DISTRIBUTION ROUND-UP: SITE ME A PRICE

Everywhere we turn users are asking for site licensing, and vendors are sighing about it. It's not that big an issue, except to people who are used to living by rigid price lists and discount schedules. Like any other contract, site licenses are negotiable, and it's easy enough to specify limits to the "site." Site licenses are merely a form of one-on-one discounting, just like the special-bid contracts vendors are used to making with distributors or large chains. (The only real issue is implementation; see next page.)

Yes, the advent of site-licensing is inexorable, but everyone is delaying it as long as possible. Talk to any right-thinking software vendor, and he'll assure you that he sells only through dealers, now and forever. And for the moment at least, micro software companies are catering to dealers.

But the fundamental laws of economics militate against this situation. Vendors promote directly to large accounts, and they support them directly. Often the presence of the dealer who collects the commission is simply a formality (although frequently it isn't). In cases like these, there's no justification for the retailer to receive his cut, except for a sort of reverse featherbedding. Long run, there's just no way this practice can prevail — although pity the first poor soul who attempts to change it. Even industry leader Lotus has bowed to dealer pressure and handed all its corporate accounts over to dealers, as have Samna, Ashton-Tate and MultiMate. The only organization powerful enough to resist successfully — beyond some mainframe carpetbaggers such as Cullinet and (unsuccessfully) MSA -- is IBM.

Get any of these vendors behind closed doors and they'll readily admit that it will be different in a few years; just don't ask them to make it happen. The other guy can lead the charge.
SOFTWARE PIRACY AND PRICING

But the issue is not just pricing or dealer commissions. Vendors' discomfort with site licensing comes also from the mere thought of customers duplicating their own software, turning it into an intangible of dubious worth, invalidating the packaging the vendors spent months and thousands of dollars to create, and, worst of all, ceding control over the creation of copies.

This brings up the whole issue of piracy and protection, which occupied a good half of all the micro-oriented sessions at the recent ADAPSO (Association of Data Processing Service Organizations) meeting in Miami. ADAPSO members are rightly disturbed at the vast though unquantified amounts of software copying going on. It ranges from the guy copying a disk for a friend, to the outfit where one of the "girls" comes in early each morning to turn on her workgroup's PCs and load each of them one by one with a single copy of a certain popular software package, to the wholesale distribution of copied software throughout an organization. Nationwide software sales are suspiciously low compared to the rate of hardware sales.

Our fix on this issue is still, as we said last summer (RELease 1.0, June 18), that the best solution is simply to sue the -- uh, jerks. But for those who aren't satisfied, ADAPSO's MWSA (Microcomputer Software Association) group is pulling together the specs of a physical solution. (For information, call Ron Palenski, Esq., at ADAPSO, (703) 522-5055.) There's a little delicacy on this issue, since the effort is being heavily sponsored by industry heavyweight Lotus, and some others complain they're being railroaded. But what do they want? They know well that it takes marketplace power, not a committee, to get such a solution accepted in the marketplace. If Lotus succeeds, the others could follow along easily. Without such a bellwether, any effort will fail.

Aside from possible marketing problems, the scheme has covered all the bases. It's small, fits into the RS-232 port (and thus fits virtually any computer), and yet has a port at the other end so the RS-232 still functions as well as ever. It allows several pieces of software to run simultaneously, since each "cradle" can hold several "keys," each unlocking a different package. Keys can be removed and kept in a pocket (or a locked drawer); the cradle itself, the size of a cigarette pack (or a mouse), can be carried around fairly easily. It's a bit of an encumbrance on a portable, but not too much. While the key's physical format is determined, the exact locking algorithms, etc., are up to the whim of each developer. There are ways to prevent the use of each key by more than one terminal (as on a network or multi-user system). Although calls to the key can theoretically be patched out, a developer could put some necessary part of the program's code into the key, making the patching job more painful than simply buying a new copy of the software. Finally, and here's the scheme's one selling point to reluctant users and dp managers, they can also use it to secure their own data or to limit mainframe access.

We're not convinced that Lotus can make the scheme fly -- not if it can't even bypass its own dealers to sell to corporate accounts (see above). And that's the crux of the matter. It's the corporate accounts that are the real copying problem in terms of lost revenues, and they're just not going to welcome a scheme of this sort. The fact remains that the scheme, one, insults its customers, and two, requires them to spend their own money ($30 or so) for the privilege of that insult. Many buyers may buy some other program -- of which there are sure to be many in response to this method -- rather than go to the trouble of buying a $30 cigarette box, even if they have no intention of stealing software. Still, it's

RELease 1.0, November 13, 1984
this kind of invention that drives people to retaliation... We suspect that in
the long run several major vendors, starting with Lotus, will simply absorb the
cost of the hardware in order to overcome at least the pricing obstacle. If the
problem is as bad as it seems to be, regained revenues would make up for the
vendor's extra costs.

Meanwhile, a number of software developers are now using schemes whereby a floppy
disk may be copied only X number of times, with some provision for "decopying." This
protection may be patched out whenever customers are smarter than developers,
but it deters all but the most crooked of copiers. The expert in this field is a
company called Softguard Systems in Santa Clara, CA, which supplies this
technology on an OEM basis to customers such as Ashton-Tate, Cullinet and Software
Arts.

Protection from price competition

The other thing that disturbs vendors is the advent of serious price competition. While fashionable commentators are telling us that our society is becoming
service-based, software vendors are diligently trying to build up a product,
commodity business. In the long run, that effort will fail, as their products
command only the prices accorded to commodities. Long run, the protection problem
will be solved not by dongles (the British term for "cradles"), but by selling
support along with, but unbundled from, the product. (Unfortunately it will be a
long time before we have genuinely support-free products.) Thus vendors and
dealers who are now supporting large accounts by selling to them and supporting
them after the sale will charge for those services; commissions or margins on the
sale of commodity disks will provide only minor returns for minor effort.

Currently a rash of high-priced new products and dealers' desire for high price
points is shoring up software prices, with the standard price for a hot new
product at $495. However, we expect to see prices of new products dropping more
quickly than the five years it took WordStar to sink from $495 to $350 as vendors
discover that "ease of purchase" helps increase sales.

Protection from competition

The current slack market, coupled with an overabundance of products and resellers,
is fostering an appalling attitude among those who consider themselves high-end
dealers. Just as dealers abhor the idea of vendors selling direct, so do many
reject the thought of distributor sales. "I sell it to a dealer and get him all
excited about carrying it, and then he turns around and finds everybody's selling
it; they can get it through Softsel," says one vendor. Yet that same dealer
complains if the product doesn't get the kind of word-of-mouth that broad
distribution encourages. The days of availability selling -- when anyone could
coin money by selling the restricted-availability, short-supply IBM PC -- are
over. Dealers must now sell service, not product. And so must vendors.

A diminishing number of vendors, notably Electronic Arts and Lightyear, is still
swallowing this dealer push for restricted availability, while Software Arts,
State of the Art, and Samna have all recently broadened their channels to include
national distributors. Ultimately, they recognize, restricted availability of
product means only that they'll find few buyers, although those few buyers will be
able to feel themselves members of an exclusive club. It's the vendor's job to
support his product and ensure its wide availability, not to keep it out of all
but the most "qualified" hands.
Price cuts. IBM's recent de facto price cuts will have the effect of causing retailers to buy more product than they need in order to qualify for higher discounts. Much of this excess product will end up in the grey market and then at discounters', dropping retail prices overall. It should be a great Christmas for consumers, but they may have trouble getting service for their equipment in the New Year as many resellers probably won't last much beyond December 31.

