Datamotion

# faith, hope and parity

Edited by JACK MOSHMAN

the

kludge

kapers

1
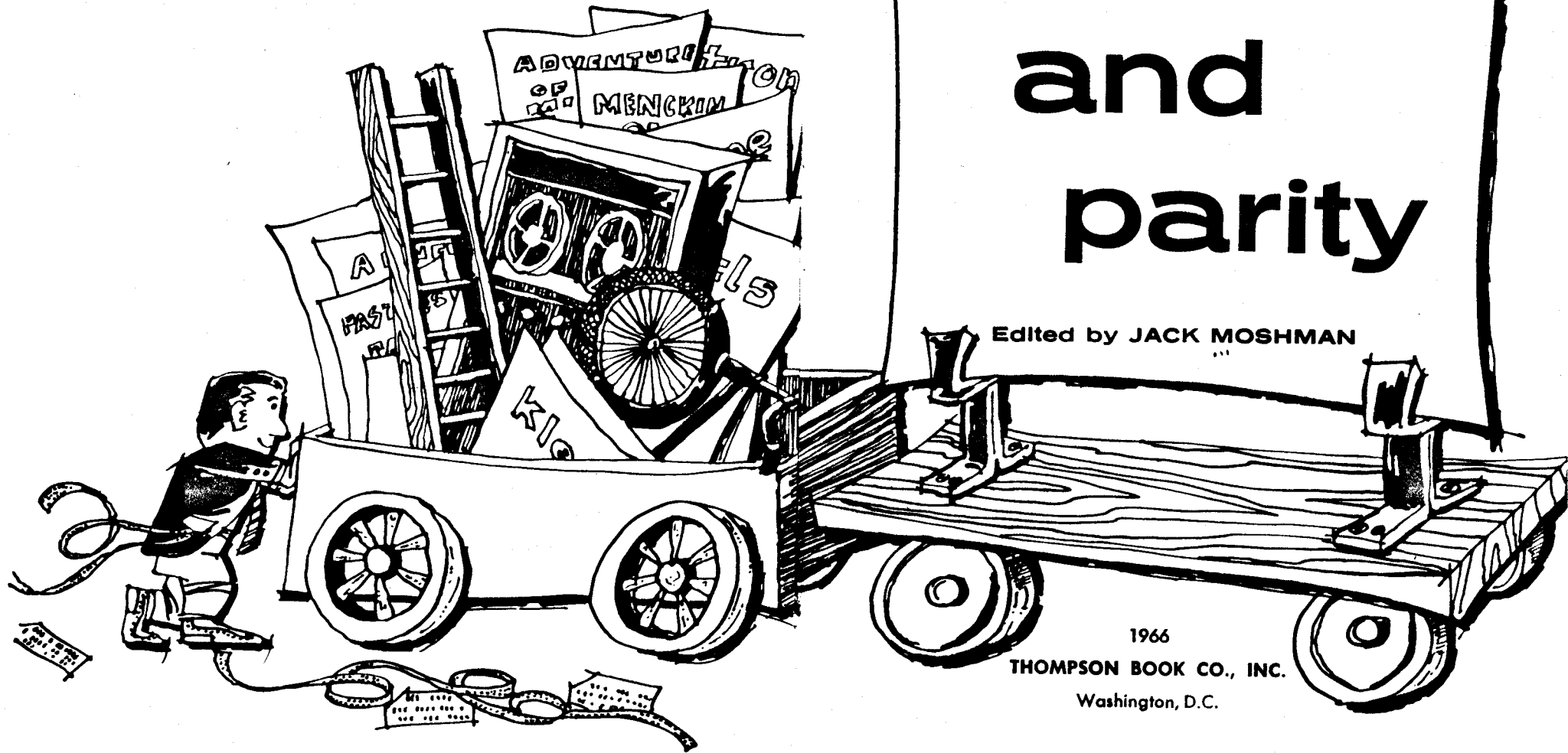
Jackson Granholm's first Kludge article in 1962 stimulated a stream of successors on other phases in the activities of the KKK, the Kludge Komputer Korporation, and its products.

Readers will see in the articles an attempt to evoke humor out of the frustrations in dealing with computers and their manufacturers. The selections reprinted here provide the general flavor of the Kludge Kapers.
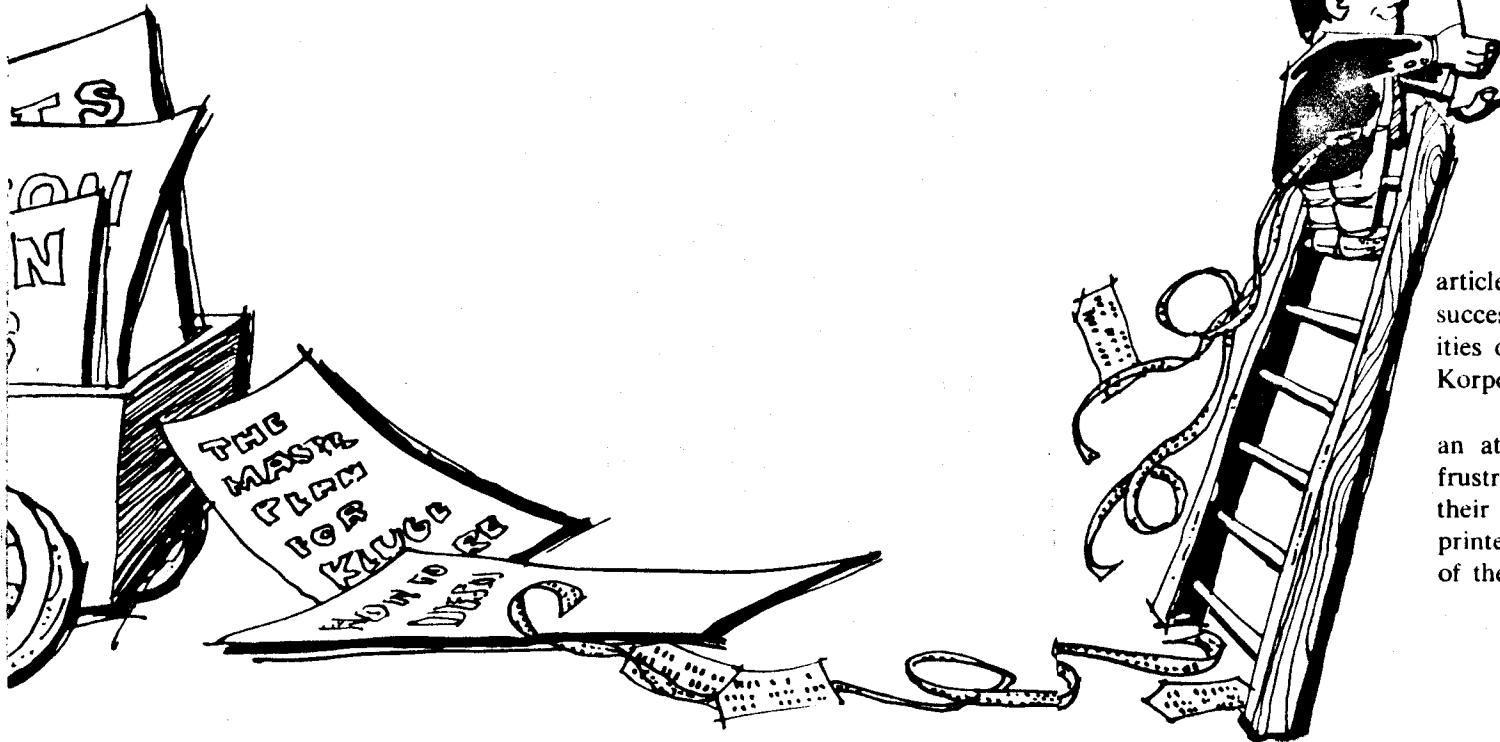
how to
design
a kludge

