minutes before realizing that they'd commented out EVERY LINE. The user said, "Well, that was the only way I could get it to compile."

Shaman -----

"Last week, when the moon was full, the clouds were thick, and Formahaut was above the horizon, I typed f77, and lo, it did compile."

Advantages: Gives insight into primitive mythology.
Disadvantages: Few scons are anthropology majors.
Symptoms: Frequent questions about irrelevant objects.

'cal 9 1752' explained

Real Case: One user complained that all information on one of their disks got erased (as Norton Utilities showed nothing but empty sectors, I suspect nothing had ever been on it). Reasoning that the deleted information went *somewhere*, they wouldn't shut up until the scon checked four different disks for the missing information.

X-user -----
"Will you look at those. . .um, that resolution, quite impressive, really."

Advantages: Using the cutting-edge in graphics technology.
Disadvantages: Has little or no idea how to use the cutting-edge in graphics technology.
Symptoms: Fuzzy hands, blindness

Real Case: When I was off duty, two users sat down in front of me at DEC station 5000/200s that Systems was reconfiguring. I suppressed my laughter while, for twenty minutes, they sat down and did their best to act like they were doing exactly what they wanted to do, even though they couldn't log in.

Miracle Worker -----
"But it read a file from it yesterday!" 'Sir, at a guess, this disk has been swallowed and regurgitated.' "But I did that a month ago, and it read a file from it yesterday!"
Advantages: Apparently has remarkable luck when you aren't around.
Disadvantages: People complain when scons actually use the word 'horse-puckey'.
Symptoms: Loses all ability to do impossible when you're around. Must be the kryptonite in your pocket.

Real Case: At least three users have claimed that they've loaded IBM WordPerfect from Macintosh disks.

Taskmaster -----
"Well, this is a file in MacWrite. Do you know how I can upload it to MUSIC, transfer it over to UNIX from there, download it onto an IBM, convert it to WordPerfect, and put it in three-column format?"

Advantages: Bold new challenges.
Disadvantages: Makes one wish to be a garbage collector.
Symptoms: An inability to keep quiet. Strong tendencies to make machines do things they don't want to do.

Real Case: One user tried to get a scon to find out what another person's E-mail address was even though the user didn't know his target's home system, account name, or real name.

Maestro -----
"Well, first I sat down, like this. Then I logged on, like this, and after that, I typed in my password, like this, and after that I edited my file, like this, and after that I went to this line here, like this, and after that I picked my nose, like this. . ."

'cal 9 1752' explained

Advantages: Willing to show you exactly what they did to get an error.
Disadvantages: For as long as five or six hours.
Symptoms: Selective deafness to the phrases, "Right, right, okay, but what was the ERROR?", and a strong fondness for the phrase, "Well, I'm getting to that."

Real Case: I once had to spend half an hour looking over a user's shoulder while they continuously retrieved a document into itself and denied that they did it (the user was complaining that their document was 87 copies of the same thing).

Princess -----

(Unfair, perhaps, as these tend, overwhelmingly, to be males)
"I need a Mac, and someone's got the one I like reserved, would you please garrote him and put him in the paper recycling bin?"

Advantages: Flatters you with their high standards for your service.
Disadvantages: Impresses you with their obliviousness to other people on this planet.

Symptoms: Inability to communicate except by complaining.

Real Case: One asked a scon to remove the message of the day because he (the user) didn't like it.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Paging Game Game Rules

The Paging Game

(Jeff Berryman, University of British Columbia)

Rules

1. Each player gets several million *things*.
2. Things are kept in *crates* that hold 4096 things each. Things in the same crate are called *crate-mates*.
3. Crates are stored either in the *workshop* or a *warehouse*. The workshop is almost always too small to hold all the crates.
4. There is only one workshop but there may be several warehouses. Everybody shares them.
5. Each thing has its own *thing number*.
6. What you do with a thing is to *zark* it. Everybody takes turns zarking.
7. You can only zark your things, not anybody else's.
8. Things can only be zarked when they are in the workshop.
9. Only the *Thing King* knows whether a thing is in the workshop or in a warehouse.
10. The longer a thing goes without being zarked, the *grubbier* it is said to become.
11. The only way you get things is to ask the Thing King. He only gives out things in multiples of eight. This is to keep the royal overhead down.
12. The way you zark a thing is to give its thing number. If you give the number of a thing that happens to be in the workshop it gets zarked right away. If it is in a workshop, the Thing King packs the crate containing your thing back into the workshop. If there is no room in the workshop he first finds the grubbiest crate in the workshop, whether it be yours or somebody else's, and packs it off with all its crate-mates to a warehouse. In its place he puts the crate containing your thing. Your thing then gets zarked and you never knew that it wasn't in the workshop all along.
13. Each player's stock of things have the same numbers as everybody else's. The Thing King always knows who owns what thing and whose turn it is, so you can't accidentally zark somebody else's thing even if it has the same thing number as yours.

Notes

'cal 9 1752' explained

1. Traditionally, the Thing King sits at a large, segmented table and is attended to by pages (the so-called "table pages") whose job it is to help the king remember where all the things are and who they belong to.
2. One consequence of Rule 13 is that everybody's thing numbers will be similar from game to game, regardless of the number of players.
3. The Thing King has a few things of his own, some of which move back and forth between workshop and warehouse just like anybody else's, but some of which are just too heavy to move out of the workshop.
4. With the given set of rules, oft-zarked things tend to get kept mostly in the workshop while little-zarked things stay mostly in a warehouse. This is efficient shop control.
5. Sometimes even the warehouses get full. The Thing King then has to start piling things on the dump out back. This makes the game slower because it takes a long time to get things back of the dump when they are needed in the workshop. A forthcoming change in the rules will allow the Thing King to select the grubbiest things in the warehouse and send them to the dump in his spare time, thus keeping the warehouse from getting too full. This means that the Thing King won't have to get things from the dump so often. This should speed up the game when there are a lot of players and the warehouses are getting full.

Long Live The Thing King

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Parable of the Two Programmers

The Parable of the two Programmers
Neil W. Rickert
Dept. of Math, Stat., and Computer Science,
University of Illinois at Chicago.

Once upon a time, unbeknownst to each other, the "Automated Accounting Applications Association" and the "Consolidated Computerized Capital Corporation" decided that they needed the identical program to perform a certain service.

Automated hired a programmer-analyst, Alan, to solve their problem.

Meanwhile, Consolidated decided to ask a newly hired entry-level programmer, Charles, to tackle the job, to see if he was as good as he pretended.

Alan, having had experience in difficult programming projects, decided to use the PQR structured design methodology. With this in mind he asked his department manager to assign another three programmers as a programming team. Then the team went to work, churning out preliminary reports and problem analyses.

Back at Consolidated, Charles spent some time thinking about the problem. His fellow employees noticed that Charles often sat with his feet on the desk, drinking coffee. He was occasionally seen at his computer terminal, but his office mate could tell from the rhythmic striking of keys that he was actually playing Space Invaders.

By now, the team at Automated was starting to write code. The programmers were spending about half their time writing and compiling code, and the rest of their time in conference, discussing the interfaces between the various modules.

His office mate noticed that Charles had finally given up on Space Invaders. Instead he now divided his time between drinking coffee with his feet on the table, and scribbling on little scraps of paper. His scribbling didn't seem to be Tic Tac Toe, but it didn't exactly make much sense, either.

Two months have gone by. The team at Automated finally releases an implementation timetable. In another two months they will have a test version of the program. Then a two month period of testing and enhancing should yield a completed version.

The manager of Charles has by now tired of seeing him goof off. He decides to confront him. But as he walks into Charles's office, he is surprised to see Charles busy entering code at his terminal. He decides to postpone the confrontation, so makes some small talk then leaves. However, he begins to keep a closer watch on Charles, so that when the opportunity presents itself he can confront him. Not looking forward to an unpleasant conversation, he is pleased to notice that Charles seems to be busy most of the time. He has even been see

'cal 9 1752' explained

to delay his lunch, and to stay after work two or three days a week.

March 20, 1985

- 2 -

At the end of three months, Charles announces he has completed the project. He submits a 500 line program. The program appears to be clearly written, and when tested it does everything required in the specifications. In fact it even has a few additional convenience features which might significantly improve the usability of the program. The program is put into test, and, except for one quickly corrected oversight, performs well.

The team at Automated has by now completed two of the four major modules required for their program. These modules are now undergoing testing while the other modules are completed.

After another three weeks, Alan announces that the preliminary version is ready one week ahead of schedule. He supplies a list of the deficiencies that he expects to correct. The program is placed under test. The users find a number of bugs and deficiencies, other than those listed. As Alan explains, this is no surprise. After all this is a preliminary version in which bugs were expected.

After about two more months, the team has completed its production version of the program. It consists of about 2,500 lines of code. When tested it seems to satisfy most of the original specifications. It has omitted one or two features, and is very fussy about the format of its input data. However the company decides to install the program. They can always train their data-entry staff to enter data in the strict format required. The program is handed over to some maintenance programmers to eventually incorporate the missing features.

Sequel:

At first Charles's supervisor was impressed. But as he read through the source code, he realized that the project was really much simpler than he had originally thought. It now seemed apparent that this was not much of a challenge even for a beginning programmer.

Charles did produce about 5 lines of code per day. This is perhaps a little above average. However, considering the simplicity of the program, it was nothing exceptional. Also his supervisor remembered his two months of goofing off.

At his next salary review Charles was given a raise which was about half the inflation over the period. He was not given a promotion. After about a year he became discouraged and left Consolidated.

At Automated, Alan was complimented for completing his project on schedule. His supervisor looked over the program. With a few minutes of thumbing through he saw that the company standards about structured programming were being observed. He quickly gave up attempting to read the program however; it seemed quite incomprehensible. He realized by now that the project was really much more complex than he had originally assumed, and he congratulated Alan again on his achievement.