Channel moves. What about the PCjr in K mart, then? Is there an irresistible force pulling -- or pushing -- that product into mass-merchant channels? Many dealers' cries of outrage at the thought ring hollow, seeing that few dealers we know want to sell the product themselves. Yet, courtesy of Bill Coggshall's Software Access International (Mountain View, CA), we note an increasing proportion of families who intend to buy their pcs through specialty stores, not K mart and its ilk. This indicates that people are beginning to recognize the value of support...and that neither IBM nor Apple should succumb to the blandishments of K mart et al. if they want to maintain their reputations.

IN-CHANNEL CONFLICT

There are several components to distribution:

- product definition
- product selection
- packaging
- merchandising & promotion
- demos & other pre-sale support
- delivery
- collection
- post-sales support
- delivery of updates

And there are several players all clamoring to perform these functions:

- authors
- publishers
- distributors, electronic or physical
- OEMs
- VARs
- retailers
- consultants
- in-house MIS/DP/purchasing people
- third parties
- training companies

We don't have the graphics capabilities to draw lines between each item on the right and each on the left. But certainly each of the outfits on the right may consider itself capable of performing most of the functions listed on the left. The essence of "channel conflict" is not just conflict among channels, but also conflict over which segments of the channel each competitor is entitled to. In theory, the outfit that provides the service is the one entitled to the reward. When that structure breaks down -- as when a dealer demonstrates a product for free and the customer runs off to buy it from a low-price, no-frills discount house, or when a vendor provides extensive post-sales support for a large customer who bought the product from a retailer -- the conflict is most bitter.
ARTIFICIAL KNOWLEDGE: GARBAGE IN, ADVICE OUT

In the midst of all the hoopla over artificial intelligence and the capability of machines to "reason," both software developers and users may miss the essence of what makes software useful. An expert system, say the textbooks, consists of (1) an inference engine and (2) a knowledge base from which it draws its conclusions. It's the inference engine that's generally considered most exciting: It reasons, good grief, just like a person. It uses if-then rules, it may be written in LISP, it's technologically sophisticated.

But the useful part of much software is frequently the knowledge it incorporates. (AI purists wince at the use of the term "knowledge base" in any context other than AI.) Over the next few years, in fact, we're going to see a lot more work accomplished with "knowledgeable" pc software than with fancy reasoning, grammar parsing, etc. Much of the value of AI software is found not in its reasoning capabilities -- if-then rules, data hierarchies, etc. -- but in its ability to codify and process common knowledge. "This is how we judge loans here at First National of Vaporville." "This is what we do when our VaporSnack sales fall off." "These are the facts we need to assess our VaporMen." (What we can't do without AI is have that knowledge do any more than what we tell it to do directly. It can't take facts and proceed independently to conclusions -- although to the user (not the builder) it certainly looks as if the software is doing so. But none of this is necessary for software to do useful work and pass on useful knowledge.)

All this is nothing new to mainframers, who have worked for years with software that codifies accounting rules, payroll procedures, budgets, etc. But such information is mostly available only on a batch, non-interactive basis to dp professionals; it doesn't have much to do with strategy and goals, but only with transactions. Now computing has broadened into a tool for managers in their management role, where it used to help them only as implementors and analyzers.

In the pc world, software generally used to mean productivity software -- one person's analytical tool; now it means a way of managing the company. People can learn from software much as they once learned from employee manuals, training classes, or even lunchroom conversations.

Rather than just a tool, personal computer software will become a repository and a transmitter of information about businesses, supplementing the two functions pcs now perform: transactions (in a small business) and analysis. Now, analysis can generate transactions, as in Trigger, or can influence decisions, reflecting the experience or policies of a company, as in Lightyear. Software will actually direct procedures and control the flow of data, decisions, and analysis, much as is already happening on mainframes but in a way invisible to end-users.

What you know, not how you reason

The two programs described below are just the beginning of this trend. Neither has "artificial intelligence" as formally defined, yet to the un schooled eye, or the opportunistic marketing guy, they have that flavor. In fact, all they have is the ability to disseminate and act upon knowledge. Rather than any esoteric reasoning techniques, Lightyear (from Lightyear of Santa Clara, CA) uses plain old arithmetic calculations; Trigger (from Thoughtware of Coconut Grove, FL) uses plain old data base processing -- calculations, sorts and compares -- with the results triggering a variety of actions. Both work primarily with knowledge -- criteria, goals, etc. -- entered by a builder-user, and act on data entered by
users or (ideally) imported from "production" data bases. The very requirement of 
entering that knowledge -- which spurs self-knowledge on the part of builder-users 
-- may be one of their most valuable attributes, while end-users can pick up 
guidance, folklore, and company policies through these packages.

Both Trigger and Lightyear are unique, but fairly easy to imitate -- now that 
their developers have used their imaginations. Both will have to execute their 
marketing plans superbly, and rapidly establish both data-transfer and joint-
promotion liaisons with the companies that control access to the vast amounts of 
data already stored out there: both internal corporate data via dBASE II, 1-2-3, 
and mainframe files, and external data from Dow Jones, Data Resources, et al. But 
both can rest secure that new competition will have a tough time getting funded in 
this parlous market, while established companies will probably spend their 
resources either working with Thoughtware and Lightyear (if they're smart) or 
explaining to all and sundry why the concepts make no sense.

Builder-users, end-users, and knowledge

The advent of knowledge-based pc software will sharpen the nascent division 
between builder-users and end-users -- a division well-known to mainframe types 
which was hidden in the micro world because the initial users of micros were all 
builder-users. They developed their own VisiCalc models, wrote their own 1-2-3 
macros, formatted their own disks and installed their own software. The most 
visible of the builder/user micro products is Ashton-Tate's dBASE II (which in 
spirit is not a "personal" computer product anyway, but a business tool). Newer 
and notable are Symphony and Framework, as well as IBM's recent Business 
Management and Personal Decision Series, all of which are beyond the capabilities 
of the typical new end-users.

Others are likely to follow, but perhaps more slowly, as vendors realize that the 
audience for these builder products is small, and growing smaller proportionally 
as the pc user base expands to include less computer-sophisticated people. Unless 
Lotus and Ashton-Tate find some way to market runtime versions of their "inte-
grated" products -- both of which are currently overflowing the distribution pipe-
lines -- they have little hope of reaching a wide audience with them.