J. B. JACKSON W. GRANHOLM
Thousand Oaks, California

A phone call to Phineas Burling can be revealing. Phineas Burling is the Chief calligrapher with the Fink and Wiggles Publishing Company, Inc. Fink and Wiggles are, of course, the well-known publishers of the NEW MULTI-LINGUAL DICTIONARY. According to Burling, the word "kludge" first appeared in the English language in the early fifteen-hundreds. It was imported into the geographic region of the lowlands between King's Lynn (then Bishop's Lynn) and the Isle of Ely by Dutch settlers arriving there to reclaim tidelands of the Wash as rutabaga fields. These Dutch Settlers had been driven from their homelands by a great storm in the North Sea which swept inland over The Netherlands, greatly enlarging the Yssel Meer.

3

The word "kludge" is, according to Burling, derived from the same root as the German *Kluge* (Dutch *Kloog*, Swedish *Klag*, Danish *Kløgg*, Gothic *Klaugen*, Lettish *Kladnis*, and Sanskrit *Veklaunna*), originally meaning "smart" or "witty." In the typical machinations of language in evolutionary growth, the word "Kludge" eventually came to mean "not so smart" or "pretty ridiculous." Today the leading definition given by the NEW MULTILINGUAL is, "An ill-assorted collection of poorly-matching parts, forming a distressing whole."
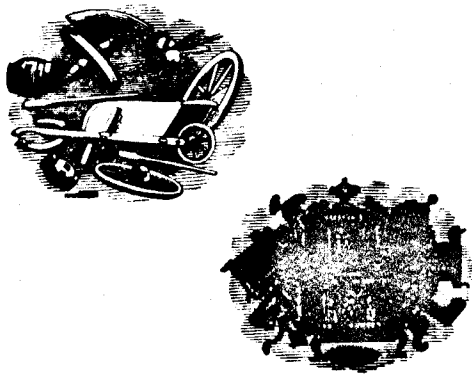
It is in this latter sense that "kludge" is used by computer hardwaremen. Today "kludge" forms one of the most beloved words in design terminology, and it stands ready for handy application to the work of anyone who gins up 110-volt circuitry to plug into the 220 VAC source.

The building of a Kludge, however, is no work for amateurs. There is a certain, indefinable, masochistic finesse that must go into true Kludge building. The professional can spot it instantly. The amateur may readily presume that "that's the way computers are."

As an aid to young computer designers, then, it may be helpful, and a social service, to write down carefully, as Campylos[1] has done, some of the elementary rules for "kludge-manship" or "Designasininity," as Winchell has named it.

A word of caution is in order. One should *not* lash up an arithmetic unit, for example, which does not work. It must actually do arithmetic. The expert Kludge constructor will design his arithmetic unit to perform 2^{512} distinct kinds of addition, each called forth by opcodes six words long. None of these should, of course, be fixed point addition with sign and overflow indication. Overflow, in no instance, should be allowed to merely indicate something. It should generate a three-stage interrupt, preferably interrupting the computer in the next room, or, better yet, in the next plant.

The expert kludge builder will find ways to require three accesses to each of nine index registers in an instruction to query the real-time clock. With a little thought, he can cause automatic clearing of the upper third of memory in the event of an attempt to take square root of a negative number.

But it is in the lashing together of whole modules of equipment that the opportunity for applied kludgemanship presents itself to the hilt. Here, in what is often laughably called "interface resolution" there is an outstanding chance to glitch the user *without his even realizing it.*

Beginning with the most rudimentary of the I/O devices, the console keyput, the correct design approach is to use one of the standard, well-known electric typewriters, but to *alter the character set.* Of course, one chooses an internal machine character set different from that of any other manufacturer (after all, theirs are no good either) and this internal character set must, at all costs, be *different from the one on the console keyput.*

Moving on a step to cards, the field widens. Here one has a choice of card styles. It is *not* good taste to mix these. Do not go round-hole on input and square-hole on output. Stick to one or the other all the way. Use some more subtle device, such as formatting all card input in card-image and all output in Hollerith, or vice versa.

With magnetic tape the paths are well-charted. Use both odd and even parity, and as many widths as one can find reels for. Some special circuitry must, by all means, be designed into the tape connectors in order to make it, if not impossible, at least fabulously expensive to connect the tape machines of any other, second-rate manufacturer to your kludge.

Punched paper tape, in spite of its venerability and antiquity, is the latest hot diggety these days. Here is a great chance to go ape, for if the formats of cards are manifold, those of paper tape are megafold. The proper approach is to select a different one for each use, say four-channel on the input photoreader, nine-channel at the output punch, and six-channel as an auxiliary to the console keyboard. Give the user free a 28 foot-per-hour paper tape rewinder (set up for seven-channel).

Where drum and disc file function as auxiliary stores, there is room for application of some of the magnetic tape technique. But for a full discussion of kludgemanship applied to these and similar devices the reader is referred to the complete and comprehensive work, "Minimum Latencycraft and Random Accessmanship Applied in the Field" by Otis Remack.

The newer devices such as air-film floating cryogenic heads and tunnel diode delay lines are too much in the laboratory or R&D stage to be susceptible to concrete rules of use.
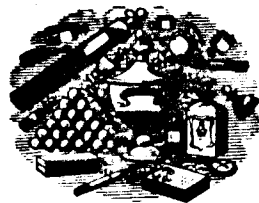
From the foregoing, though, the beginner should begin to form a picture of proper kludge building.[2] The essence of it is the designer who is so clever he outsmarts himself. The method lies as much in character set land and mishapen softwaresville as in techniques of solder and scope. The proper approach lies in producing a machine with maximum unforgivability. To go too far in one direction is to produce a completely impossible machine. To err the other way is to come out with just another ordinary computer.

Enoch Mote suggests the inclusion of a translation function in every hardware data transmission. By Mote's method, information standing punched in Hollerith hexa-decimal will read into core in three's complement form with bit order reversed.

Cicero Beam suggests a random hardware permuter built into every machine to add the contents of a random number table into all instruction regions in memory every 500 milliseconds on the even millisecond. But then, no one pays much attention to what Beam says.