'cal 9 1752' explained

The team had produced over 3 lines of code per programmer per day. This was about average, but, considering the complexity of the problem, could be considered to be exceptional. Alan was given a hefty pay raise, and promoted to Systems Analyst as a reward for his achievement.

March 20, 1985

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Past Is Prologue

THE PAST IS PROLOGUE
by
Leonard Kleinrock

The past is prologue so they say.
So Scene 1 was played today.
It set the stage to point the way
To high speed nets on Friday.

And old slow IMP, a costly link,
Codes to fix the lines that stink,
Ideas born in tanks that think,
Tomorrow's distance sure to shrink.

But first tonight we'll drink and eat.
We'll take some time good friends to greet.
Hear Bible class from Danny's seat.
Those good old days were bittersweet!

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Power User's Guide to Power Users

THE POWER USER'S GUIDE TO POWER USERS

Power Users never read their software manuals; instead they get petty cash from their secretaries and use it to buy books which contain the phrase "Power User" on the cover. They then keep the receipt, to claim against tax.

Software manufacturers write their manuals badly, and in computerese, in order to con Power Users into buying the manual ("XYZ for the Power User!") a second time. This extra revenue compensates the manufacturers somewhat for all the people who pirate their software and then buy Power User Guides to replace the manuals they never had...

Power Users never read their "Power User's Guide to ..." books, for the same reason they didn't read the software manuals in the first place. They do however skim the first two chapters, in which they make copious annotations (e.g. underlining phrases like "to get a directory listing, type 'DIR C: '. Note do not type the word '', or the quotes.")

Power Users get their companies to buy them 130MHz 80586 PS/4s with 100MB RAM and 5-gigabyte optical drives, which they bring home:

- to run Lotus 1-2-3G spreadsheets, producing PostScript graphs of their mortgage repayments;
- to DTP stern memos forbidding their Real Programmers from using unregistered shareware and PD utilities at work. For this task, they get their computer upgraded with a 4096x4096, 12 billion colour hyper-VGA video display, and the memo employs a minimum of seven different fonts, plus bolding and italics, with at least five revisions to correct spelling errors, and to order the Cc: list in the most politically acceptable manner), and
- to play pirate copies of Tetris and PC-Golf which they haven't realised are infected with a virus.

Power Users scold their children for referring to their machines as personal computers. "It's NOT a PC, Jimmy, it's my Professional Workstation, No Intergalactic Space Zombies for you tonight! Now, go to your room!"

Power Users get an identically equipped PC at work, so they can do the work they would do at home, if only ten-year-old Jimmy would stop playing Intergalactic Space Zombies for five consecutive minutes. The money for this PC comes out of the Real Programmers' software tools budget for the next three years.

Having worked out their mortgage repayments for the next 100 years, and having failed consistently to beat ten-year old Jimmy at Intergalactic Space Zombies, Power Users never touch their computers again; at work, they keep themselves occupied in meetings, so nobody will see them staring blankly at their PC screen. Meanwhile, the Real Programmers who work for them struggle by with aging IBM PCs (the originals ones, with a grudgingly-added Tallgrass disk drives - yuck!)

'cal 9 1752' explained

Rather than read their "Real Users Guide to..." books, Power Users turn to their ten-year-old kids for technical advice ("yes, Jimmy, I understand that, but how do I get the directory on the _D_ drive?")

Power Users get frustrated when they press the 'Print Screen' key and nothing happens: they thump it a dozen times before realising they've left the printer off-line.

Power Users sneak their children in outside office hours to work out why their spreadsheet figures don't add up and the Chairman's end-of-quarter report is due tomorrow.

In a strange twist of human psychology, the ten-year-old children of Power Users think that when they grow up, they'll become Real Programmers and make shit loads of money writing a game better than Intergalactic Space Zombies. (Sadly, they end up chugging out accounting software for Power Users.)

Power Users could master any PC application, if only they could figure out how to start it ("Uhhhm, it must be on this menu somewhere..".)

Power Users attend innumerable Power User courses, where they get a set of loose-leaf binders of notes they never read (but whose titles in genuine imitation gold leaf look impressive beside the "Power User's Guide to..." books which now accumulate a thick layer of dust on the shelf). They also drink a lot, and commiserate with each other how their Real Programmer subordinates are a bunch of overpaid, long-haired layabouts who can't be coerced into wearing shirts and ties, never mind a suit; and of course to swap Power Techniques like how to format a 360k disk in a 1.2MB drive and thus get more than 360k of data onto it ("I'll have my secretary call IBM Technical Support about all the bad sector things I'm getting on this disk.")

Power Users carry a pocket calculator for working out the cell values in their Lotus spreadsheets ("Um, I guess I didn't get to the section on formulas yet in my 'Power Users Guide to Lotus 1-2-3'".)

Power Users think "Your computer is stoned" is part of the DOS copyright banner.

The ten-year-old children Power Users mischievously stick pieces of cheese into every crevice of their parent's mouse, not realising that this causes testicular problems later in life (for the MOUSE, twit!).

Power Users don't think that last joke was funny.

Power Users get their secretaries to call IBM Technical Support to fix their defective mouse, because they're too embarrassed to asked any of their Real Programmer subordinates how to open it to remove the cheese.

When nobody is looking, Power Users pretend their mouse is a toy car, and race it around the desk.

Power Users keep a large box of tissues on their desk to wipe the saliva off the screen after playing Test Drive (BRRRRRM! BRRRRRM!)

Power Users can't figure out how to make their modems stop auto-answering, so they always lunge on their phone when it rings in an effort to beat it. They're never fast enough, and spend the first 30 seconds of the conversation apologising, while the modem auto-ranges, and they

'cal 9 1752' explained

earnestly promise that they'll have their secretary call IBM Technical Support to have the problem rectified.

Power Users panic when they lose those dumb keyboard templates that come with programs like Turd Perfect (which are too brain-dead to have a decent user interface). They invariably mix up the templates when switching between programs.

Power Users have problems with Windows, when they have two or more applications running, but room for only one keyboard template.

Power Users buy those dumb mice that have a nearly full ASCII keyboard built-in to them ("Swiss Army Mouse (tm)").

Power Users believe computer salesmen.

Power Users will buy ANY program that makes wild promises on the box about increasing productivity. These boxes always look impressive on the bookshelf, beside the "Power User" books and course notes.

Power Users use MicroJerk ProjectMeister to schedule their wife's pregnancy, and get confused when they can't work out how to assign tasks and set milestones. They try to persuade the obstetrician to induce labour when she's late.

Power Users unreservedly believe their MicroJerk ProjectMeister when it says the project will be complete at 5pm on the last Friday in September next year, but eighteen months later, they won't believe the Real Programmer who says it'll be done "Real Soon Now (tm)".

Power Users believe the ads for 4GLs and Application Generator packages, and think that in two weeks they'll be able to fire all their Real Programmers. (Ha ha ha... remember "The Last One"?)

-Richard Murnane (richardm@runx.oz.au)

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Rules of Netnews Debating

RULES OF NETNEWS DEBATING (partial list)

Rule #1

The proper time for a vicious ad hominem attack is when you have no logical recourse. If you have been arguing a point with a person or persons for 142 odd weeks, and an article comes across that logically tears down the final shred of evidence that you thought you had, that is the time to call the author of that article (1) a mindless twit who attacks other people's beliefs for no reason, (2) an egotistical flaming typical wombat aggie melonhumping cheesewhizzing nanosexual subuseless MOTNS clamholder whose submissions are apparently sneezed onto his/her terminal, or (3) something unpleasant.

Corollary

Or, if that doesn't work, back track and insist that everything you said was an enormously funny joke and if that person had (a) a sense of humor or (b) any sign of intelligence, he/she would have instantly perceived the delicate and subtle wit of your comments.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Sex Life Of An Electron

The Sex Life of an Electron

by Eddie Currents*

One night when his charge was pretty high, Mirco-Farad decided to seek out a cute little coil to help his discharge.

He picked up Milli-Amp and took her for a ride in his Megacycle. They rode across the Wheatstone Bridge and stopped by a Magnetic field with flowing currents and frolicked in the sine waves.

Micro-Farad, attracted by Milli-Amp's characteristic curves, soon had her fully charged and proceeded to excite her resistance to a minimum. He gently laid her at ground potential, raised her frequency, and lowered her reluctance.

With a quick arc, he pulled out his high voltage probe and inserted it in her socket, connecting them in parallel. He slowly began short circuiting her resistance while quickly raising her thermal conductance level to mill-spec. Fully excited, Milli-Amp mumbled "OHM...OHM...OHM!"

With his tube operating well into class C, and her field vibrating with his current flow, a corona formed which instantly caused her shunt to overheat just at the point when Micro-Farad rapidly discharged and drained off every electron into her grid.

They fluxed all night trying various connectors and sockets until his magnet had a soft core and lost all of its field strength.

Afterwards, Milli-Amp tried self-induction and damaged her solenoids, and, with his battery fully discharged, Micro-Farad was unable to excite his field. Not ready to be quiescent, they spent the rest of the evening reversing polarity and blowing each other's fuses.

BUT WAIT!!! THERE'S MORE!

Micro was a real-time operator and dedicated multi-user. His broad-band protocol made it easy for him to interface with numerous input/output devices, even if it meant time-sharing.

One evening he arrived home just as the sun was crashing, and had parked his Motorola 68000 in the main drive (he had missed the 5100 bus that morning), when he noticed an elegant piece of liveware admiring the daisy wheels in his garden. He thought to himself, "She looks user-friendly. I'll see if she'd like an update tonight."

Mini was her name. She was delightfully engineered with eyes like COBOL and a Prime mainframe architecture that set Micro's peripherals networking all over the place.

He browsed over to her casually, admiring the power of her twin, 32-bit floating point processors and inquired, "How are you, Honeywell?" "Yes, I am well," she responded, batting her optical fibers engagingly and smoothing her console over her curvilinear functions.

Micro settled for a straight line approximation. "I'm stand-alone tonight," he said. "How about computing a vector to my base address?"