But whether the medium is Symphony, Framework, Trigger, Lightyear or other, yet-
to-be-developed products, the potential is great. Knowledge in software can 
disseminate company policies, rules of thumb, expense-account practices, office 
procedures, and all manner of handbook-type information. The new employee who 
used to spend weeks asking questions each time a new task came up can now be 
quickly productive, while existing employees can learn new procedures en masse and 
perform current ones with greater consistency. Indeed, a fundamental problem of 
today's service- and information-based society is how to maintain consistent 
standards of behavior; knowledge-based software can help do so, whether it's 
complaint-handling procedures, loan decisions, inventory management, or even 
determining who gets invited to a Comdex hospitality suite. The best of these 
products will be so easy to use, even in builder mode, as to meet the promise of 
"Every user a builder!" So far, only Lightyear qualifies. On the other hand, 
there are a lot of so-called expert systems around which are really nothing more 
than data bases of boilerplate text which is selected and displayed appropriately 
according to a user's responses to questions or prompts. Such systems include the 
Human Edge series (RELease 1.0, Oct. 17, 1983) and Thoughtware's own Expert 
Manager/ManageMentor. These products may be useful, but they don't think.
BRILLIANTLY NAMED, Trigger ($495, through retail channels and direct) attempts to help people manage, rather than analyze, their businesses. Like a lot of management software, its power in part consists of its ability by its very existence to force people to examine their businesses, determine the salient indicators, and monitor them. As Thoughtware president Jack Levine points out, "It's easy to learn the program, but it takes several hours to set it up" — determining which factors you wish to monitor, by how much they should be off to "trigger" a response, what the causes and appropriate responses might be, and so forth. A party-planner, for example, might monitor responses to his invitations to a Comdex party. Initially, he would tell the system what level of responses he expects, within a range. He could then set the system up so that below-plan RSVPs would trigger a memo to the sales manager to ask his salespeople to make a barrage of phone calls; above-plan responses would invoke a memo asking the planner's assistant to call the caterer to order more food.

In that way the program encourages interaction among the people within a business. Later, it will demand feedback noting the results of the actions it suggested. The program then builds up a file of causes, actions and results, enabling the manager to fine-tune his action plans as experience accumulates.

Trigger is a nice first step in a category we expect to become increasingly pervasive — integrated management software, as opposed to integrated productivity software. (Integrated means with other software and with reality, not integrated within itself.) Trigger is meant for a business, not an individual, and it integrates business processes, not tools. Does the manager let the computer run his business? No, because he provides the initial criteria, and can modify them as he sees fit. The results of actions also modify the files, so that probable causes of a slippage may be reordered. (And yes, above-the-mark results could trigger a congratulatory note or even a dinner invitation.)

Clearly, Trigger begs for third-party support in the form of templates, with the measures and ratios considered acceptable by experts in any given field. There could be a restaurant template in fast food, standard and gourmet versions, a hairdresser template with measures such as cuts per hour per employee and ratio of hair patients persuaded to have a manicure, publishing templates with subscription figures and renewal rates, and the like. Such software is not just a tool, but a medium for dissemination of knowledge.

Trigger also begs for better data-import capabilities, to obviate the need for periodic entry of after-the-fact, noncritical data. Indeed, Trigger's vulnerability is its lack of authority. Unlike HAL the computer or even Hal Geneen (the ultimate by-the-numbers manager), it doesn't force you to obey its commands, nor does it require you to report on your progress each week. It may help if a manager delegates the reporting task to a clerk — as long as he won't let it slip if there's some other "emergency" task that needs handling. After all, the widgets can go out, the calls can be made, the invitations to the sales seminar extended without Trigger. It may end up like most home finance software (but not Quicken; see RELease 1.0, September 11), widely bought and rarely used. Weekly reports will slide and become monthly and then quarterly as people manage to conduct business without Trigger. Yet even if Trigger is used only to the extent of getting the manager to sit down and think about what it is that triggers his business, it will have served at least part of its unstated purpose.
LIGHTYEAR: VISICHOICE

A recent study by Jeffrey H. Moore, sponsored by Epson, showed that most users don't use decision support tools to make decisions, but rather to implement and monitor them. In the same way, we suspect that Lightyear, a decision-modeler, won't so much help people to make decisions as to explain why they've done so. It will help them to analyze their thinking, to understand their own criteria, and to communicate them to -- or impose them on -- others.

Lightyear reminds us of VisiCalc in many ways: It's startlingly obvious -- once you've thought of it -- and thus tremendously vulnerable to being "flattered" by imitation. (The $495 price may be a bit high, considering the company's need to get itself a large, loyal user base before competition can arise.) For its time, Lightyear is amazingly easy to learn and use. We sat through half a demo, started playing with the product ourselves, and gave a demo to someone else who came in early for the next appointment. Unlike Trigger, Lightyear is actually fun to use, quick and interactive; we wanted to keep on playing with it rather than give it up to the newcomer. It was probably the same way with VisiCalc for number-crunchers, but Lightyear appeals to people who never handle concepts more exact than low, medium and high, or good, great and lousy.

The product works like this: You're trying to decide among, or just rank, a number of items. You determine a number of criteria (diagram A); in a job search, for example, you might assess location, quality of management, quality of work, salary, equity, company prospects, job title. You would weight each of these criteria, perhaps indicating in some cases (with a "rule;" diagram B) that if salary is below a certain level, that job is automatically out of the picture. Next you go through and assess each choice on all criteria (rather than doing each choice at a time, which might distort the rankings). The result is a bar graph showing the weighted attractiveness of each option (diagram C). You can also compare two options in detail in a set of bar graphs, showing how they compare in each of the criteria (diagram D). Criteria can be shown graphically (diagram E), and a company's detailed ranking on each criterion can be displayed (diagram F).

Now comes the fun part -- the what-ifs and the updates. What if you decided location weren't so important after all, and reduced its weighting? Suddenly the bars are redrawn and the options reordered. Suppose you added another criterion, such as related job opportunities for your close companion? Suppose you suddenly get a better offer? Or suppose you reassessed Company Y's management quality? You can do that easily enough; the value of this program is that you can't do so without noticing. Maybe you're a manager assessing promotion candidates; do you have to keep fudging to stop the women from walking away with all the plum jobs? The program won't stop you from fudging -- but it will draw that practice to your and your management's attention.

Lightyear can be used for everything from job candidates to loan approvals, site evaluations, choosing among courses of action, story ideas, marketing plans -- even former Apple Lisa manager John Couch's recent decision to join Lightyear as president. But it's far more than just a personal tool. It can transmit knowledge -- a corporate culture or policies -- either in loose form with criteria defined but evaluations and weightings left up to the user, or in strict form, with everything but the specific assessments of individual options defined. Meanwhile, the criteria don't have to be subjective; they could be things such as the prime rate, the level of housing starts in a local market, the price of soybeans, loss experience ratios, D&B credit ratings, or grade averages.