The pricing structure of a machine is often a clue to its status as a kludge or quasi-kludge. Exemplary of the kind of thing one may expect to find is the following for a "basic" machine (Kludgevac 990B):

Central Processor	$ 6200.00 (month)
Console	$ 500.00
Magnetic Tapes (8)	$ 4400.00
Card Reader (1050 cards/min)	$ 211.50
Card Punch (10 cards/min)	$ 987.01
Data Channel (to connect all of the above together)	$56800.00

Outstanding in this case is the exorbitant money asked for the card reader.

The actions of management turn out to be quite important in kludge design, as, indeed, they sometimes are in other matters. It is sad, but true, that a kludge cannot be designed under just any old oganizational structure. One of the most helpful of atmospheres in which a kludge may arrive at full flower is that of complete, massive, and iron-bound de-

partmentalization. It is a good idea if the I/O men, say, not only are not allowed to speak to the mainframe designers, but also that they have, in fact, never met them.

Interface crosstalk should, by all means be done by edict and directive (Beam says "ukase"), and not by memo and design note. After all, someone has to hew to the line on design philosophy, or people will go off in all directions. The "Project Manager," if there be one, should preferably be chosen from among the junior-ranking personnel of the Marketing Research Department.

Finally, and this is advice of the most sound sort, if it looks like, in spite of all efforts, your kludge will begin to resemble an ordinary computing machine, it is time to put more men on the job.

Whatever weird shape your final product may assume, after a year or so of careful kludgecraft, there is one thing to keep always in mind: Don't apologize for it!
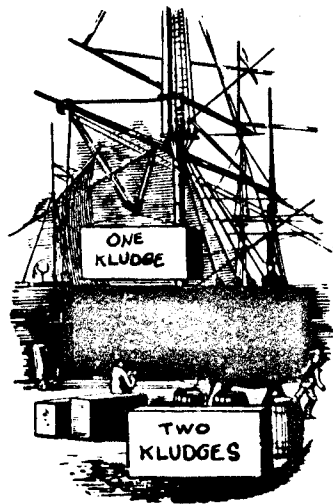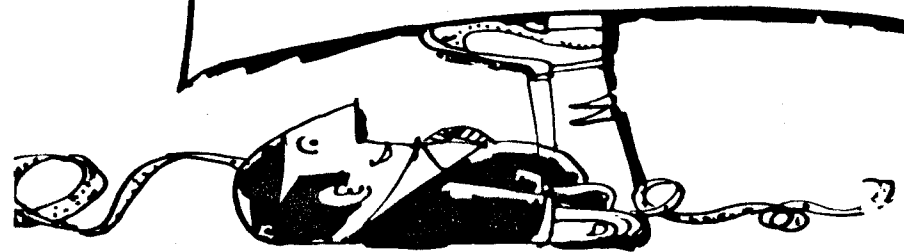
NOTES:
1. Lucy Campylos, Secretary to Dr. Rupert B. Pooble.
2. The Author is indebted to Mr. Norman G. R. Sanders for permission to use his notes on "design glitches in the FRED Project."

8

the master plan for kludge software

MORRIS L. MORRIS & AUSTIN O. ARTHUR

Kludge Komputer Korp.

Previous writers in this series have attempted to show how the road to non-computing can be fostered at its foundations by:

(1) Engineering glitches into the hardware. With suitable ingenuity, these can be either new botches (discovered after the prototype is constructed, usually, and then firmly cast in concrete) or loving repetitions of what never did work right.

(2) Marketing the resulting Kludge with the same techniques used to peddle deodorants and cigarettes. The analogy is painfully accurate.

(3) Applying time-tested rules for maintenance which operate faithfully to minimize uptime. These rules are a sort of check list for the Kludge Fixer so he can save time by listing excuses by the numbers.

9
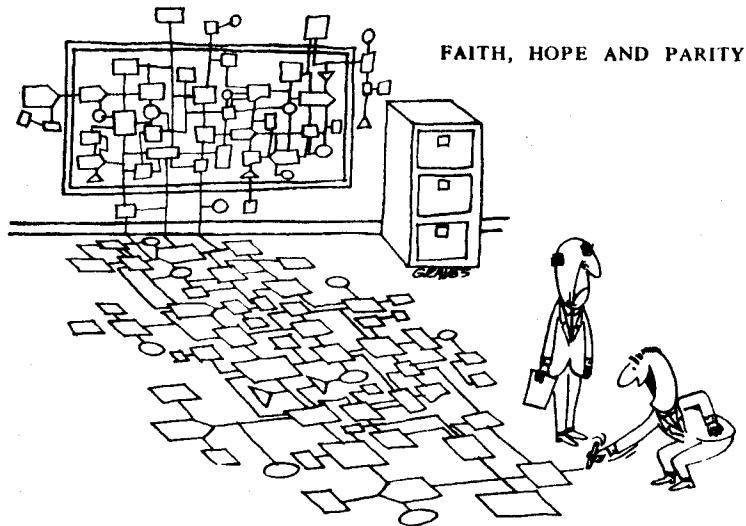
Now, brief reflection quickly reveals that the above approaches are *only* the foundation. The real approach today has far greater scope, since it strikes at the heart of the matter—software. Everyone knows that software is *the* thing. The planners at Kludge Komputer Korporation are keenly aware of this trend. Having pioneered so well in the fundamentals, they can be expected to excel here too. What follows is the basic outline followed at KKK. The Master Plan is the result of years of study and represents the ultimate in software systems for users of the famous line of Kludge hardware.

Kludge software master plan

The cardinal Commandment of any software development program is "Announce it first, worry about producing it later."

The first and most important Edict is, "Always start with fresh programmers." This is a simple rule and its raison d'être is obvious. Any programmer who has used or designed another system has been adulterated or biased and such bias (sometimes referred to as experience) may well influence him in the task he is to perform.

Edict 2. "Never let your fresh programmers talk to the programmers who designed the last software package, if this can be avoided—but at all costs never let any software programmers talk to the people who design, sell or maintain the hardware." This rule needs no explanation.

Edict 3. "Never let the software specifiers talk to the software implementers." This form of warping young programmers' minds is to be avoided like the plague. Software designers always seem to have the weird idea that they better than anyone else, know how to implement their designs.

Edict 4. "Never let a software group know that there are other groups." A weaker form of this rule is also useful, "Never let one software group know what the others are doing." The stronger form of this rule tends to create very high morale or esprit de corps brought about by a feeling of exclusiveness. In the weaker, but more pragmatic form, the effect is acquired by implanting the idea that all of the people in all of the other groups are a bunch of inexperienced poopheads.

Edict 5. "If you must document the software, be sure that the documentation is done by a wholly separate group, preferably technical writers who are not too knowledgeable about computers and programming." Many benefits accrue from this approach, but the greatest one is the high regard with which your documents will be copied. Worry not about their accuracy; users are more adaptable than you think. The proof of this is found in the great number of installations still operating Kludges with *no* documentation.

Turning our attention now to the design phase of software (predicated of course upon the above personal practices) the following rules have served not only the Kludge Komputer Korporation but many of its fellow companies for many years.