'cal 9 1752' explained

I'll output a byte to eat, and maybe we could get offset later on." Mini ran a priority process for 2.6 milliseconds, then transmitted 8K. "I've been dumped myself recently, and a new page is just what I need to refresh my disks. I'll park my machine cycle in your background and meet you inside." She walked off, leaving Micro admiring her solenoids and thinking, "Wow, what a global variable. I wonder if she'd like my firmware?"

They sat down at the process table to a top of form feed of fiche and chips and a bucket of Baudot. Mini was in conversational mode and expanded on ambiguous arguments while Micro gave occasional acknowledgements, although in reality he was analyzing the shortest and least critical path to her entry point. He finally settled on the old, Would-you-like-to-see-my-benchmark routine. But Mini was again one step ahead.

Suddenly she was up and stripping off her parity bits to reveal the full functionality of her operating system software. "Let's get BASIC, you RAM," she said. Micro was loaded by this stage, but his hardware policing module had a processor of its own and was in danger of overflowing its output buffer, a hangup that Micro had consulted his analyst about. "Core," was all he could say, as she prepared to log him off.

Micro soon recovered, however, when Mini went down on the DEC and opened her divide file to reveal her data set ready. He accessed his fully packed root device and was just about to start pushing into her CPU stack, when she attempted an escape sequence.

"No, no!" she cried, "You're not shielded!"

"Reset, baby," he replied, "I've been debugged."

"But I haven't got my current loop enabled, and I can't support child processes," she protested.

"Don't run away," he said, "I'll generate an interrupt."

"No, that's too error prone, and I can't abort because of my design philosophy."

Micro was locked in by this stage, though, and could not be turned off. But Mini soon stopped his thrashing by introducing a voltage spike into his main supply, whereupon he fell over with a head crash and went to sleep. "Computers!" she thought, as she compiled herself, "All they think about is hex!"

* Your mother was **WRONG!** If you leave things laying around long enough, eventually someone **WILL** pick them up for you:-) After leaving this story laying around my website for a lot of years, I received the following note in April of 2006.

Believe it or not, I am the author of that cute little story. I wrote it back in 79 or 80 when I was going to Ferris State College (now a university) in Mich. And had gotten through my second year of electronics there (two more to go) when I was sitting around the dorm in the summer and took typing for an elective class. I got up to about 60 wpm and started thinking of all the stuff and formula's and acronyms and names for all the different units (like mho's for example) and started writing this story. I think a rewrote it a couple of time to get in as many words relating to electronics (flux

'cal 9 1752' explained

lines and tickler coils etc) to where it would make sense only to someone with a background in electronics.

I met a guy from Motorola on a project back in the late 80's and he handed me this story. I told him that I had written the darn thing back in college but of course no one believed me. Well, I'm sure you're withholding judgment also, but I found it on your site and noticed it's all over the internet.

Sure wish I had put my name to it back then. Oh well, at least it's gotten a few smiles out there now and then I'm sure. Well, take care, and on my honor, I'm the true author of this cute little story that I find funny even to this day. Bye!

— Richard Hart

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Structured Programmer's Soliloquy

Structured Programmer's Soliloquy

SP or not SP -- that is the question:
Whether 'tis nobler in the mind to suffer
The rules and exceptions of outrageous FORTRAN
Or to take arms against a sea of transfers
And by structuring end them. To code -- to test
No more; and by a test to say we end
The heartache, and the thousand natural mistakes
That FORTRAN is heir to. 'Tis a consummation
Devoutly to be wish'd. To code -- to test.
To test -- perchance to bomb: aye, there's the rub!
For in that test of code what bugs may come
When we have shuffled of this FORTRAN code,
Must give us pause. There's the respect
that makes calamity of so long lists. [??]
For who would bear the whips and scorns of time-sharing
Th' operating systems wrong, the computer's crash,
The pangs of despis'd code, the turnaround's delay,
The insolence of compilers, and the spurns
That patient coding of FORTRAN takes
When he himself might his quietus make
with PL/I? Who would this FORTRAN Bear,
To grunt and sweat under a weary language,
But that the dread of something after FORTRAN
The undiscover'd country, from whose bourne
No programmer returns -- puzzles the will,
And makes us rather bear those ills we have
Than fly to others that we know not of?
Thus conscience does make cowards of us all,
And thus the native hue of resolution
Is sicklied o'er with the pale cast of thought,
And enterprises of great pith and moment
With this regard their currents turn away
And lose the name of action.

- Henry Kleine and Philip H. Roberts
April DATAMATION

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Sun'll Come Up Tomorrow

The Sun'll Come Up Tomorrow
(to the tune of "Tomorrow",
from the Broadway show "Annie")
Jim Cordy
Queen's University
with very minor mods by magi

The Sun'll come up tomorrow
Bet your bottom dollar that tomorrow
There'll be Suns
Just thinking about tomorrow
Drives away the crashes and the buglist
Of the VAX ...
Tomorrow, tomorrow, the Suns come tomorrow
It's always a day away

The net comes up tomorrow
Bet your bottom dollar that tomorrow
There'll be news
Just thinking about tomorrow
Gives me hope that somehow by tomorrow
I'll have mail ...
Tomorrow, tomorrow, the net comes tomorrow
It's always a day away

T-roff comes up tomorrow
Bet your bottom dollar that tomorrow
There'll be text
Just thinking about tomorrow
Makes me think I'll see my words tomorrow
On the page ...
Tomorrow, tomorrow, T-roff comes tomorrow
It's always a day away

(Bridge)
Whenever we ask of Dave or Lori
When the bugs will be fixed
They smile and say ...

The Sun'll be up tomorrow
Bet your bottom dollar that tomorrow
Things'll be back
Just thinking about tomorrow
Makes my fingers itchy for tomorrow
I'll play 'hack'
Tomorrow, tomorrow, the Sun comes tomorrow
It's always a day away
Tomorrow, tomorrow, the Sun comes tomorrow
It's always a day ... a ... way!

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Ten Commandments for C Programmers

From klode@syslab.csd.uwo.ca Thu Dec 19 14:32:24 1991
Received: from syslab.csd.uwo.ca (rubble) by csd.uwo.ca;
(id AA29259) Thu, 19 Dec 91 14:32:22 EST

Message-Id:
Date: Thu, 19 Dec 91 14:27:34 EST
From: Claude Morin
To: magi@syslab.csd.uwo.ca
Subject: formatted ten-commandments
Status: R

The Ten Commandments for C Programmers

Henry Spencer

- 1 Thou shalt run `_l_i_n_t` frequently and study its pronouncements with care, for verily its perception and judgement oft exceed thine.
- 2 Thou shalt not follow the NULL pointer, for chaos and madness await thee at its end.
- 3 Thou shalt cast all function arguments to the expected type if they are not of that type already, even when thou art convinced that this is unnecessary, lest they take cruel vengeance upon thee when thou least expect it.
- 4 If thy header files fail to declare the return types of thy library functions, thou shalt declare them thyself with the most meticulous care, lest grievous harm befall thy program.
- 5 Thou shalt check the array bounds of all strings (indeed, all arrays), for surely where thou typest ```foo''` someone someday shall type ```supercalifragilisticexpialidocious''`.
- 6 If a function be advertised to return an error code in the event of difficulties, thou shalt check for that code, yea, even though the checks triple the size of thy code and produce aches in thy typing fingers, for if thou thinkest ```it cannot happen to me''`, the gods shall surely punish thee for thy arrogance.
- 7 Thou shalt study thy libraries and strive not to reinvent them without cause, that thy code may be short and readable and thy days pleasant and productive.
- 8 Thou shalt make thy program's purpose and structure clear to thy fellow man by using the One True Brace Style, even if thou likest it not, for thy creativity is better used in solving problems than in creating beautiful new impediments to understanding.
- 9 Thy external identifiers shall be unique in the first six characters, though this harsh discipline be irksome and the years of its necessity stretch before thee seemingly without end, lest thou tear thy hair out and

'cal 9 1752' explained

go mad on that fateful day when thou desirest to make thy program run on an old system.

10 Thou shalt foreswear, renounce, and abjure the vile heresy which claimeth that ``All the world's a VAX'', and have no commerce with the benighted heathens who cling to this barbarous belief, that the days of thy program may be long even though the days of thy current machine be short.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Toaster Happiness Koan

AI Koans: (by Danny)

A disciple of another sect once came to Drescher as he was eating his morning meal. "I would like to give you this personality test", said the outsider, "because I want you to be happy." Drescher took the paper that was offered him and put it into the toaster- "I wish the toaster to be happy too".

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Truth About GNU

From: chain@barn.COM (Charles Chain)
Subject: The Real GNU conspiracy
Date: 20 Jan 90 08:41:15 GMT

After years of research, I have discovered that GNU is not an idealistic group of people promoting free software, but actually a conspiracy of disk drive and memory manufacturers. These bastions of free enterprise, always eager to supplement their sales of bigger and faster hard disks; as well as more and more memory, needed a good reason for this.

People weren't buying disks fast enough for them, they said. So, back in 1978, they hatched a plan to create an excuse for deprived computer weenies to spend money on disks and memory. They saw the work being done at Duke and North Carolina, and they hatched a plan to cash in on this. By brainwashing a UC-Berkeley graduate student and a high school student, the concept of Usenet was put into position. These two became the first unwitting pawns in a game of market oversatuation.

Usenet became very popular, and soon thereafter, every site reading news started requiring more and more disks to store messages, and more and more memory to batch and unbatch all news. The manufacturers celebrated with glee, especially since the recession of the early 80's marked the demise of many companies who were too ethical to sign up for the conspiracy.