Release 1.0, November 13, 1984
CRITERIA | MODE | HEIGHT
--- | --- | ---
Annual Sales $M | N | 50
Mgmt. Quality | V | 80
Public Image | G | 70
Balance Sheet | G | 60
Expansion X | N | 90
Antitrust Prob. | V | 55
Price $Million | N | 80
Availability | V | 25
Wall St. React. | V | 80
Profit Margin X | N | 70

F1=HELP F2=ENTER VALUES F3=MAIN MENU F4=DELETE CRITERION

CRITERIA | MOST DESIRABLE | LEAST DESIRABLE
--- | --- | ---
Company A | | |
Company B | | |
Company C | | |
Company D | | |
Company E | | |
Company F | | |
Company G | | |
Company H | | |

F1=HELP F3=FINISHED ENTERING VALUES

GROWTH RATE
| Expansion $X MUST BE GREATER THAN 8 (ELIMINATION RULE) |
PRICE | Price $Million SHOULD BE LESS THAN 350 (HEIGHT = 70) |
Mgmt. IMAGE | IF Mgmt. Quality LESS THAN Above Average THEN Public Image MUST BE GREATER THAN 90% (ELIMINATION RULE) |
WALL STREET | Wall St. React. MUST NOT BE LESS THAN Fair (ELIMINATION RULE) |
BALANCE-PROFIT | IF Balance Sheet LESS THAN 50c THEN Profit Margin $ SHOULD BE AT LEAST 5 (HEIGHT = 40) |
SIMILARITY | Corp. Similarity SHOULD BE GREATER THAN 80c (HEIGHT = 35) |

F1=HELP F2=CREATE RULE F3=MAIN MENU F4=CHANGE/DELETE RULE

SUMMARY EVALUATION: CRITERIA AND RULES

RANK | SUMMARY EVALUATION: CRITERIA AND RULES | SCORE
--- | --- | ---
1 | Company A | 940
2 | Company B | 865
3 | Company C | 841
4 | Company D | 823
5 | Company E | 803
6 | Company F | 794
7 | Company G | 771
8 | Company H | 769

POSSIBLE = 1160

DETAIL EVALUATION: CRITERIA AND RULES

Company A | 940 | 1160
Company B | 823 | 1160

Earnings Rise $X

Expansion $X

Mgmt. Quality

Wall St. React.

Public Image

Simillarity

F1=HELP F2=ENTER VALUES F3=MAIN MENU F4=SUMMARY EVALUATION

ALTERNATIVES | LEAST DESIRABLE | MOST DESIRABLE
--- | --- | ---
Company A | | |
Company B | | |
Company C | | |
Company D | | |
Company E | | |
Company F | | |
Company G | | |

F1=HELP F3=FINISHED ENTERING VALUES

Balance Sheet $M

Corp. Similarity

Antitrust Prob.

Annual Sales $M

P/E Ratio

Price $Million

Cost Savings

POSSIBLE = 1160

F1=HELP F2=ENTER VALUES F3=MAIN MENU F4=SUMMARY EVALUATION

Release 1.0, November 13, 1984
Or Lightyear can be used in reverse, to let users customize various data bases, where data on specific items is the knowledge to be constructed by the builder, and the end-user adds his own criterion assessments, rules, and weightings. For example, suppose someone were to evaluate the upcoming round of Comdex events. The criteria would include food, topic if any, quality of people, entertainment, location, time and day, presence of other events in the same hotel. The evaluator would ship the data out on a Lightyear disk, and the user would customize it, weighting and ranking the criteria according to his own needs: He's allergic to seafood, so the presence of seafood doesn't score with him. Nor does he care for dance bands or comedians, although he's nuts about the Beach Boys and Linda Ronstadt. He's staying at the Flamingo Hilton, so he orders the ranking of the various locations to take that into account (although the combination of the Hilton and a time during or close to show hours in the hall adjacent almost always ranks favorably, as reflected in a default rule supplied by the evaluator). He likes communications, is bored by micro software (except Lightyear), so he ranks the topics to reflect that prejudice. And he's leaving on Friday morning, so he adds a rule: "Day must be less than Friday." Presto, he has an event ranking tailored to his needs.

Class of Fall Comdex '84

Lightyear compares and contrasts easily with Analytica (page 11), this Comdex's other hot start-up. (Thoughtware has been around for a while, and is partially funded by Alexander Proudfoot Companies, a large, private consulting firm.) Both are Regis McKenna clients; both operate in the Bay Area; both are trying to create a new product category. Both need to get established quickly to forestall competition. (Lightyear already has some in the form of Expert Choice from Decision Support Software of McLean, VA, a product which somehow lacks that critical mass of charm, simplicity, and enthusiastic supporters that is attracting funding, customers, and perhaps eventual success to Lightyear and Reflex.) But Lightyear has total funding of about $1 million, compared with Analytica's $5 million (and $3 million more in prospect).

But Lightyear has 12 people while Analytica has 40. While Analytica plans to ship in January with the product widely heralded by ads in all the print media, Lightyear is already shipping and relying so far on highly favorable word of mouth. Analytica is using distributors as well as retailers to distribute the product; Lightyear is relying on a handful of key dealers. Reflex has the team polish of a 15-person-year development and QA effort, while Lightyear has the personal sparkle of a one-designer product.

We could hardly think of a better controlled experiment to observe...although as usual the results will depend more on implementation than on strategy.
DOWN THE BLACK HOLE OF DBMS

The software industry's version of the Bermuda Triangle is surely the black hole that lies between dBASE II and pfs:file/report. Teetering on its edge are a number of worthy products such as Power-base and the Concentric Information Processor. dBASE II and pfs have their flaws, but they've been around long enough to fix up their bugs and to attract a loyal corps of users who pass on favorable word-of-mouth. But that black hole still seems irresistible, as demonstrated by three new contenders. Two are sidestepping the black hole, Analytica/Reflex by

<table>
<thead>
<tr>
<th>USER</th>
<th>pfs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2-3 Reflex</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SPREADSHEET</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Symphony</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BUILDER</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MacLAND</td>
<td></td>
</tr>
<tr>
<td>ProVUE</td>
<td></td>
</tr>
</tbody>
</table>

offering a cross between a dbms and a spreadsheet that's off the line between pfs and dBASE II, and ProVUE/OverVUE by working in another dimension -- the Macintosh. But Infocom, with its new Cornerstone, is staring right into the black hole, clutching onto an extremely straightforward but powerful interface in an attempt to keep from being sucked down.

REFLEX: MAKING SENSE OF ROWS 21, 27 & 22, COLUMNS 44-71

Reflex, the "analytical database" from Analytica of Fremont, CA, is primarily a personal tool. Although it can and will be formatted by builders to analyze data according to company models, its value lies in its flexibility more than in its ability to produce predefined reports and analyses. It is most likely to be used as an aid after those predefined reports produce puzzling results. Why are margins down in CleveLand? Which products generate the heaviest proportion of DOAs? Why didn't anyone attend our party?

RELEASE 1.0, November 13, 1984
The REFLEX™ Form View™ shows all the details, one record at a time.

The REFLEX™ List View™ shows the records in a column-and-row format.

The REFLEX™ Crosstab View™ provides a numeric overview for analyzing trends and making comparisons.
Where most dbmses are *repositories* for data, with some minor calculation and sorting capabilities thrown in, Reflex helps to make the data meaningful with an extremely interactive "reporting" capability that lets you twist and flex your data around, combining numbers according to various criteria, consolidating selected items from various rows and columns much more flexibly than any traditional data base or spreadsheet we've ever seen. In fact, it combines the flexibility and feeling of control of 1-2-3 with the data handling and sorting capabilities of a data base.

The data can be viewed in a variety of ways (see illustrations): lists (tables -- columns are fields; rows are records), forms, reports (not shown), graphs, and cross-tabs, a facility well-known to market analysts and sold elsewhere as multi-dimensionality in a spreadsheet. Three views may be shown at once in separate windows -- for example, a form with full data for a single record, plus a graph with that data point highlighted on it, plus a cross-tab of the two axes being highlighted.