Rule 1. "Every software package must have a monitor, irrespective of the size of the Kludge." Without a monitor to occupy between 25 and 50% of the available fast store (and at least ¼ of the backup storage) the users' programmers will become careless. They will eventually discover that one can trade time for space—which leads to reduced utilization and thus, rental income. An appropriately designed monitor on the other hand can be depended upon to *waste* a minimum of 10% of the available time just searching the system tapes

Rule 2. "All error messages must be as coy and ambiguous as possible." The object here is to make the console oper-

ator (and, later, the programmer) unstable. For example, what could be simpler than "AN IMPOSSIBLE ERROR HAS OC-CURRED ON AN UNSPECIFIED UNIT WHILE EXECUT-ING AN UNIDENTIFIABLE PROGRAM." It should be pointed out that the creative souls who can compose such wonderful phrases are rare beasts, indeed, and when found should be coveted and nurtured.

Rule 3. "There should be more phases in the monitor than there are jobs or programmers in a given shop." This is the only realistic settlement to the ever raging argument between the one-phase and three-phase proponents. Besides, how else can you provide the capability for clobbering programmer A's phase p results while running programmer B's phase q interlude if everyone knows what is to happen in each phase? And what better way to keep an operator guessing—you see he is much less likely to call for the KF's under these circumstances —another form of saving not to be discounted.

Rule 4. "Every software routine should have a snappy acronymic name." For instance, we at Kludge Korporation called our Kludge Monitor System KLUMSY; the Kludge assembler, KLAP; and the alternate version, KLAP-TRAP, required for a Kludge with traps (working or not).

Rule 5. "Before letting pragmatic aspects interfere, be sure that the things that count are taken care of. SEE THE SALES DEPARTMENT FIRST AND THE COPY WRITERS IM-MEDIATELY." As we all know, if a Kludge can't be sold on its hardware merits alone, (and it can't) the software must carry the burden.

A Kludge cannot subsist on a monitor alone. Thus it behooves the Kludge software suppliers to provide working languages for the Kludge. The more the merrier. Since the list of OK languages changes from season to season, and position within the list is not constant (or computable), one had best consult one's marketing people to find out which ones have strongest current motivational appeal. Within these bounds, the guiding principles of Kludge languages follow.

Principle 1. "Hop aboard all of the current band wagons." If FORTRAN is the current best selling gee-wiz, write (promise) a better one. Change the name slightly so that you

don't lost your identity, but not so much so that you can't tell who whelped it. Thus, Kludge Korporation's version of FOR-TRAN will be called KLUDGTRAN.

Principle 2. "Always release preliminary undebugged versions of the translators, compilers, generators, assemblers, etc." Why NOT? Let your customers debug the things. Why should you spend your programmer's time and operate a machine with all the awful overhead? If your customers want it badly enough, they'll check it out for you. Unfortunately, the day is gone when you could get him to design and implement it, too.)

Principle 3. "No 'preliminary' or 'field test' version of any translator should be compatible with the monitor or any other translator." The savings realized by the elimination of coordination and liaison would amaze you. This also prevents future coordination and liaison because each "field test" version gets too deeply imbedded for anyone ever to want the final version anyway.

Principle 4. "Join and actively support any and all government sponsored and international magic language generation efforts." This lets you know what the opposition is thinking. But be careful: contribute only those ideas which you know are impossible to implement on the competitors' hardware, or will at least make it look bad. Always be ready with a claim to have a working version of whatever language is the current vogue.

Principle 5. "Never, NEVer, NEVER write a decent or useful training manual for any system." Remember, you may be taking the bread out of the mouth of some striving young author.

Principle 6. "If you must supply an assembler, do it under duress." The best ploy to use to get out of this one is to tell the customer he never ever needs to know the basic language of the machine. All of his problems can be solved with the New International Magic Language Number 6.5. Well, our version of it anyway. As we all know, magic languages are the thing, and you can't sell a Kludge without magic.

We must now turn our attention to software maintenance; an area too often ignored.

Tenet 1. "Each program should be on a separate tape with its own unique format." This is logical. You wouldn't want the corrections for one system to be acceptable to another, would you?

Tenet 2. "Corrections should be distributed at such a rate (empirically determined) as to keep the users from inundating you with requests for additions and changes to the system." It seems best to protect our programmers as much as possible from new and different ideas. Besides, we've always done business this way.
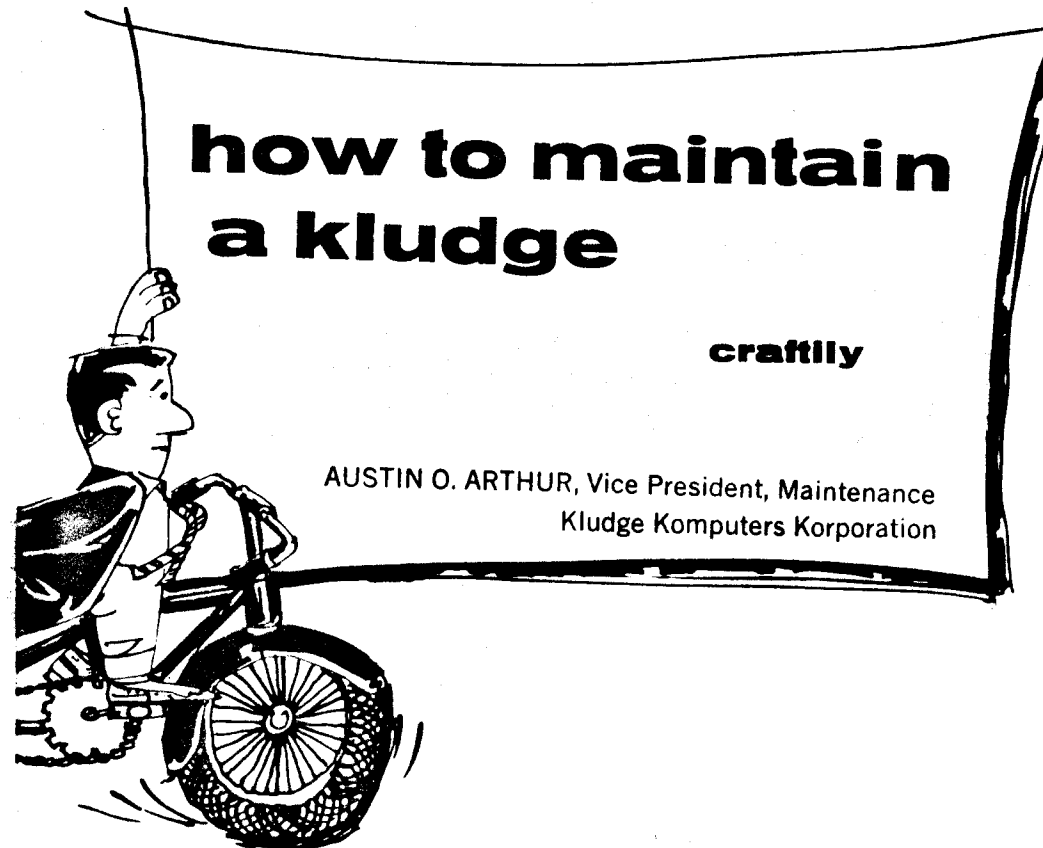
Tenet 3. "Distribute new versions of each system as soon as the previous one is showing signs of being checked-out." This guarantees job security for a very large segment of the programmer community known as "System Programmers."