The manufacturers were happy, riding a wave of euphoria. But they knew they had better keep moving ahead to make sure that the the market never slacked. The next step was to find a tangential reason, completely unrelated to the first, that would help boost sales. Then the concept arose: a man so blinded by ideology that all attempts to rationalize would be fruitless, but also someone who could believably create a beast that was so large, so unwieldy, and so space-consuming, that it would be declared useful by sheer size. There had to be more than that; to keep it large, it bundled existing pieces back into itself, and there existed a constant flow of releases to the extent that keeping up would be a constant consumption. Stallman was invented. He clicked off perfectly, galvinizing people with his political diatribe, as well as enough code to convince anyone he was for real. It was believed, and EMACS rolled into GNU, becoming the spirit-of-the-age political computing disk consumer. It has worked well.

This made the conspirators even more and more bold. Sales were booming, and their swagger suggested they could not be stopped. And they continued. Their next step was to branch out, stop thinking single machine sites, and more on distributed computing sites. They needed a system that would consume massive disk space, and be plausibly useful enough to require massive addition to memory to run reasonable. It needed to consume the disk space of GNU, but show a real increase of memory for constant use.

Then the idea came: network window systems. Make the screens big, give them 8 bit color, or 24 bit color, use lots of fancy words for overlays, etc., make them feel a need for more memory. In desperation for a name, they called it X. Then they called it X11, taking advantage, once again, of the many-release idea of "updates," with a constant outpouring of software. This, the manufacturers think, will carry us into the 90's.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The UNIX Guru Maintenance Manual

From: cj@eno.corp.sgi.com (C J Silverio)
Subject: unix without words
Organization: SGI TechPubs

During one particularly vitriolic flame war about the uselessness of documentation, I wrote the following proposal. I never posted it, because I am a coward. I finally post it here, for your edification.

Subject: UNIX ohne Worter

Well! I've been completely convinced by the arguments presented here on the uselessness of documentation. In fact, I've become convinced that documentation is a drug, and that my dependence on it is artificial. I can overcome my addiction, with professional help.

And what's more, I feel morally obliged to cease peddling this useless drug for a living. I've decided to go back to math grad school to reeducate myself, and get out of this parasitic profession.

Perhaps it just reveals the depth of my addiction to documentation, but I do see the need for SGI to ship *one* document with [our next release]. I see this book as transitional only. We can eliminate it for [the following release].

Here's my proposal:

TITLE: "UNIX without Words"

AUDIENCE: The UNIX novice.

OVERVIEW: Gives a general strategy for approaching UNIX without documentation. Presents generalizable principles useful for deciphering any operating system without the crutch of documentation.

CONTENTS:

INTRO: overview of the 'no doc' philosophy
why manuals are evil
why man pages are evil
why you should read this book despite the above
"this is the last manual you'll EVER read!"

CHAP 1: guessing which commands are likely to exist

CHAP 2: guessing what they're likely to be called
unpredictable acronyms the UNIX way
usage scenario: "grep"

CHAP 3: guessing what options they might take
deciphering cryptic usage messages

'cal 9 1752' explained

usage scenario: "tar"
guessing when order is important
usage scenario: SYSV "find"

- CHAP 4: figuring out when it worked: silence on success
recovering from errors
- CHAP 5: the oral tradition: your friend
- CHAP 6: obtaining & maintaining a personal UNIX guru
feeding your guru
keeping your guru happy
the importance of full news feeds
why your guru needs the fastest/whizziest machine available
free Coke: the elixir of your guru's life
maintaining your guru's health
when DO they sleep?
- CHAP 7: troubleshooting: when your guru won't speak to you
identifying stupid questions
safely asking stupid questions
- CHAP 8: accepting your stress
coping with failure

Now that I think about it, maybe only chapters 6 & 7 are
really necessary. Yeah, that's the ticket: we'll call it
"The UNIX Guru Maintenance Manual."

cj@eno.corp.sgi.com C J Silverio/Brahms Gang/Berkeley CA 94720

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The UNIX Hierarchy

People who come into contact with the UNIX system are often told: "If you have trouble, see so-and-so, he's a guru," or "Bob there is a real unix hacker." Often, they are baffled by these appellations, and do not pursue the matter further.

What is a "Unix Hacker?" How does he differ from a "guru?"

To answer these and other questions, I present a draft of the ``UNIX HIERARCHY``:

NAME	DESCRIPTION AND FEATURES
----	-----
beginner	<ul style="list-style-type: none">- insecure with the concept of a terminal- has yet to learn the basics of vi- has not figured out how to get a directory- still has trouble with typing after each line of input
novice	<ul style="list-style-type: none">- knows tha `ls' will produce a directory- uses the editor, but calls it `vye'- has heard of `C' but never used it- has had his first bad experience with rm- is wondering how to read his mail- is wondering why the person next to him seems to like Unix so very much.
user	<ul style="list-style-type: none">- uses vi and nroff, but inexpertly- has heard of regular-expr.s but never seen one.- has figured out that `-' precedes options- has attempted to write a C program and has decided to stick with pascal- is wondering how to move a directory- thinks that dbx is a brand of stereo component- knows how to read his mail and is wondering how to read the news
knowlegable user	<ul style="list-style-type: none">- uses nroff with no trouble, and is beginning to learn tbl and eqn- uses grep to search for fixed strings- has figured out that mv(1) will move directories- has learned that "learn" doesn't help- somebody has shown him how to write C programs- once used sed to do some text substitution- has seen dbx used but does not use it himself- thinks that make is a only for wimps
expert	<ul style="list-style-type: none">- uses sed when necessary- uses macro's in vi, uses ex when necessary- posts news at every possible opportunity- write csh scripts occasionally- write C programs using vi and compiles with cc- has figured out what `&&' and ` ' are for- thinks that human history started with `!h'
hacker	<ul style="list-style-type: none">- uses sed and awk with comfort- uses undocumented features of vi- write C code with `cat >' and compiles with `!cc'

'cal 9 1752' explained

- uses adb because he doesn't trust source debuggers
 - can answer questions about the user environment
 - writes his own nroff macros to supplement std. ones
 - write scripts for Bourne shell (/bin/sh)
 - knows how to install bug fixes
- guru
- uses m4 and lex with comfort
 - writes assembly code with `cat >`
 - uses adb on the kernel while system is loaded
 - customizes utilities by patching the source
 - reads device driver source with his breakfast
 - can answer any unix question after a little thought
 - uses make for anything that requires two or more distinct commands to achieve
 - has learned how to breach security but no longer needs to try
- wizard
- writes device drivers with `cat >`
 - fixes bugs by patching the binaries
 - can answer any question before you ask
 - writes his own troff macro packages
 - is on first-name basis with Dennis, Bill, and Ken

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Usenet Oracle Speaks

The Usenet Oracle is now available to answer all your questions!
To find out how to ask the Oracle a question, send e-mail to:

```
oracle@iuvax.cs.indiana.edu  
or {ames,rutgers}!iuvax!oracle
```

with the subject titled "help". You will be mailed a helpfile.

Example "oracularities" are included below and different sets have also been posted to these newsgroups:

```
alt.sex          misc.misc        rec.humor  
alt.sources     news.misc        rec.misc
```

The Usenet Oracle has pondered your question deeply.
Your question was:

```
> How many Mountain Dews will I drink in my life?
```

And in response, thus spake the Oracle:

```
} Well, let me put it this way.  If we stack all the cans one atop  
} another, the stack would reach to the Moon and back 36 times!  Believe  
} it or not!  You will pass enough sweetly carbonated fluid to fill Hoover  
} Dam with enough left over for an Olympic-sized swimming pool or two.  My  
} God, Man!  Consider your actions!  Think about the environment and all  
} that stuff!
```

The oracle has pondered your question deeply.
Your question was:

```
> While walking home today, I found a penny on the sidewalk.  A block  
> further, I found a nickel.  Another block further, I found a dime.  Yet  
> another block further, I found a Kruggerand.  Is there any significance  
> to this?
```

And in response, thus spake the oracle:

```
} YOU FASCIST WHITE-SUPREMACIST APARTHEID LOVING RACIST!  Put that  
} Kruggerand back, or else go back to South Africa where you belong!!!!  If  
} you can't tell the difference between good ol' Mom's Apple Pie [TM]  
} Yankee money and racist forgeries, you shouldn't be allowed outside your  
} playpen.  (This is the long answer.)  
} Short answer:  No.
```

You owe the oracle 3 minutes of life.

The oracle has pondered your question deeply.
Your question was:

```
> Oh, great oracle!  What doest thou do with thy newt's eyes?
```

And in response, thus spake the oracle:

```
} Everybody!  Follow the bouncing ball ...  
}  
} Oh!  A newt's eye is a wonderful thing,
```

'cal 9 1752' explained

```
} It makes you laugh, it makes you sing,  
} If you don't like the news I bring,  
} Better go consult the I Ching.
```

You owe the oracle 4 years of slavery.

The oracle has pondered your question deeply.
Your question was:

```
> Why does it seem like everyone else in the whole world is crazy?
```

And in response, thus spake the oracle:

```
} The basic axioms here are:  
} (1) Sane people think that sane people are sane, and that crazy people  
}     are crazy.  
} (2) Crazy people think that sane people are crazy, and that crazy  
}     people are sane.  
} And our premise is:  
} (3) You think that everybody else is crazy.  
} This leads us inexorably to one of two conclusions.  Either  
} (A) You are sane, everybody else is crazy.  Or  
} (B) You are crazy, everybody else is sane.  
} To decide between these, we note the empirical observation:  
} (4) I am sane.  
} Therefore: you are crazy, everybody else is sane.
```

You owe the oracle 2 cases of root beer.