Suppose you have a listing of trade-show parties, with the number of guests invited, the number responding yes and no, the number actually attending, food or other attraction mentioned in the invitation, cost, and time of day, and day of trade show (opening night, first day, second day, last day). This you could easily put into a data base, and determine the total number of guests, the average number per event, the count of events, even the minimum, maximum, variance and standard deviation of the number of guests per event. You could also automatically calculate cost per guest, ratio of no-shows, etc.

But now you'd like to have some *meaningful* data. Are there any correlations? Do breakfasts pull better than cocktails? Do people tend to fade away after the first night of a trade show, or on the contrary do they not show up until the middle? Using cross-tabs, you could plot the no-show ratio against the time of day. Or you could plot the no-show ratio for each event, ordered by time and day. Finding an anomalously high point on the graph, you could select it with the cursor and zoom in to the data for that particular event. Depending on the completeness of your data base, you might see that it was scheduled against a party by Microsoft or Lotus, or perhaps one of those fierce Las Vegas storms.

Reflex can determine the count, average, sum, minimum, maximum, variance, and standard deviation of any set of numbers. These can then be displayed as the raw data, as an index, as a percent by row, a percent by column, or a percent of the total shown at the lower right, which appears as 100%. Additionally, Reflex can easily generate information -- incrementing a hypothetical price by $50 from $1000 to $2000, for example -- for automatic what-if analysis. A cross-tab view might show a series of calculated fields for each of those price levels.

Reflex works entirely in memory, just like Lotus 1-2-3. That's entirely appropriate for a *personal* product, although it limits the product's usefulness when it comes to analyzing mainframe-size volumes of data such as the market-share reports many of Reflex's users might have. (Those people would be better off using a Metaphor system; RELease 1.0, September 11.) At the moment Reflex can import data from Lotus 1-2-3, dBASE II, pfs, DIF, and ASCII files, and thus indirectly from a mainframe via Informatics' Answer/DB and other query tools, all of which are a little cumbersome.

To appreciate Reflex best, one has to be familiar with both spreadsheet and dbms -- and wish one could combine the best features of each into a single product.

RELease 1.0, November 13, 1984
But to the first-time user, Reflex may seem hard to learn and unfamiliar. Besides, once he's learned it, he'll say, "Well, isn't this what software is supposed to do? It makes everything clear." Experienced users are more likely to be properly grateful.

The $495 pricing too strikes us as expensive. Although Reflex does provide currently unique capabilities, they won't be unique for long if the price keeps Analytica itself from rapidly establishing an installed base, and provides too much of an umbrella for competitors. For the moment, Analytica's position is ideal, since no VCs are likely to fund a competitor. The company had a fairly easy time raising its initial $5 million (from Merrill Pickard, Institutional Venture Partners, Arscott Norton and J.H. Whitney); but the market is tougher now. It is selling the second round at a lower valuation and only on the premise of an existing product with no direct competition.

OverVUE: MAKING MAC MASTERFUL

OverVUE comes from ProVUE Development Corp. of Huntington Beach, CA. It's a (substantial) rewrite of DataVUE, ProVUE's dbms for the 68000-based Alpha Micro line -- where it has been successful with the sale of 250 copies and maybe a fifth of that small market. The numbers are a little higher in the Macintosh market, and the prices are lower -- $295, vs. $1000 for the Alpha Micro product. So far, ProVUE has shipped about 3200 copies to dealers and distributors (Softsel and First Software), in addition to 5400 distributed through Apple's sampling program.

OverVUE would be a fine product in any environment, but it positively shines within the somewhat small and unsatisfying universe of Macintosh software. It has the flavor of Reflex, and even does cross-tabs, but in a more cumbersome way. (First you create summaries; then you delete the records that have been summarized. You can't quite get the wonderful push-of-a-button feel that Reflex offers.) On the other hand, OverVUE does have that incredible Mac ease-of-use. Its documentation, a 144-page manual that uses a long-distance phone bill for a sample analysis, is clear and well laid-out, averaging about one screen illustration per page.

Since the Mac is 68000-based like the Alpha Micro, the job of porting the functions was easy, but developing the user interface was a little tougher. And OverVUE went all the way, rather than simply sticking a couple of mouse-menus on top of what it had done for AM. Nonetheless, while most of the work was in the calls to the Mac interface, much of the function is in ported assembler code -- enabling the program to run faster than anything else we've seen on the Mac. Certainly it's far more useful. Release 2, due next summer, will include graphics capabilities (not available in DataVUE).

ProVUE was founded in 1978 by Jim Rea, formerly a programmer with Alpha Micro, and is internally funded. Marketing director Marty Rezmer joined the company after seven years as owner and manager of the thirteenth Byte Shop (since renamed).
CORNERSTONE: ENOUGH, TOO LATE

Infocom, based in Cambridge, MA, has persuaded most people to preview its new product Cornerstone ($495) only by hiding the fact that it's a dbms. Without a doubt, it could beat pfs:file and dBASE II all hollow in any sort of forget-history context, but the market is history. Infocom has followed the cookbook of marketing, first defining its customer base -- all of us business types, as pictured on the packaging -- and only then defining the product. With funding limited mostly to the profits generated by the company's text-based adventure games (which earned profits of $256,000 on revenues over $6 million in 1983, such as Zork and the newly announced Hitchhiker's Guide to Galaxy), Infocom will initially limit distribution to the distributors that know it so well -- Softsel, Micro D, Softeam, First Software and Software Distribution Services, from whom even the likes of Businessland, Sears, ComputerLand and the classy direct-call dealerships will have to purchase the product. A $1.5 million advertising budget will aid awareness -- more than it would have last year when Vaporware was in full swing.

Cornerstone is as easy to use as it could be given that it does complex things such as joins, data verification, derived fields, and all those other wonderful things relational dbmses can do. Fundamentally, it works the way you've always thought a dbms should -- and like Reflex, it will be best appreciated by those who have struggled through imperfect renditions of the same concept.

Although it seems almost overburdened with help text, the system leads the user through any problem he may have with context-sensitive help. Any data once typed in need never be retyped; much of it is provided automatically with table look-ups (or automatic joins, in relational dbms parlance). For example, if you type in a customer's name, the system will automatically fill in from another file anything else it may know about him in the appropriate fields of that record. Data can be locked in so that, for example, a customer name may be updated only in the customer file, thus ensuring that data remains consistent throughout all files. In addition to joins to data in other files, fields can have "enumerated values," summoned by the "options" key, both to save typing efforts and to enforce consistency of phrasing ("initial meeting" vs. "first meeting") and spelling ("follow-up" vs. "followup" or "Desert Inn" vs. "Desert I."). Obviously, it takes a little bit of intelligence even to understand these capabilities -- and therefore to make use of them. But unlike most software, Cornerstone doesn't make anything harder than it absolutely has to be.

Aside from being an elegant data base, Cornerstone also understands human conventions -- befitting a company that made its fame with text-based interactive games. It has not just date arithmetic, for example, but the ability to understand "next Saturday," "noon," and "last month." It comes with a built-in phone/address list and a client-tracking file. It enables you to insert up to three pages of comments in a record -- and to search for any word in those comments (although it's much quicker to find things using indexed fields, of course). The forms, reports and file structures are easy to define, and to change after the fact.