Tenet 4. "Whenever a new system is proposed or implemented, refuse to continue maintenance on some other (any other) existing system." How far can *you* make a rubber band stretch? We've already provided for keeping our programmers busy!

Tenet 5. "Never let the programmers who implemented the system maintain it." It has been found over the years that many system programmers acquire a certain attachment for their own code and refuse to consider sullying it with corrections. Thus, the only solution is to have some other programmer do the appropriate surgery (preferably a brand new one, fresh from the university) in the form of absolute binary patches.

In closing, we should like to point out that we of the Kludge Komputer Korporation have managed to keep abreast of the competition in software as well as hardware by having a loyal claque. Which brings us to the last Commandment:

"FORM A USERS GROUP," whence springeth all that is worthwhile. Wine and dine them, buy their loyalty and in every way possible make sure that *your* users are satisfied and happy. Channel their desires appropriately. Hold meetings at least twice a year and see that all of the attendees get smashed (we at KKK sometimes pick up the tab) and go home generally feeling loved, wanted and appreciated by all.

how to maintain a kludge

craftily

AUSTIN O. ARTHUR, Vice President, Maintenance
Kludge Komputers Korporation

After proper attention has been paid to the design of the Kludge[1] and to its marketing,[2] the focus shifts to a necessary evil; namely, maintaining the thing. The customers seem to want maintenance, or at least the appearance of it, and without a large service organization, marketing the Kludge is difficult.

Of course, proper maintenance really begins with the design. The alert manufacturer carefully observes simple rules such as:

(1) Design the Kludge around fifteen basic circuit cards. Then, during production and in all subsequent field changes, modify each individual card with a resistor or condenser or such. This process soon produces a machine that really has 4200 distinct circuit cards, and *none of them is*
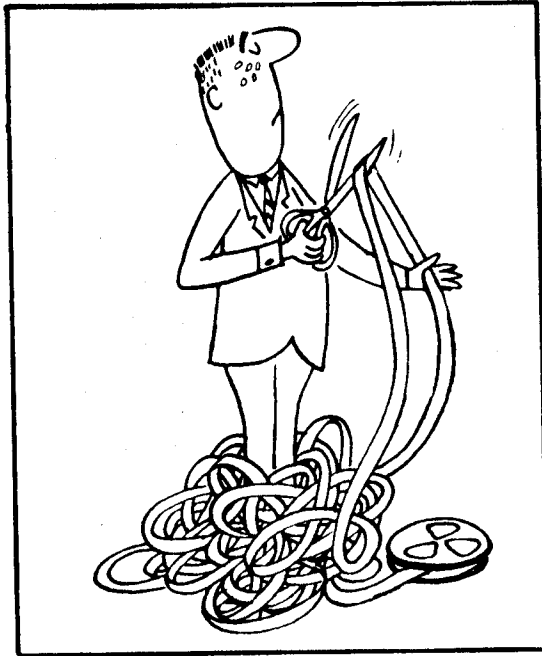
ever available in spare parts. The maintenance man, of course, is continually gulled into thinking he has just the card he needs (and he can reassure the customer to that effect). In trying to add that one resistor, he can delay operation of the machine indefinitely.

(2) Make all checking circuits non-fail-safe. If the checking circuit ever fails, then, the customer can have his machine down for days and never even know it. When he does notice that he's in trouble, he can have lots of fun proving it to the Kludge maintainer.

(3) Be sure to avoid any mention of possible hang-up conditions in the machine's reference manual. When one of these conditions is eventually encountered and the machine hangs up with no apparaent cause, it will induce much merriment between the customer and the soldering iron jockies. Each separate such condition is good for half a day of downtime.

(4) Fix up lots of indicator lights that have meaning only to the maintenance man. *Put these lights on the console.* They amuse visitors. The maintenance man will find them.

He then either has to keep running back and forth to look at them, or he'll need another man to call them out to him. Proper application of this principle can cut the effectiveness of your service crew at least in half.

(5) Don't put all the fuses and circuit breakers in one place. Hide a few in obscure spots. Include some that give no visible indication when tripped; these are best placed *behind* something. It's an easy rule: the less chance there is for a fuse to trip, the more you hide it.

But all the above are obvious rules of design. Kludgemanship comes into its own in the field at the maintenance level. To the customer, the maintenance man (sometimes called the Kludge Fixer; or KF) is the on-the-spot representative of the vendor. The care and feeding of the KF warrants our attention. Again, experience dictates elementary, easy-to-follow rules.

(1) Don't pay them. If you did, you'd upset industry standards, for one thing. And these standards have been sensibly arrived at. After all, if you doubled their pay, you'd have the same quality of work done, wouldn't you, but costing twice as much? Of course you would! So try for high school graduates at the lowest possible salary you can get by with. (If you remind them every day of the glorious opportunities ahead, they can stay at that low level for years. And there'll be a new crop along every June.)

(2) Indoctrinate them into the proper philosophy. Assure them, for example, that when they turn a checked-out machine over to the customer, then until the next scheduled period of maintenance, all troubles described by the customer are imaginary. Or, at the very best, the troubles are intermittent and have gone away by now. Teach them to ask "are you sure it isn't programming trouble?"—this always makes for good feeling.

Point out to them that they, not the customers, are really in charge of the machine. The customers are the ones who keep the tape drives spinning in those awkward intervals between maintenance periods, but the real purpose in all this is to keep the main frame warm.

Have them also learn to instill the proper philosophy into the customer. They should encourage the customer to learn to live with a malfunction, rather than be known as a chronic complainer. Eventually the user will find a way to "program around it," since he can't afford to have it known that he tolerates a machine with less than 90% of uptime.

(3) "Trust your diagnostics"—that's the watchword. Those diagnostics were designed by better men than you, son, and if they work, the machine works. If any part of the system could fail, the all-wise diagnostics would surely have revealed it. So in the final analysis, the job is simple: ignore what the customer is yapping about and run the diagnostics. Then if no errors are reported, the customer was obviously dreaming and can safely be ignored.

Be sure, of course, not to let the KF know anything *about* the diagnostics. If you do, you'll soon have some smart-aleck KF poking into one of them and trying to improve it.

(4) Teach them to take first things first. If the customer reports more than one trouble at a time, then one of the troubles has higher priority than the others. It is only good sense, and good customer relations, to do that one first, right? But now notice that if you play your cards just right, some of the low priority troubles can slide along for weeks. A top notch man can sometimes stall one until his successor (some naive youngster) takes over. One of the all-time great KF's, Herman Schmelzer, managed to ignore a broken start key throughout the entire life of a machine.