The Usenet Oracle has pondered your question deeply.
Your question was:

```
> What are those little "crumblies" on the bottom of an English Muffin,  
> and why the hell are they there in the first place?!?
```

And in response, thus spake the Oracle:

```
} Well, I'm glad someone finally asked.  Now I get to expose one of the  
} most diabolical conspiracies of all time!  These little "crumblies", as  
} you so cutely call them, are actually the cocoons of tiny, genetically  
} engineered parasites that, once you eat enough of them, invade your  
} brain cells, especially the centers for speech and propriety, and make  
} you start spelling words with extra letters and driving on the wrong  
} side of the road.
```

The Usenet Oracle has pondered your question deeply.
Your question was:

```
> O Mighty Oracle, please do enlighten me ...  
>  
> Three sailors get a hotel room for $30.00.  ($10.00 apiece) The hotel  
> manager makes a mistake and gives them back $5.  He sends the bell boy  
> up to their room with the $5.  However, the bellboy is upset that he  
> didn't get a tip from the sailors, so he keeps $2 for his tip.  He gives  
> the sailors the remaining three dollars.  That means that the sailors  
> only paid $9 apiece for the room, and the bellboy got $2.  That only  
> adds up to be $29.  Where did the other dollar go?
```

And in response, thus spake the Oracle:

```
} It's lodged snugly and quite permanently in Candy's nasal cavity.
```

'cal 9 1752' explained

```
}  
} You see, the sailors' call girl used it to snort some coke. In her  
} enthusiasm, she snorted the bill, too.  
}  
} The serial number of the bill is D67039973F. It's an `85 issue out of  
} Cleveland.  
}  
} The sailors missed their ship, and are now serving time in the stockade.  
}  
} The hotel manager's incompetance was interpreted as evidence of  
} embezzlement by upper management. He's currently on the run from the  
} FBI.  
}  
} Candy and the bellboy got together. With his $2 and her, uh, nose for  
} money, they compounded their assets into a small fortune in the penny  
} stock market and were last seen on the French Riviera.
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Virus Directory (1992 edition)

VIRUS TYPES From PC COMPUTING Nov '92

AT&T Virus:

Every three minutes it tells you what great service you're getting.

Adam and Eve Virus:

Takes a couple of bytes out of your Apple.

Airline Virus:

You're in Dallas, but your data is in Singapore.

Arnold Schwarzenegger Virus:

Terminates and stays resident. It'll be back.

Chicago Cubs Virus:

Your PC makes frequent mistakes and comes in last in the reviews, but you still love it. (Also known as the Maple Leafs Virus.)

Cleveland Indians Virus:

Makes your 486/50 machine perform like a 286/AT.

Congressional Virus (Party Strain):

Computer locks up, screen splits vertically with a message appearing on each half blaming the other side for the problem.

Congressional Virus (Status Quo Strain):

Runs every program on the hard drive simultaneously, but doesn't allow the user to accomplish anything.

Dan Quayle Virus (College Strain):

Their is sumthing rong with yor compueter, ewe just can't figyour out watt.

Dan Quayle Virus (Murphy Strain):

Prevents your system from spawning any child processes without joining into a binary network.

Elvis Virus:

Your computer gets fat, slow, and lazy and then self-destructs, only to resurface at shopping malls and service stations across rural America.

Federal Bureaucrat Virus:

Divides your hard disk into hundreds of little units, each of which do practically nothing, but all of which claim to be the most important part of the computer.

Freudian Virus:

Your computer becomes obsessed with being marryied to its own motherboard.

Gallup Virus:

Sixty percent of the PCs infected will lose 38 percent of their data 14 percent of the time (plus or minus a 3.5 margin of error).

'cal 9 1752' explained

George Bush Virus (Japanese Strain):

Eats some of your files, then immediately regurgitates them.

George Bush Virus (American Strain):

It starts by boldly stating "Read my text . . . No new files!" onscreen, proceeds to fill up all the free space on your hard drive with new files, and then blames it on the Congress virus.

Government Economist Virus:

Nothing works, but all your diagnostic software says everything is fine.

Healthcare Virus:

Tests your system for a day, finds nothing wrong, and sends you a bill for \$4,500.

Imelda Marcos Virus:

Sings you a song (slightly off-key) on boot up, then subtracts money from your Quicken account and spends it all on expensive shoes it purchases through Prodigy.

Jeffrey Dahmer Virus:

Eats away at your systems resources piece by piece.

Jimmy Hoffa Virus:

Nobody can find it.

Kevorkian Virus:

Helps your computer shut down whenever it wants to.

LAPD Virus:

It claims it feels threatened by the other files on your PC and erases them in "self-defense."

MCI Virus:

Every three minutes it reminds you that you're paying too much for the AT&T virus.

Mario Cuomo Virus:

It would be a great virus, but it refuses to run.

Michael Jackson Virus:

Hard to identify because it is constantly altering its appearance. This virus won't harm your PC, but it will trash your car.

New World Order Virus:

Probably harmless, but it makes a lot of people really mad just thinking about it.

Nike Virus:

Just does it.

Ollie North Virus:

Turns your printer into a document shredder.

Oprah Winfrey Virus:

Your 200MB hard drive suddenly shrinks to 80MB, and then slowly expands back to 200MB.

'cal 9 1752' explained

PBS Virus:

Your PC stops what it's doing every few minutes to ask for money.

Paul Revere Virus:

This revolutionary virus doesn't horse around. It warns you of impending hard disk attack- once if by LAN, twice if by C:.

Politically Correct Virus:

Never calls itself a "virus," but instead refers to itself as an "electronic microorganism."

Quantum Leap Virus:

One day your PC is a laptop, the next day it is a Macintosh, then a Nintendo.

Right to Life Virus:

Won't allow you to delete a file, regardless of how old it is. If you attempt to erase a file, it requires you to first see a counselor about possible alternatives.

Ross Perot Virus:

Activates every component in your system, just before the whole thing quits.

Sears Virus:

Your data won't appear unless you buy new cables, power supply, and a set of shocks.

Star Trek Virus:

Invades your system in places where no virus has gone before.

Ted Turner Virus:

Colorizes your monochrome monitor.

Terry Randall Virus:

Prints "Oh no you don't" whenever you choose "abort" from the "Abort, Retry, Fail" message.

Texas Virus:

Makes sure it's bigger than any other file.

Warren Beatty Virus:

Constantly tries to prove its virility by attaching itself to younger or newer files.

Willard Scott Virus:

Keeps track of all family birthdays and renders verbose birthday wishes each time you request weather predictions.

--

With thanks to
dave@csd.uwo.ca

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The VisiCalc Song

[ala' "Let's Get Physical", made popular by Olivia Newton-John]

I'm savin' all of those back issues of "Byte"
Making the micro conversion
I gotta handle text just right
Ya know what I mean?

I took you to a local computer store
Then to a compu-fair shopping spree
There's nothing left to purchase now
'less it's, programmability...

[BEGIN Chorus (invoked later)]

Let's get VisiCalc*, VisiCalc
I wanna get Visi-Calc, let's invoke VisiCalc
Let me hear your modem talk, your floppies squawk
Let me hear your I/O rock...

[END Chorus]

I've used paper, I've used wood
Tried to keep my pen on the table
It's getting hard, this hardware stuff
Ya know what I mean?

I'm sure you understand what eleven's* do
You know the software intimately
You gotta know, you're bringing out
the VisiPlot* for me...

[Invoke Chorus]

* VisiCalc, VisiPlot are TM's of VisiCorp, Inc.
Eleven is a trademark of Digital Equipment Corp.

{ Original material by Randal L. Schwartz }

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

The Way It Is

THE WAY IT IS

Recently someone called me from one of the "Out on the Floor Offices", an ethereal place rumoured to exist only in hyperspace, populated by mysterious beings called Users.

She was quite frantic. She was having trouble running a program through the computer, and her message was clear enough, although rather ill-conceived: "MY FILES ARE FULL!"

I furrowed my brow, lit a smoke, and explained to her, "Really now, Miss Butterman, I don't have time for this." I slowly exhaled the menthol vapours as I stopped her process, crushing any hope she may have had of ever again seeing that document she had spent three hours slaving over.

"I was a typing this REALLY important letter, and it HAS to be ready in an hour...there's all this stuff on my screen that I didn't type... it says something about an error, should I read it to you?"

"No point. Just press return."

"Oh my, it wants my username. Can I restart that where I left off?"

"Not a chance." I drew another puff and tossed the phone aside.

It occurred to me that if I had to hear one more of those whining complaint sessions, heads were gonna roll. Where do you people GET this stuff? I'm going to tell you what's really going on here. Now LISTEN UP. I'm not going over it a second time:

Computer

The black box that does your work for you. That's all you need to know.

Response Time

Usually measured in nanoseconds; sometimes measured in calendar months. The general rule is: Shut up your complaining about response time.

Hardware

See "Computer." Again, not your concern.

Software

If we want you to know we'll tell you about it, otherwise, leave us alone.

Network

Don't worry about it, we'll take care of it. Use it to send mail among your half-wit selves, and don't think we won't read it all. What do you think we do all day? By the way, Butterman... shame about your mother's pancreas.

'cal 9 1752' explained

Data

The General rule is: Don't use any data files and if you find any, delete them before before I find out about them. In fact, just stay off the computer (see "Response Time").

- 2 -

System Crash

Don't ever call the system manager to tell him you think the computer is down. Don't call him to ask him when it will be up again. The more you bother him, the longer it takes.

Downtime

Like I said, don't ask.

Uptime

Don't be ridiculous.

Vacation

A time during which I don't have to put up with your sniveling. Don't try calling. There's no point.

Computer Room

Keep out, you're not invited. Don't knock on the door- don't even think about it. I broke the phone last time one of you jerks called me, and I'm not about to replace it. And keep your greasy fingers off the windows.

My Office

The name says it all... it's mine; stay out.

Your Problems

not my concern.

Deadlines

The general rule is: Deadlines are not acknowledged by me; they're not my responsibility. Go tell somebody who cares.

Maintenance

- a) A Valid reason for shutting down the system at any time.
- b) Much more important than anything any of you bozos do.
- c) Anything I choose to call maintenance.

Software Upgrades

Far too complex for you to comprehend. If I tell you I'm upgrading the system, just be quietly thankful. It's for your own good, even if it does mean extensive downtime during peak hours.

Electronic Mail

I delete it before reading it, so don't bother sending any to me.

Defaults

We like them just like they are; we chose them for a reason. Don't mess with them; consider them mandatory.

Error Messages

'cal 9 1752' explained

I'm not interested. I'm going to kill your process anyway, so keep them to yourself.

Killing your Process

- a) Don't ever ask why.
- b) It's beyond your control.
- c) No warnings are given.
- d) The highlight of my day.
- e) If you call, it's going to happen. No exceptions.

- 3 -

Password

I reserve the right to change them without notice at any time. I choose them, and the more you bother me, the more degrading yours will be. (Example: BUTTERMAN:SNOTFACE)

Users

- a) They slow down the computer.
- b) They waste my time.
- c) A general nuisance.
- d) Worse than that, actually.

Software Modification

You don't know what you want - we'll tell you what you want. It stays like it is. Period.

Privileges

I've got them, you don't need them. Enough said.

Priority

Mine is higher than yours, accept it. That's the reason my games run faster than your lousy accounting package.

Terminals

Before calling me with your terminal problems, consider this:

- a) Are you prepared to do without one for weeks?
- b) Do you REALLY want your process killed?
- c) Did you just trip over the cord again?
- d) Of course you did

Disk Space

I set the quotas, you live with them. If you need more space, check "Data Files".

Operator

I hired him and I trained him. He does what I tell him to. Usually armed; always dangerous.

Backups

- a) A good idea
- b) If I gave a sh*t
- c) Which of course I don't

Lunch

The only time calling my office won't result in the killing of your

'cal 9 1752' explained

process.

Data Security

That's your problem. I'm certainly not going to lose any sleep over it. My files are locked up tight. I feel secure.

Jiffy

Length of time it takes me to resolve your problem by killing your process.

Eternity

Length of time it takes me to give a sh*t about any problems that can't be resolved by killing your process.

- 4 -

Impossible

- a) It can't be done (as far as you know).
- b) I can't be bothered.
- c) You're starting to annoy me.

Inevitable

- a) Couldn't have been avoided.
- b) Not my fault (as far as you know).
- c) The result of annoying me.

Menus

If it's not on the menu, don't ask for it. It's not available. If it is on the menu, it's probably of no use or it doesn't work. We're working on it (See "eternity").

Utilities

I find them quite useful, you'll find them quite inaccessible. Besides, they're not on the menu, are they. What did I tell you about that?

Nuisance

You

Of course, I reserve the right to add, change, or remove anything from the above list. I'm not asking you to accept these matters without question, I'm telling you.

Now that we all know where we stand, I'm sure there'll be no future problems. If you have any questions or comments please feel free to keep them to yourself. If you feel the need for more information, I highly recommend that you ask someone else.

Sincerely,

System Manager

P.S. The new disk quota of 30 blocks per user became effective yesterday.

'cal 9 1752' explained

Anyone caught exceeding the quota will lose their accounts (this means you, Buttermen!)

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

To C or not to C

The following is reprinted from SIGPLAN Notices, December 1983.

It was written by Wes Munsil during discussions on the topic of whether Pascal or C should be used in a project.