We love Cornerstone, and so do all the resellers who have seen it. It is one of the new products that we are buying a PC AT to use. If there is anything left but a black hole between pfs and dBASE II after all these years, Cornerstone is the product that could fill it. It has everything it takes except timing.

RELease 1.0. November 13, 1984
THE INTERFACE IS FAMILIAR...

The builder/user concept mentioned above has also spawned a number of front-ends, or interfaces for applications, as separate products divorced from the applications they cosmeticize. But their impact is more than cosmetic. They can make an almost unusable product usable, or accessible to much larger numbers of users. They can make its use faster and easier. And most can enhance its capabilities by defining series of functions ("macros" or "procedures") to be executed automatically. (Is mail-merge a front-end combining data base and a word-processor, or is it a separate application?) Most front-ends come in two forms, as interfaces for existing products (WordStar, Lotus 1-2-3 and dBASE II, usually) for use by end-users, and as front-end generators, enabling builder-users to create custom interfaces and applications for end-users within their companies or resale customers.

Front-ends are a way of making it easy and simple for end-users to generate the commands and data that drive any application. To many users, those commands are hard to remember and impossible to guess at or find in a manual. Moreover, they may not know what data the computer needs to perform its tasks, or in what syntax. The sequence of commands itself can be easier or tougher to remember (an application itself is only a front-end to a system of calls to a programming language and operating system), but most people by now agree that it's cruel and unusual punishment to require any user to run an application only with commands.

Many front-ends will be on display at Comdex -- and even on a few shelves thereafter. Over the next few pages we consider a variety of front-ends, using both menu selection and "natural-language" input.

Menus

One elementary interface is simply a series of menus that lists all possible commands and lets the user select the appropriate one. A more complex interface combines commands. (Internally, of course, each menu choice generates the proper sequence of commands, which may result in a prompt for data or display a second menu if further input is required.) The menus save the user the trouble of remembering and then typing in commands. And then there are top-line menus, pull-down menus, pop-up menus, and so on and so forth -- all of them merely, dressed-up menus. Another now-familiar front-end is the query-by-example form whereby one fills in or queries a data base -- station: WBZ; time of day: greater than 6.45; programs: all; ratings: all -- which translates into data base commands such as "select from program-log all programs where time-of-day is greater than 6.45-pm station = WBZ and all ratings from network-analysis where program.program-log = program.network-analysis." The system, of course, knows what fields exist in its files, and displays them for the user to filter from.

Indeed, a cardinal rule of interface design is that if the information is in the machine, the user shouldn't have to type it in, but merely select it. Of course, this rule breaks down at the extreme: A frequent user may prefer to type in a brief command rather than select it; good systems can make the act of selection the same as the command (e.g. "Type control and then select the command by typing its first [highlighted] letter."). Also, in the far future, when expert systems abound and software may "know" so much that it's impossible to display it all on the screen, it may make more sense to have the user define his request. But the machine should also know enough about the user that it can tailor a menu to his needs and interests.
Natural language

Most complicated to devise, and theoretically the easiest of all to use, is natural language: Tell the system what you want, and it will do it for you. There are three problems: You may not know what you want, or that it's available; you might like to know what our choices are. "Anything" is rarely a spur to useful communication. Two, you may not be able to communicate with the machine successfully despite its "natural-language" interface. Three, the machine may not be able to do what you want.

Most natural-language interfaces are akin to a dark room. They provide an illusion of space, or boundless understanding. True, you can't see the boundaries; they may be infinite for all you know. The trouble is, you can't see where you're going, and you may easily walk into one of those boundaries, especially if you're near the wall and you walk towards it. (Natural language may make sense once software can truly do almost anything, including figure out what you want before you do.) With a menu system, you lose the feeling of boundlessness, but that's only an illusion anyway -- and sometimes a bewildering one. You can see the system's limits, but you can also see all its capabilities -- and you have a better chance of walking in the right direction.

Thus we find menus that guide the user's communication a fundamentally better tool than natural language. In either case, of course, the concept of menus is a small part of the implementation; a poor implementation can far outweigh the value of any interface "concept."

TI's NaturalLinks

Despite the issues we raised, people still like that illusion of natural language; it feels "friendlier" than some cryptic menu selection. Of all the menu systems, we think TI's "prompted" NL NaturalLink system -- basically, NL menus -- is the winner conceptually. Unlike a hierarchical menu system, it lets you see where you are, by showing you a fair amount of the room. Yet it's not a static menu. It lets the user build a natural-language query with minimal typing and no guessing by selecting sentence fragments from a screenful of them. The interface knows what constitutes an allowable query, and so the fragments available change as the user proceeds through the selection process, and he automatically ends up with a legitimate command or query. He can do what the system lets him, and nothing else. This, of course, is the same as with any other system, except that in those others he can make mistakes.

However, because most of the interfaces work only on top of nine popular applications (see box on page 18), the best vision of the system is in TI's Access to Dow Jones News Retrieval and Access to MS-DOS NaturalLink package, developed in conjunction with Dow Jones and Microsoft, and tightly coupled to the underlying programs. TI's goal is to work more closely with its OEM application vendors to get around this problem. Moreover, the TI NL interfaces take over the whole screen, tearing the user away from the application he's actually working with. There's no reason, however, that a partial-screen custom system couldn't easily be developed with the developer system (price $8000), although that would sacrifice some of the visible depth of the menus.

So far, TI's NL system works only on the TI Professional. They and the developer system could probably find a far larger market if TI management were willing to relax this constraint.

RElease 1.0, November 13, 1984
THE PROBLEM WITH INDEPENDENT INTERFACES

The problem with the independent add-on approach is precisely its independence. As long as the interface operates independently of the underlying application, it can only send it commands almost in a batch mode; it can't really interact and synchronize with it. Thus you generally can't have the nice effect in some Mac software where certain item menus go grey when the user can't use them (a line can't have a font, for example). The interface doesn't know where you are in the program. The most obvious example is products like Polytel, where the interface is fixed (printed on a template). You can use it to send instructions, but it never changes in response to those instructions -- although it could, of course, cause the application to display, for example, a list of files to be selected from. Meanwhile, the application's original interface peeks through underneath -- a), Lotus menus, or whatever. This is why, even for so non-invasive an interface product as TopView, it will be hard to write good software that supports TopView without being dependent on it. The state of "TopView-awareness" implies an extremely limited use of TopView's facilities.

Of course, the divergence problem can be gotten around, but the effort is complex. Trillian's shell for DOS, WordStar, and 1-2-3, for example, senses when the user exits Lotus, even if he does so by selecting the 1-2-3 exit command rather than working through the shell, so that it won't display a 1-2-3 menu window while the hapless user is wrestling with the intricacies of DOS. But it's easy enough for an aggressive user to get his package and its interface out of synch. The menus will, of course, respond to the user's interaction with them, but if they then issue an inappropriate command because the user has also interacted directly with the underlying package, difficulties can arise. And these programs are designed especially for the kind of users most likely to be totally bewildered by such an experience.