(5) Operate on the symptoms, not the disease. This one is so obvious, it's almost not worth listing. Just as an example, suppose a tape drive won't read successive records at high speed. The solution is to lengthen the inter-record gaps. That'll fix it.

Notice how nicely things begin to compound now? If a good solid glitch is designed into the machine, and the KF is carefully trained to operate only on its symptoms (all the while telling the customer that he's imagining his troubles), the Kludge may not function properly for years. And there is never any worry about competition, since all the other manufacturers know all these rules, too.

(6) Introduce your boys at an early age to the telephone. Point out to them that a good man can spend six out of every eight hours talking to other KF's at other installations (not to mention talking to wifey and all the other members of the bowling league). The customers at both installations may believe that their troubles are being fixed.

(7) From time to time, circuit changes will have to be made on the machines in the field. Get several of your best factory KF's to make the change and time them. Pick the shortest of these times. Then when the change is sent out to the field, quote this time as the estimate. The KF in the field can now play a real fun game with his installation managers. "Give me your machine for three hours to make this improvement," he says. It'll really take 53 man-hours, counting the shocking effect of rule (6). For best results, schedule these changes around month-end report time.
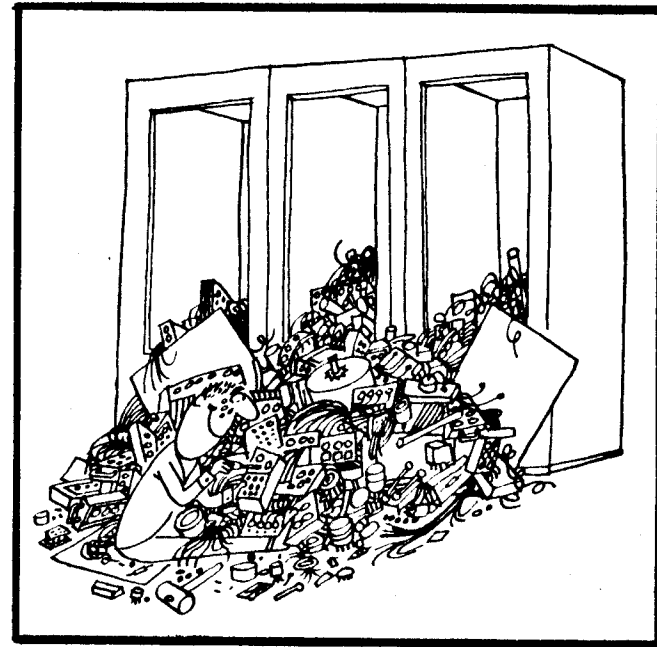
(8) In spite of all these eminently sensible rules, one of your boys may draw one of those oddball customers who insist that the whole Kludge work properly, even to the opcodes that are never used. There is one in every territory. Then the watchword is "If you can't fix it—inspect it." A thorough inspection requires that every part be disassembled, cleaned, oiled, and reassembled. And no customer has the gall to point out, after this lengthy process, that the thing still doesn't work. He's honor bound to prove it to himself all over, and the alert KF can apply rule (2) again. Incidentally, caution the KF's to dispose of the leftover parts after reassembly.

In spite of your good intentions, some of your men will wind up fixing machines. Pretty soon the customers notice this and get spoiled rotten. When this happens, your only recourse is rule 9:

(9) Transfer them.

NOTES:
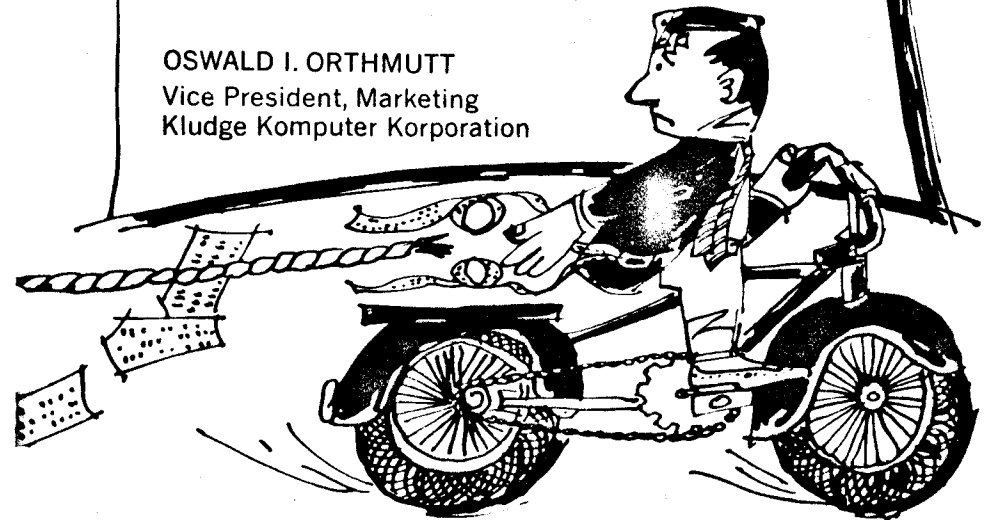
(1) See How To Design a Kludge by Jackson Granholm, Datamation, February, 1962.

(2) See How To Market a Kludge by Oswald I. Orthmut, Datamation, May, 1962.

20

how to market a kludge

shrewdly

OSWALD I. ORTHMUTT
Vice President, Marketing
Kludge Komputer Korporation

With deference to previous and future authors of this series, it should be emphatically understood that to market a kludge is somewhat more difficult than to design and manufacture one. For proof of this congealed pudding, merely reflect on our record. The 650th design of the Kludge Komputter Korp. uses biquinary, BCD, and binary for arithmetic, the Dandy-1000 uses binary internally, but cannot read or punch it; and among the lower production versions is the HOT-FLASH X2E which was born an orphan and has remained parentless for want of massive sales support from our Korporate headquarters in Peachfuzz, Alabama.

To compete or not to compete is the primary marketing question! To answer it properly, money from Peachfuzz must be encouraged. All other factors are incidental and largely

21

unnecessary. To obtain backing, a report of some sort is certainly nice to think about. Typing it up is quite another matter.

To accomplish this, one must analyze with polished shrewdness, the characteristics of the market, the potential reception of unwashed users and ultimately, the performance of KLUDGEVAC 990B.

One method of being certain to obtain the most favorable report is to seek the counsel of a management or marketing consultant with an active curiosity about the Komputer field. Experience is undesirable. And since consultants are generally seated in a strikingly unbiased position, their input may be gently colored so that only correct conclusions are drawn.

After a report has been received, it will be couched in such general terms that it must be interpreted to management, orally and over cocktails. This fact alone guarantees that the budget and staff befitting such an endeavor will be obtained from the Korporate headquarters without curious visitations to the manufacturing facility.