```
Use C, or not use C, that is the question:  
Whether 'tis nobler in the mind to suffer  
The flags and warnings of a rude compiler,  
Or to take arms against a sea of errors,  
And by debugging, fix them? To code, to hack,  
No more; and by a hack to say we end  
The type-check and the thousand other checks  
Pascal is heir to, 'tis a compilation  
Devoutly to be wish'd. To code, to hack;  
To hack! perchance to test: ay, there's the rub;  
For in that hacker's bliss what bugs may come,  
When we have written out this awful code,  
Must give us pause: there's the respect  
That makes development of such long life.
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

To Engineer is Human

Glancing through the book "To Engineer is Human" by Henry Petroski (one of my favourites -- I had earlier quoted a passage from the book which quoted from Herbert Hoover about the engineering profession's hazards, I was once again reminded about the numerous parallels between what the book describes (it emphasizes structural/civil engineering, drawing examples from bridges, buses, materials, etc) and software engineering. For instance, in chapter 4 "Engineering as hypothesis", he describes the definition of structural engineering, which The Structural Engineer, the official journal of the British Institution of Structural Engineers, carries on its contents page:

Structural engineering is the science and art of designing and making, with economy and elegance, buildings, bridges, frameworks, and other similar structures so that they can safely resist the forces to which they may be subjected

Each time I see that passage, I am struck by how close that comes to defining software engineering - a few slight changes and we could have

Software engineering is the science and art of designing and writing, with economy and elegance, applications, libraries, systems, and other similar programs so that they can safely resist the forces to which they may be subjected

The book is strongly recommended reading. Petroski carefully analyzes the problems of failure, and the lessons to be learnt. Some specific details are obviously not relevant (but still interesting) eg. the discussion on metal fatigue, but the overall perspective is relevant to any profession involved in designing/building practical contrivances. Other quotes:

Thus it is the essence of modern engineering not only to be able to check one's own work, but also to have one's work checked and to be able to check the work of others. In order for this to be done, the work must follow certain conventions, conform to certain standards, and be an understandable piece of technical communication.

Success is foreseeing failure.

Economic constraints are often imposed by the demands of the marketplace, but the requirement for elegance is often self-imposed by the best in the profession in much the same way as artists and scientists alike see elegance in the sparsest canvases and the most compact theories -- or in the axiom of minimalist aesthetics and design, "less is more".

"Is it becoming easy to take on design work outside an engineer's area of expertise simply because a software package is available? How can civil engineers guarantee the accuracy of the computer program and that the engineer is qualified to use it properly?" - quote from the announcement for the Mead Prize competition for the best paper on the topic "Should the Computer be Registered" by the American Society of Civil Engineers.

Bugs entered engineering calculations long before the

'cal 9 1752' explained

computer, and wondering whether he had thought of every possible failure mode has kept many an engineer awake (at night).

(He has a fascinating anecdote about his search to check the antecedents of the famous quote "Those who cannot remember the past are condemned to repeat it" from George Santayana's "The Life of Reason". He also quotes Santayana "We must welcome the future, remembering that it will soon be the past; and we must respect the past, knowing that it was once all that was humanly possible.)

Engineering, like poetry, is an attempt to approach perfection. And engineers, like poets, are seldom completely satisfied with their creations.

No one wants to learn by mistakes, but we cannot learn enough from successes to go beyond the state of the art.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Topics in Dog Signal Processing

Topics in DSP (Dog Signal Processing)
G. D. Abbott

FILTERS

Work in Dog Cascade Filters centers around the following problem: Ginger is awakened by the cycling of the refrigerator and begins to bark. Molly, awakened by Ginger's barking, also begins to bark. The neighbors' dogs, depending on certain threshold criteria, may also begin to bark. The neighbors' neighbors' dogs will then bark, etc. In most cases the barking animal density function (BAD) becomes non-convergent. Adequate filtering to prevent dog cascading is essential.

An even worse problem occurs when Ginger, satisfied that the refrigerator poses no threat, becomes interested in the barking of Molly and the neighbors' dogs - and barks back. In this case we need to apply the recursive forms of the filters.

BARK RECOGNITION

The need for good bark recognition is not a pressing one; after all, who cares what the dog wants to say? On the other hand, many researchers prefer this field over human speech recognition because it is easier. First, the vocabulary is highly restricted. The dog only wants to say things like "There's some one at the door" or "I need to go outside" or "I'm hungry"; at least that's all we think the dog should want to say and that's good enough for us. The second advantage of working on bark recognition is that we don't have to "train" the recognition system to the dog's bark, it is easier to train the dog. The training procedure involves setting up a simple microphone and computer system with a three-light display. The bark recognition system can base its discrimination on any one of several features: formant structure, pitch, or loudness. When the dog barks in a certain way a light will indicate that the dog wants to "go outside". Another light will mean that the dog wants to eat. A third light indicates danger at the door. If the appropriate action is taken with each light (i.e. letting the dog out, feeding the dog, yelling "Foo ! Go Away !" at the door) the dog will begin to understand what each bark means and will use them to get what he wants. (This last assertion is controversial. Some psychologists prefer to believe that the association works the other way - the dog begins to want what he gets, and barking in a certain way will cause the dog to suddenly need to go outside.)

BARKER RECOGNITION

Related to the problem of Bark Recognition is that of Barker Recognition. The importance of Barker Recognition can be shown by another example. Ginger, a mature dog, may safely be left inside, alone. Molly, the puppy, exhibits destructive doggy demon processes in the absence of tight Master Control, and, therefore, must be left in the background, or backyard. During long trips by the Master, however, even the mature dog needs a break, and a solution has been proposed, based on Maxwell's Demon, where an automatic door would allow a particular dog to enter and exit freely while excluding all others., bring a new meaning to "Dog Security Systems". The key problem is that of recognizing the "safe dog". While some are attempting to solve this problem with methods involving "paw-print" analysis or robot vision systems that count spots, the barker recognition approach seems to hold the greatest promise, due to its simplicity. Just as bark recognition can be

'cal 9 1752' explained

based on simple forms of analysis, barker recognition is again a matter of training the barker to produce the right stimulus for the desired response. While in high security circles it is noted that undesired dog may learn to imitate the welcome dog (referred to as "dog aliasing"), leading researchers feel this is not currently a problem, as the puppy "probably won't get it".

BARK SYNTHESIS

Communicating our wishes the the dog can often be frustrating (see "Programming the User Friendly Dog", Cottrell 1984). While part of the problem lies in the restricted vocabulary (instruction set) of the dog, DSP researchers feel that another part of the problem is phonetic. Dogs simply do not understand English, especially when they don't want to. Useful devices are now being developed to communicate to the dog in his own language. A hand-held device, including a small speaker and a keyboard marked with dog phonemes, has been developed which can produce sounds that any dog will immediately understand. Useful phrases, such as "I'm happy to see you !", "Let's play !", and "Get away ! This is MY food !", can be produced at the touch of a button. While this approach proves to be wonderfully useful, the acoustic method is now being augmented with tactile stimulation, as researchers have shown that many dogs experience temporary hearing-loss at certain times of the day.