There's a marketing problem in trying to couple interface and applications too closely; ISVs don't generally welcome the efforts of such third-party vendors. However politely, the mere existence of such a front-end implies a flaw in the underlying software. That implication is frequently true, although it may just mean that certain people prefer certain interfaces. But, aside from a few features here and there, all most commodity products have is their interface. Yet Polytel, which provides generic templates entitled WORD PROCESSING and SPREADSHEET that run on top of a variety of wp and spreadsheet packages, is particularly unsubtle in delivering this message. The company's marketing materials point out that once you're familiar with its spreadsheet template, you can use any supported spreadsheet without having to learn it.

Clearly, as software gets better and as more ISVs write their software using such interface builders as TopView and the Macintosh environment, there will be little need for such add-on interfaces. Long-run we expect to find interfaces incorporated into a product by the developers. Many will adopt one of the standards -- TopView or Macintosh -- or do a good enough job themselves that there's no room for a third-party add-on. Both MicroPro and Lightyear make a point in their marketing materials of expressing their support for TopView and Macintosh, while Analytica's Reflex and Infocom's Cornerstone could hardly be improved by any add-ons. Each presents its underlying features as clearly as possible; further interfacing would only gum up the works. While they too will probably run under TopView, they will probably use it more as an environment -- to run in conjunction with other products -- than as an interface design.

RELease 1.0, November 13, 1984
Clout extends its reach

Microrim's Clout, the natural-language query program that has helped spur the success of Microrim's R:Base 4000 relational dbms, is now available in a new release "designed to work with files from many popular products" such as Lotus 1-2-3, VisiCalc, Multiplan, dBASE II and the pfs: series. Or rather, Clout 2 ($249) now includes a file translation capability, File Gateway, so that the output of those files can be converted into Clout 2 files and queried by the program. The distinction is fine, but it means that you can't just get at your spreadsheet files automatically, but must transfer them into Clout 2 each time you want to use them (unless you haven't updated them in the meantime). That's a minor inconvenience -- about 25 keystrokes, estimates Microrim chairman Wayne Erickson. We imagine Clout 2 will win Microrim a lot of new customers. The great benefit of the program is that you could, say, combine inventory files (dBASE II), customer files (pfs:file), and sales forecasts (1-2-3) in order to find out the names of all salespeople for customers who've ordered out-of-stock products in sales territories where sales are under budget -- a clear sign of planning problems.

Suddenly the capabilities of a relational data base manager -- to combine records from different tables by "joining" records with common entries -- prove their full power. (Yes, the user does have to tell the system that "customers" in one data base equals "purchaser" in another; but he'd have to know that much to make much use of the data in any case.) In fact, Clout 2 has all the file-joining and most analysis capabilities of a full relational data base management system, although it lacks the data entry and updating capabilities of R:base 4000 or other dbmses.

Clout 2 remains a query-only system -- you can't manipulate the information once you've received it -- but you can load those answers into a file and then into another program -- a word-processor with mail-merge to write warning memos to those delinquent salespeople and their managers, perhaps.

The major reason Clout can interface only to non-query applications is that query is fairly simple, and requires a limited number of verbs and commands supplemented by nouns from the user's personal dictionary (or other file) and their synonyms. The vocabulary (and syntax) required to manipulate an application, or to enter data, is far more complex -- and ambiguous.

Indeed, we think that the true application of natural language ultimately will be the creation of order from disorder -- the ability to feed a machine competitive sales reports, for example, and have it produce a competitive analysis. (More on this next time.) Now that's a lot more interesting than simply being able to ask, "How do we compare with the other guys?" instead of "Show comparisons, all."

Polytel

Polytel of Sunnyvale, CA, has just announced Keyport™ 300, an off-screen interface which operates on top of a membrane pad that sticks into the game port. In essence, Keyport 300 is a macro-generator with 300 function keys. The user system ($195) uses a single "spreadsheet" and a single "word-processing" overlay ("Keyware," $79 each); they work with Lotus 1-2-3, SuperCalc, VisiCalc and Multiplan, and with WordStar, pfs:write, MultiMate, and EasyWriter. (Yes, some functions of certain packages are left out.)
It lets the user see all of what's available to him all the time, even though it
doesn't combine it into English sentences. With the builder system ($109), the
builder-user can use the key definition procedure to customize the preprogrammed
templates with a few macros of his own, or he can create his own set of
procedures and design his own overlay on extra sheets provided by Polytel.

The system makes sense mostly for people who type so slowly or use the system
only casually, so that use of a supplementary "keyboard" is no disadvantage. We
suspect it will be far more successful as an application-generation product for
use by corporations or VARs for tailored applications. (The Keyport
300keyboard, with its different-sized keys created by assigning multiple keys to
the same command and letting them look like a single large key to the user,
looks exactly like those customized keyboards showing up more and more
frequently in fast-food restaurants.) Indeed, Polytel will be marketing
initially only to large corporations and VARs and even some ISVs building new
products, with a key dealer sales effort to start early in 1985.

(Koala Technologies of Santa Clara, CA, has been shipping a similar product,
SpeedKey, since September, with volumes now running in the thousands. SpeedKey,
based on Koala's touchpad, has 37 keys -- perhaps a blessing to the novice. It
also has the ability to emulate a mouse in its touch-pad mode.)

Trillian's Visuall

The compromise between the TI and the Polytel approaches -- keeping the stuff on
the screen, but only when it's needed -- is the approach of most of the
proliferating window "environments" with pop-up menus, and those systems that
offer pop-up menus alone. Their menus show up as necessary, and discreetly
disappear the rest of the time.

Pop-up and pop-down menus are the least intrusive and potentially most helpful
interface. They also require a lot more work on the part of the builder-user:
To be done right, they should be context-sensitive; the right menu must show up
with the right choices highlighted at the right moment. Such menus, being in
the software, are more easily modified than templates with printed overlays.

Among the pop-up systems is Visuall (formerly VisuALL, but people kept on
mispronouncing it), from Trillian Computer Corp. of Los Gatos, CA. Basically a
menu builder that displays its menus in pop-up windows, Visuall comes in two
forms -- Visuall Plus ($150, a runtime version including front-ends to WordStar,
Lotus 1-2-3, and PC-DOS) and the Visuall Design Kit ($200, plus a copy of the
runtime version). Founded in the summer of 1983 with a stake of about half a
million dollars from Jim Toreson, president of Xebec, the company started
shipping the product seriously in September after months of beta testing.
Without the funds (or the product) for broad distribution, says president Peter
Redford, the company has been concentrating on telemarketing and has sold about
500 copies, including about 70 designer kits, to some 70-odd companies. An OEM
agreement with ITT hasn't brought in as much revenue as Redford hoped for.

Mouse Systems

Another pop-up menu system gaining wider acceptance is Mouse Systems' Designer
Pop-up Menus ($195, down from $295, and bundled with a Mouse mouse), originally
produced as a way of selling mice. Sold at retail by Mouse Systems of Santa
Clara, CA, which has gained a substantial retail presence, the Mouse menus are
pretty much the standard menus of each application, but controlled by a mouse. They sit on top of Lotus 1-2-3, dBASE II, VisiCalc, Multiplan, Personal Editor, Volkswriter, WordStar, SuperCalc\(^3\), MultiMate and pfs:write. (The package also includes a design facility for builder-users.) For end-users, the primary concern is whether he wants to use a mouse for input, rather than the visual interface. The system can come in very handy with a spreadsheet, enabling a user to enter numbers with his right hand on the numerical keypad, and control the spreadsheet with the mouse in his left.