It is best if the above market survey can be completed before the hardware manual is finalized and documented. If a manual on the equipment exists, one runs the risk of having it read and subsequently losing control of the report. If a manual exists before the market survey is in, there is a serious risk of being stuck with a definitive survey report which will never do.

setting the price

After the firm decision to produce is made, the price must be established. In this regard it is best to price as vaguely as possible so that one may compete with entrenched machines *now* in the field. Prices on existing, operable equipment are generally higher than those to be delivered at the same time as the 990B and hence the markup on the 990B will appear to be greater. For added pricing flexibility, offer quotations with a profusion of peripheral gear. Never quote on the main frame alone!

After the greatest possible markup is obtained, hold out for impressive commissions when the letter of intent is
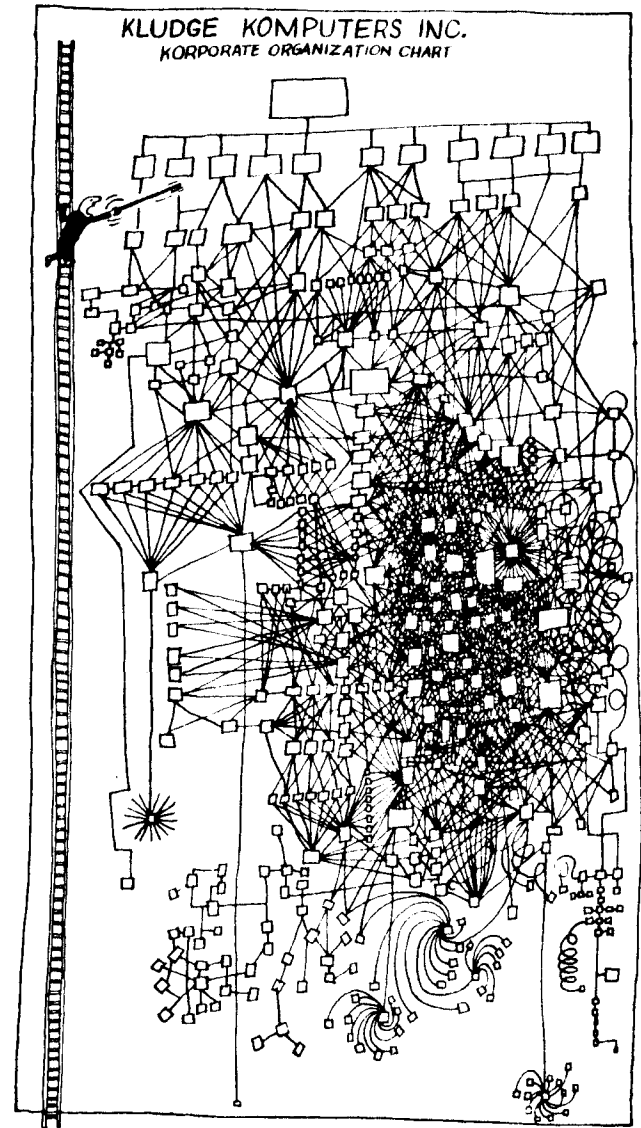
KLUDGE KOMPUTERS INC.
KORPORATE ORGANIZATION CHART

signed. That way you get yours. It's best not to enter the discussions on manufacturing volume at this time. Let manufacturing and engineering bother their heads with this small problem. Later when the costs soar out of believable proportions, produce the marketing report which will most certainly show the expected volume even higher than anticipated. Thus, one protects a high commission scale with infallible evidence perfectly timed.

the gimmick

Now that the market has been surveyed, the volume established, and the price determined, all that remains is the advertising. Before one can produce news releases, ads, give papers, etc., a gimmick is needed. The days are gone when arithmetic speed, flexible in-output, compatibility, or throughput per dollar will sell machines. We need a thing! The thing can be a simple come-on or it can camouflage some glitch in the kludge.

For instance, the KLUDGEVAC 990B has an incurable cross coupling between the load button and the tape write circuits. Every time the load button is depressed, the character under the read-write head is erased. While this might be unhandy for some applications, it could be called "Auto Message Protection" (AMP) and sold to a variety of government agencies.

Another unique feature of the 990B is Recent Rollback Record (R^3). If the master power switch is thrown unintentionally or if the power supply fails, transients cause the B to reference memory at the current instruction counter and to reset that cell by destructive readout. This may be heralded as Recent Rollback Record. The restart program has to merely determine which cells won't hold ones, compare this with the previous cumulative-to-date list of R^3(s) and the first difference is the contents of the instruction counter at the instant of power failure.

In addition to emphasizing these unique features on the 990B, it is always advisable to contribute to the glossary of the field. This is preferably done by taking some familiar, prevalent concept and establishing a synonym for it. The com-

petition is already entrenched in the input/output area with such terms as block, record, item, unit, and message. There is even some disagreement over such terms as index register and B-box. This obviously is bigger than the marketing division, and hence a Management Policy is called for.

This is a particularly fertile area for management participation as they probably can't define the accepted terms in the first place. from such a series of meetings you will have the opportunity to meet the new members of management council: the Razor Head Auditor, the Kiddy Games Planner, and a Ball Point Pen Comptroller. From such a collection of advisors, the sales plan will gradually emerge. The device which contains all of the operator buttons, switches, lights and panels shall be known as the Associated Display Device of Elementary Registers (ADDER). The device connected between the mainframe and the individual input-output units from which source data is read from or punched on 80 column unit record shall be known as a Complete Assembled Record Distributor (CARD). The device which places symbols on a sheet of paper in a line by line fashion shall be known as a Writer Of Record Data (WORD). We now have a full product line!

announcement

Timing an announcement for a major press conference is most important. Do not plan it for a Sunday. Late Saturday afternoon is best. Invite the secretaries. Invite the salesmen. Invite the president of your company to sing the Star Spangled Banner. At KKK the vice president for marketing is always on hand to explain the wonders of software systems.

Most important, an announcement should be bold and decked with the prickly rose bushes of tomorrow's world as directly influenced by the 990. Applications such as simulating rattlesnake brain envolvements, varying the parameters in sausage recipes, and pipe organ effects uplift the interest factor among news beagles and other hounds.

Equally most important, don't invite questions. If the salesmen attend, plant a few questions such as, "Will this ma-

chine truly stop the Commies?" and "What does the magic brain think about capital punishment?"

When demonstration time rolls around, explain that the machine is battery operated and ask for volunteer batteries. Show movies and keep the secretaries wandering around the room with bottles of good cheer.

Laugh it up in a cynical fashion. Press people are very cynical.

the sales staff

Now that the announcement is completed and press clippings collected, the selection of a sales force must receive immediate attention. The first task in this regard is to clean house. Move the timid ones back out into the field. Let them rejoin your competitors. Since you now have a budget you may choose a few good men with contacts or many neophytes and let the customers train them. A compromise that is often tried is to hire the Disgruntleds back from the competition. This plan saves all those travel and training dollars. You can fill in the weak spots with recent college graduates or engineering throwbacks.