DOG BEAMFORMING

The application of phased-barker arrays for military purposes is classified at the present time and cannot be discussed here.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Translating Seminar Announcements

Translating Seminar Announcements

WHAT HE SAYS

WHAT HE MEANS

Autonomous Navigation Using
Evidential Reasoning Techniques

How to Get Around
the Office by Yourself

Purposeful autonomous behavior requires having reliable locative information in order to help assess the desirability of pursuing particular actions to sense, perceive, or alter one's position in the environment.

You must know where you are in order to get around.

Having the ability to ascertain location based on imperfect perceptual cues that are extracted from real-world contexts is a critical requirement for truly autonomous closed-loop behavior.

You'd like to be able to get around with your glasses off or with your eyes closed.

While some of the well known non-perceptual locative techniques such as dead reckoning, registration of inertial or geo-stationary-satellite location data with an environmental map, and maintaining a global cartesian-map representation throughout movement activities remain topics of investigations, they require near perfect measurements of direction and traversed distances, significant computation resources, and are limited by the range of their sensors.

You can walk around with your eyes closed, but it's hard.

Some people have tried navigating with satellite uplinks, but this didn't work well in our office.

To extend current autonomous navigation capabilities that are limited by these requirements, increasing attention has focused on developing robust perceptual-based inference techniques that derive location hypotheses through the interpretation of perceptual cues within the context of environmental map descriptions.

We have found that we can get around by feeling our way along the wall of our office.

Some recent work on perceptual-based navigation schemes has emphasized the integration of distinct sensor modality data within the context of an environmental model to assess location.

You can feel your way around better if you know the room (that's why we're using our office).

In this talk, we discuss an approach to autonomous perceptual-based navigation that uniquely exploits evidential reasoning (ER) techniques which embody interval probabilistic methods that are based on work of Dempster and Shafer.

Sometimes you have to flip a coin.

'cal 9 1752' explained

We will | describe how such ER techniques can be used | We describe how to get around |
to fuse and interpret multisensor data within | our office on a skateboard |
the context of a sparse spatial map of an of- | that we built ourselves.
fice environment to infer location, and show |
the results of preliminary experiments with |
an autonomous mobile platform.

Translated by Paul Heckbert
and Seth Teller

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Useful Error Message. Not

Yesterday morning I came in and one of our PCs was sitting there, rebooted, with an error message ("No keyboard, press F1 to continue": the world's best error message). Its keyboard was unplugged.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

VAX Privileges

From an old VAX System Dispatch:

```
OPERATING SYSTEM: VAX/VMS V2.1
PRODUCT: VAX/VMS
COMPONENT: LOGINOUT
```

GRPNAM SECURITY HOLE IN LOGIN

PROBLEM STATEMENT:

The GRPNAM privilege is an evil demon, allowing the user to invoke its secret entrance for all manner of nefarious purposes not originally intended.

RESPONSE FROM DEC:

The great wizard VMS confronted the demon, raised his great oaken staff carved in ancient runes, and spoke the magic incantation:

```
"$SETPRV IMAGEACTIVATIONENHANCEDPRIVILEGES $CMKRNL!!"
```

There was a blinding flash of light and puff of smoke, and the demon, reduced to harmlessness, scurried off into the distance.

Where his secret entrance had been was naught but a little pile of ashes, which the wind slowly drifted into letters spelling the words "FIXED IN V2.3".

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Venus Equilateral

From: Beverly Erlebacher
Subject: science fiction
Date: Mon, 26 Mar 90 01:46:28 EST

I have started reading "Venus Equilateral" a book of stories by radio engineer George O. Smith, copyright 1942. In the first chapter he describes the technology of the future:

[Description of cheerful young woman on Venus taking phone message from customer, typing it into teletype, reading it back for confirmation, and assuring customer of delivery on Earth within the hour.]

The punched tape from Operator No. 7's machine slid along the line until it entered a coupling machine.

The coupling machine worked furiously. It accepted the tapes from seventy operators as fast as they could write them. It selected the messages as they entered the machine, placing a mechanical preference upon whichever message happened to be ahead of the others on the moving tapes. The master tape moved continuously at eleven thousand words per minute, taking teletype messages from everywhere in the Northern Hemisphere of Venus to Terra and Mars. It was a busy machine; even at eleven thousand words per minute it often got hours behind.

[Description of beaming the message on the master tape up through the Heaviside layers, to a space station where]

It was amplified, it was dehashed, destatted and deloused.

[It is then sent across 67 million miles of space to Venus Equilateral Station (in Venus Trojan orbit) where]

The signal was amplified and demodulated. It went into a decoupler machine where the messages were sorted mechanically and sent, each to the proper channel, into other coupler machines. Beams from Venus Equilateral were directed at Mars and at Terra.

I have the following questions:

1. By the time of this story, will paper tape technology have advanced to the point where continuous and prolonged 11,000 word per minute tape motion is feasible with reasonable mean time between tape snarls? Consider that we apparently have one machine automatically taking tape from 70 machines.

2. Is there a significant explosion hazard from paper dust raised by all these paper tapes?

3. Is paper dust and chad (or whatever you call the round chad from tape punches) a problem in closed environment of a space station?

4. How can the cheerful young woman assure her customer of 1 hour delivery when the coupler is "often hours behind"? Why are they proud of an obviously underconfigured system?

5. In addition to the cheerful young women, how many people must be employed

'cal 9 1752' explained

to manage tape resources, e.g. fetching new tape from the stock room, changing spools, disposing of used tape, perhaps after archival storage, hauling chad, vacuuming up paper dust, not to mention servicing the machines, untangling tape snarls and patching them up with sticky tape, etc.?

6. How many trees die for the sake of these tapes? What is the cost of lifting the tape from Terra's gravity well and conveying it to Venus and Venus Equilateral? What do they do with all the used tape? In space no one will empty your blue box. I imagine the station enveloped in a writhing cloud of used paper tape and round chad. They could burn it to provide extra CO2 for the hydroponics (if any). Perhaps they fire it into the sun.

The first story seems to be about a cheerful young male engineer and his cheerful young female secretary who work on Venus Equilateral. The engineer has been acting as chief since his elderly boss became ill and retired. The Company has just sent up a pompous bean counter to take over this post and put the business on a more profitable footing by implementing cost-cutting measures. The engineer (who has a phd) is somewhat disgruntled, though still cheerful, and expects the new man to screw up amusingly. Although the reader's sympathies are supposed to be with the engineer, i kind of sympathise with the bean counter. If the engineer is such a hot shot, why doesnt he apply himself to obvious technical improvements, despite his phd? for example, he could eliminate local paper tape operations entirely by wiring up the receiver to the transmitters with some electronic sorting stuff in the middle and avoid the expensive and wasteful practice of locally punching each message twice. the technology is certainly up to it - after all, they seem to be able to manage perfect error-free end-to-end transmissions. they either have compute power for error correction protocols or adequate memory for retries over long transmission delays.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Warning to VAX users

WARNING!!!

This machine is subject to
breakdowns during periods of critical
need.

A special circuit in the machine called "critical detector" senses the operator's emotional state in terms of how desperate he/she is to use the machine. The "critical detector" then creates a malfunction proportional to the desperation of the operator. Threatening the machine with violence only aggravates the situation. Likewise, attempts to use another machine may cause it to malfunction. They belong to the same union. Keep cool and say nice things to the machine. Nothing else seems to work.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

What's a Bit?

```
> I like the teacher who said "RISC chips are faster than CISC chips  
> because they have fewer components, so the electrons don't have as far  
> to go." -- well, yes, but.....
```

Reminds me of my AI teacher who had been pursued down to the hardware level on the subject of knowledge representation. The class had a couple of people who knew nothing about hardware but wanted an answer about "what a bit was". Well the profs final attempt was:

```
student: what's a bit? how does it represent one or zero?  
prof:    ever lick a battery?  
student: yes....  
prof:    that tingle - thats a one!
```

That put an end to that digression.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

X: Dangerous Virus!

Official Notice, Post Immediately

```

X      x
X      x
  X    x
    X
  x    X
x      X
x      X

```

Dangerous Virus!

First, a little history. The X window system escaped from Project Athena at MIT where it was being held in isolation. When notified, MIT stated publicly that "MIT assumes no responsibility...". This is a very disturbing statement. It then infiltrated Digital Equipment Corporation, where it has since corrupted the technical judgement of this organization.

After sabotaging Digital Equipment Corporation, a sinister X consortium was created to find a way to use X as part of a plan to dominate and control interactive window systems. X windows is sometimes distributed by this secret consortium free of charge to unsuspecting victims. The destructive cost of X cannot even be guessed.

X is truly obese - whether it's mutilating your hard disk or actively infesting your system, you can be sure it's up to no good. Innocent users need to be protected from this dangerous virus. Even as you read this, the X source distribution and the executable environment is being maintained on hundreds of computers, maybe even your own.

Digital Equipment Corporation is already shipping machines that carry this dreaded infestation. It must be destroyed.

This is what happens when software with good intentions goes bad. It victimizes innocent users by distorting their perception of what is and what is not good software. This malignant window system must be destroyed.

Ultimately DEC and MIT must be held accountable for this heinous *software crime*, brought to justice, and made to pay for a *software cleanup*. Until DEC and MIT answer to these charges, they both should be assumed to be protecting dangerous software criminals.

Don't be fooled! Just say no to X.

[The following is in extremely fine print across the bottom of the page]

X windows. A mistake carried out to perfection. X windows. Dissatisfaction guaranteed. X windows. Don't get frustrated without it. X windows. Even your dog won't like it. X windows. Flaky and built to stay that way. X windows. Complex nonsolutions to simple nonproblems. X windows. Flawed beyond belief. X windows. Form follows malfunction. X windows. Garbage at your fingertips. X windows. Ignorance is our most important resource. X windows. It could be worse, but it'll take time. X windows. It could happen to

'cal 9 1752' explained

you. X windows. Japan's secret weapon. X windows. Let it get in
your way. X windows. Live the nightmare. X windows. More than
enough rope. X windows. Never had it, never will. X windows. No
hardware is safe. X windows. Power tools for power fools. X windows.
Power tools for power losers. X windows. Putting new limits on
productivity. X windows. Simplicity made complex. X windows. The
cutting edge of obsolescence. X windows. The art of incompetence. X
windows. The defacto substandard. X windows. The first fully modular
software disaster. X windows. The joke that kills. X windows. The
problem for your problem. X windows. There's got to be a better way.
X windows. Warn your friends about it. X windows. You'd better sit
down. X windows. You'll envy the dead.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

Zasm Relative Jumps

ZASM RELATIVE JUMPS

Distraught Z80 twin found crumpled at base of bridge, after allegedly committing suicide. Friends said she was sick with grief over the coming of 256 new PC'd children. "The neighborhood just won't be the same" she could be heard moaning over and over again into her belt. No comment was available from mesmerized big brother BitGraph.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

baby.1

BABY(1) USER COMMANDS BABY(1)

NAME

BABY - create new process from two parent processes

SYNOPSIS

BABY sex [name]

SYSTEM V SYNOPSIS

/usr/5bin/BABY [-sex] [-name]

AVAILABILITY

The System V version of this command is available with the System V software installation option. Refer to Installing SunOS 4.1 for information on how to install and invoke BABY.

DESCRIPTION

BABY is initiated when one parent process polls another server process through a socket connection (BSD) or through pipes in the system V implementation. BABY runs at a low priority for approximately 40 weeks then terminates with heavy system load. Most systems require constant monitoring when BABY reaches it's final stages of execution.

Older implementations of BABY required that the initiating process not be present at the time of completion, In these versions the initiating process is awakened and notified of the results upon completion. Modern versions allow both parent processes to be active during the final stages of BABY.

```
example% BABY -sex m -name fred
```

OPTIONS

-sex
option indicating type of process created.

-name
process identification to be attached to the new process.

RESULT

Successful execution of the BABY(1) results in new process being created and named. Parent processes then typically broadcast messages to all other processes informing them of their new status in the system.

BUGS

The SLEEP command may not work on either parent processes for some time afterward, as new BABY processes constantly send interrupts which must be handled by one or more parent.

BABY processes upon being created may frequently dump in /tmp requiring /tmp to be cleaned out frequently by one of the parent processes.

'cal 9 1752' explained

The original AT&T version was provided without instructions regarding the created process, this remains in current implementations.

SEE ALSO

cigars(6) dump(5) cry(3)

OTHER IMPLEMENTATIONS

gnoops(1)

FSF version of BABY where none of the authors will accept responsibility for anything.

NOTES

baby -sex f -name Cathryn Leigh Beck

completed successfully at the Grey Nuns Hospital on March 30 at 9:59 P.M. after 5 hours of labour. New Mom Chenelle is doing fine, as is the baby, Dad is tickled pink. Both will probably come home sometime on Tuesday. More information can be gotten from Dad by e-mail or when he brings his new little girl by to show her off (should be soon) Celebrations can probably begin in earnest after Dad catches up on all the work he couldn't do this weekend.

Sun Release 4.1 Last change: Just before I left the hospital.

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

bake.1

BAKE(1)

USER COMMANDS

BAKE(1)

NAME

bake - create edible foodstuffs from recipes

SYNOPSIS

```
bake [ -f bakefile ] ... [ -d ] [ -dd ] [ -D ] [ -e ] [ -i ] [ -k ] [ -n ] [ -p ] [ -P ] [ -s ] [ -S ] [ -y ] [ target ... ]
```

DESCRIPTION

bake executes a list of recipe commands associated with each target, typically to create a food item of the same name. bakefile contains entries that describe how to bring a baked item into being with respect to the contents on which it depends. These prerequisite targets are called ingredients. Since each ingredient is a target, it may have ingredients of its own.

Targets, ingredients, and sub-ingredients make up a tree structure that bake traces when deciding whether or not to mix a target. bake recursively checks each target against its ingredients, beginning with the first target entry in bakefile if none is supplied on the command line. If, after processing its all of its ingredients, a target foodstuff is found either to be missing, or to be uncooked, bake reheats it. Optionally with this version of bake, a target can be treated as uncooked when the ingredients used to make it have changed.

To cook a given target, bake executes the list of commands, called a recipe. This recipe may be listed explicitly in the target's bakefile entry, or it may be supplied implicitly by bake.

OPTIONS

- f bakefile
Use the description file bakefile. A - as the bakefile argument denotes the standard input - that is, whatever you pour into it will be baked into the target.
- d Bake until the target is done.
- dd Bake until the target is REALLY done.
- D Bake until the target is REALLY REALLY done - like, until it's burned and crispy around the edges.
- e Check to make sure the target is edible.
- i Include icing in the bake process.
- k Bake a German target (for example, Kuchen as opposed to Cake).

BAKE(1) USER COMMANDS BAKE(1)

- n Do not include nuts.
- N Especially not those yucky crushed almonds.
- p Bake in pie mode. This is where we get the expression "pie a la mode".
- P Bake in specified pan. Default is 11" by 14" oblong pan.
- r Do not read in the default recipe file.
- s May substitute ingredients where appropriate. For example, Crisco may be used in place of butter where shortening is needed.
- S Imitate Sarah Lee baking style.
- y Make sure the yummy bit is set when executing the bake.

USAGE

Refer to *Cooking with UNIX: Beginner's Guide*, and *Bake in Kitchen Utilities for the Sun Workstation* for tutorial information about bake.

This utility is not to be confused with *bake (2)*, which is a utility under the *Head Cold Operating System, UDIX*.

SEE ALSO

cc(chocolate-chip), ar(add_raisins), cd(cook_dinner),
get(buy_ready_made), lex(lick_extra_off_spatula),
sh(slow_heating), sccs(strawberry_cheesecakes)

BUGS

Recipes using too many eggs sometimes turn out crunchy.

Executing a shutdown invariably causes the cake to fall.

An out-of-date milk ingredient is automatically omitted instead of being replaced, even with the `-s` flag.

Bakefiles which are used too often tend to get smeared with batter.

Sun Release 3.5 Last change: 7 February 1989

2

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

ed.1

ED(1)

UNIX Programmer's Manual

ED(1)

NAME

ed - text editor

SYNOPSIS

ed [-] [-x] [name]

DESCRIPTION

Ed is the standard text editor.

Computer Scientists love ed, not just because it comes first alphabetically, but because it's the standard. Everyone else loves ed because it's ED!

"Ed is the standard text editor."

And ed doesn't waste space on my Timex Sinclair. Just look:

```
-rwxr-xr-x 1 root          24 Oct 29 1929 /bin/ed
-rwxr-xr-t 4 root        1310720 Jan  1 1970 /usr/ucb/vi
-rwxr-xr-x 1 root    5.89824e37 Oct 22 1990 /usr/bin/emacs
```

Of course, on the system *I* administrate, vi is symlinked to ed. Emacs has been replaced by a shell script which 1) Generates a syslog message at level LOG_EMERG; 2) reduces the user's disk quota by 100K; and 3) RUNS ED!!!!!!

"Ed is the standard text editor."

Let's look at a typical novice's session with the mighty ed:

golem> ed

```
?
help
?
?
?
quit
?
exit
?
bye
?
hello?
?
eat flaming death
?
^C
?
^C
?
^D
?
```

'cal 9 1752' explained

Note the consistent user interface and error reportage. Ed is generous enough to flag errors, yet prudent enough not to overwhelm the novice with verbosity.

"Ed is the standard text editor."

Ed, the greatest WYGIWYG editor of all.

ED IS THE TRUE PATH TO NIRVANA! ED HAS BEEN THE CHOICE OF EDUCATED AND IGNORANT ALIKE FOR CENTURIES! ED WILL NOT CORRUPT YOUR PRECIOUS BODILY FLUIDS!! ED IS THE STANDARD TEXT EDITOR! ED MAKES THE SUN SHINE AND THE BIRDS SING AND THE GRASS GREEN!!

When I use an editor, I don't want eight extra KILOBYTES of worthless help screens and cursor positioning code! I just want an EDitor!! Not a "viitor". Not a "emacsitor". Those aren't even WORDS!!!! ED! ED! ED IS THE STANDARD!!!

When IBM, in its ever-present omnipotence, needed to base their "edlin" on a UNIX standard, did they mimic vi? No. Emacs? Surely you jest. They chose the most karmic editor of all. The standard.

Ed is for those who can *remember* what they are working on. If you are an idiot, you should use Emacs. If you are an Emacs, you should not be vi. If you use ED, you are on THE PATH TO REDEMPTION. THE SO-CALLED "VISUAL" EDITORS HAVE BEEN PLACED HERE BY ED TO TEMPT THE FAITHLESS. DO NOT GIVE IN!!! THE MIGHTY ED HAS SPOKEN!!!

?

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].

'cal 9 1752' explained

```
C # *          hello polyglots$
C #   main.
C #     perform
C * ) 2>_$$; echo "hello polyglots"; rm _$$; exit
C     print
C     stop run.
C -*,          'hello polyglots'
C
C     print.
C     display  "hello polyglots".          (
C */ int i;                                       /*
C */ main () {                                   /*
C */     i=printf ("hello polyglots\n"); O= i return *O; /*
C */ )                                           (*
C */ begin                                       (*
C */     writeln ('hello polyglots');          (*
C */ )                                           (* )
C * ) pop 60 360                                  (
C * ) pop moveto (hello polyglots) show         (
C * ) pop showpage                               ((
C */ )                                           .(* )
C)pop%     program      polyglot.              *) {*/}
```

Ha, ha, ha. Take me back to [[the alphabetic list](#)] [[the date-ordered list](#)].