If you insist on daily call reports, they won't loiter in the office learning the product line. By getting them out of the office and on calls, they start making contacts from the day they come on the payroll. The present users are the most obvious first calls to make. They just may be unhappy enough to re-program. In addition, they are the most knowledgeable and hence the new men can learn the vocabulary "on the job."

appeal inducers

Also responsible to the Vice-President-Marketing are certain projects classed as appeal inducers. Our Sign and Slogan Group came up with the startling—

The Gimcracks and Googaws Section is working on a client's office mobile which depicts the balance of time and space over a cascade of buffered input-output units. The Social Impact and MATH (SIAM) Department is working on a new user language which offers some astounding breakthroughs. It is an

amalgam of Machine language, Algol, Generators, Interpreters, and Cobol.* It takes in absolute code in binary, hates formulas, puts out 400 cycle 220 VAC, and recognizes French and Russian verbs. It has also transposed the Washington Post March into a minor key and output as Cobol narrative a piece known as Phillips Forever, or BEMA Valentine. All in all we are very proud of it.

Early reports indicate that it slows down the KLUDGEVAC 990 so much that our net income will double for this fiscal year from increased second shift rentals alone. We at KKK are very excited about the few minutes which were spent with the programmers, metalinguisians and associated hangers-on. These sessions have done more to mask out the glitches in the basic kludge than we could have obtained with next year's engineering budget. Of course, the customer pays a slight performance penalty, but one can't have everything.

Since we don't propose to furnish machine manuals, training, or a one-for-one assembly, we feel that all of our customers will use MAGIC exclusively. So sure are we of our ability to meet our commitments that we have a standing offer of double your money back (plus a dead chicken) if we fail to compete throughput-wise with machines half our size.

This particular department is backlogged pretty bad right now, as the impact of our software is tremendous. Not only is our own MAGIC language gaining popularity but our sales force has committed us to integrate PACT, BACK-ACHE, CAGE, FORTRAN, SURGE, SPEEDCODE, COM-TRAN, JOVIAL, UNICODE, FLOWMATIC, MAD, MATH-MATIC, AND NELTAC into one grand system similar to S.O.S. This meld will be known as PBCFSSCJUFMMN. Once this is put on the system tape, our crew will be free to work on a higher level POL as PBCFSSCJUFMMN will be coded in its own language and hence is selfmaintainable by customers without ancillary documentation.

the outlook

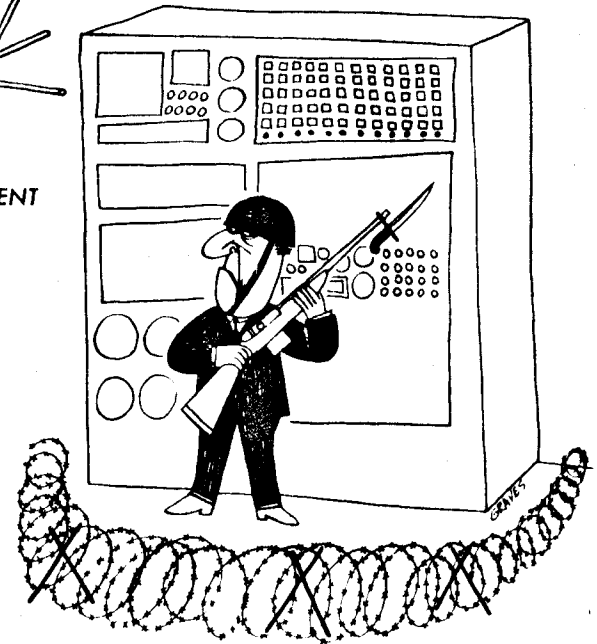I must admit that we of KKK have not been too elated over our penetration with the KLUDGEVAC 990A. Since we

have had 990A's around for 7 years, and our only sales have been to Federal Agencies (NASA, USDA, IRS, CIA, and DTMB), we are somewhat at a loss to explain our lack of commercial civilian sales. We hope that the 900 series and in particular the subminiature, petrified state, thin films, 994 (which is orderwise compatible with the 990) will raise our penetration to .003%. This is our classical share of the market and we hope that our model 904 (with the disposable mainframe) will allow us to retain it in the face of severe competition.

* We know this spells MAGIC, but we prefer to call it STUPENDOUS.
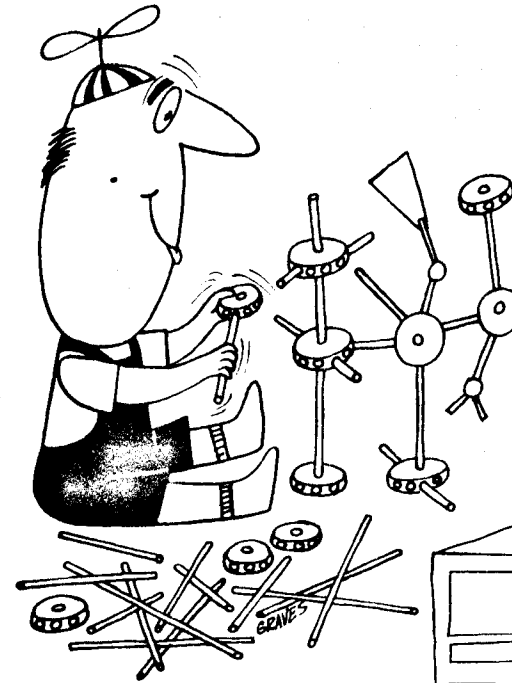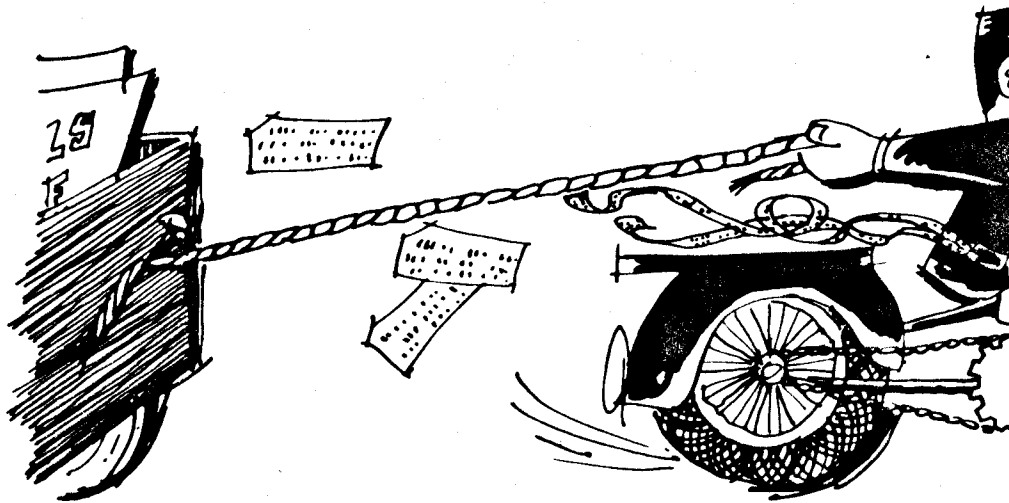
THE FIELD ENGINEER

...AS SEEN BY MANAGEMENT

...AS SEEN BY THE PROGRAMMER