Mouse Systems has sold close to 20,000 of its menu packages -- probably more than all the other interface vendors totalled together. But long-run Mouse will probably return to selling mice standalone, as software vendors build in their own mouse interfaces.

**Predictive Systems: ThinkTank for developers**

Still in development is the Mediator from Predictive Systems of San Mateo, CA. Built by David Taylor, a former associate professor of psychology at the University of Rochester, the Mediator is the most flexible front-end we've seen. It acts more as a macro-generator than as a front-end to the application, and currently comes in a runtime version only (no standard interfaces), for which Predictive Systems will build customized templates under contract to generate revenues until it can get into volume production of a developer's kit.

The design goal is to insulate the user totally from the application, letting the builder develop routines using the underlying applications as a sort of meta-programming language. Thus the Mediator solves the application-interface coupling problem by trying to keep the user away from the application. (It can also operate on several applications simultaneously, as long as the user handles -- on instruction -- the swapping of the disks in a floppy-based system.)

The Mediator's development system offers a sort of high-level systems analysis to the builder, making his job easier. While he can build a procedure from the ground up, he can also work with a series of nested routines, a sort of ThinkTank for developers. For example, he may develop a routine called "start": "Greet the user. Display introduction. Ask user's name. Create a variable for the name." And so forth. Then he breaks down each of these routines in turn, until he's reached the least common denominator of talking directly to the application. When a user later types in "start," the system will run through that series of routines, greeting him in a friendly, builder-defined fashion.

The ultimate user system is command- rather than menu-driven, although the user can display his options on demand and the system provides command-completion. (As soon as he types in enough letters to specify a unique command, the system completes it for him. This can be pretty confusing if he continues typing.)

Overall, the interface generated is dependent on what the builder builds with it, and isn't much use for simple command-replacement. But it's powerful enough to build up elaborate, cross-application routines that look deceptively simple to the user. If the builder is good, the Mediator can be very, very good; if he's bad, it's horrid.

__RELease 1.0, November 13, 1984__
And, of course, TopView and Macintosh

Like the other interfaces mentioned above, TopView and Macintosh provide menu facilities — among other things. The difference is that TopView and Macintosh are development facilities which their vendors expect ISVs to use, rather than front-ends to existing programs (with some minor exceptions in the case of TopView). IBM and Apple generally have the power to persuade ISVs to use their interfaces, whereas the other companies mentioned here understood — rightly — that they could probably not get software vendors to incorporate their interfaces into their products, and thus the interface vendors would have to do the work themselves. This is not all bad news, since the interface vendors will be able to sell the products at retail for from $50 to several hundred per application, while IBM will merely get at best one $149 (retail) sale per machine. (But of course IBM has different goals from the ISVs.)

The unique value of the TopView and Macintosh interfaces is that they will be widely used across a variety of different software, easing the user's learning and habit-forming difficulties. Products like Windows, Visi On and Digital Research's recently announced Gem (formerly Crystal), on the other hand, are likely to remain less widely used interfaces tailored specifically for graphics-intensive packages. (Gem does provide a startlingly Mac-like interface, for the moment the resolution of most PC hardware just doesn't do it justice. But it might make a new Atari machine look pretty good...) While they may have their own merits, these interface/environments lack the signal advantage of wide familiarity on the part of users; familiarity breeds value.

(Purposely omitted from this discussion is Quarterdeck's Desq, which has as its design goal the ability not to alter the interfaces of the user's favorite applications, although it does let builder-users create macros to run on top of or among them, and includes a variety of data transfer routines.)

Interfaces are only skin deep

Ultimately, any of these interfaces is only as good as its implementation — which lets ISV-oriented systems such as TopView and Macintosh off the hook. Aside from looking pretty, menus must be logically consistent, easy to understand, clear and simple. Moreover, what's the content of the menus? What happens when you're done with a task? If you're in the middle of one operation, can you get a listing of your options from which to select? Menus should be more than just program information and commands, but should also contain data that the user previously entered and from which he may now select (see especially Cornerstone, above). Overlays on top of a program generally can't offer this; programs written with the help of a menu-builder (and much more) such as TopView can.

Menus can be confusing by virtue of any number of flaws: inconsistency (Is exit always the last item?), confusing phrases, lack of a way to exit or return to a previous state, and so forth. The construction of the interfaces is not an issue that can be addressed generally; each must properly represent the capabilities of the product that lies beneath them. Elementary menus like those of the pfs: series are perfect for the software they front-end; more complex products require more complex menus. It's the builder's job to create the content in the interface and the procedures that underlie it. Moral: Don't blame the interface for the product.
Sweeping AI Into the Mainstream

"LISP is like a ball of mud, while APL is a diamond," said Guy Steele (quoting MACSYMA author Joel Moses of MIT) to an audience of AI luminaries two weeks ago at the International Forum on Artificial Intelligence in Palo Alto, CA. Steele/Moses' point was that APL is clear and pure, but relatively rigid; by contrast, LISP is open to change and can adapt to its environment. In place of APL, Steele might well have been referring to Prolog, but he was too polite to do so at a conference sponsored by Prolog house Quintus Computer Systems of Palo Alto, CA, to herald some product announcements (RELease 1.0, September 11).

Over the 26 years since its creation by John McCarthy (also present at the forum), LISP has competed with and ultimately absorbed most advances in programming languages, most notably object-oriented programming. By contrast, Prolog, newer and purer, remains brittle and resistant to change, but far more consistent. LISP comes in a bewildering variety of dialects, a bundle of scar tissue built from contact with the chaotic, careless real world, while Prolog is true to its conception as a logic programming language but less accommodating to the confusion of the real world, with its myriad kinds of data structures and relationships.

Just as LISP has co-opted much of its competition (in the AI world), so in a sense is the real world co-opting AI. On the hardware front, facilities once considered the province of AI -- bit-map, high-resolution graphics, mice, tagged architectures (soon on Sun) -- are now appearing on mainstream hardware.

And now even Prolog is addressing mainstream environments and concerns. Quintus's new Prolog, while it remains pure Prolog (written by the language's early implementor, Quintus co-founder David Warren) accommodates the real world substantially both in the machines it runs on -- the DEC VAX under UNIX 4.2 (Berkeley) and VMS, and the Sun Microsystems Sun-2 -- and in a variety of features. It sports an incremental compiler which allows selective recompilation (or a sort of interpretation) to speed up editing and debugging, an integrated editor, and a C interface allowing it to interact with programs written in C.

On the LISP front too there's a surge of mainstreaming activity, including Gold Hill Computer's implementation of LISP on an IBM PC. Another new LISP vendor is Lucid Inc., soon to change its name (new name not yet available) for trademark reasons. Lucid, also based in Palo Alto, is the creation of Richard Gabriel, co- instigator of the likely CommonLISP standard at Stanford, and ceo Tony Slocum, former president of IntelliCorp of Menlo Park and widely credited with turning that start-up into a company (Gene Kromer from TI is now turning it into a business). Lucid, funded with under $1 million from private sources, is still in start-up mode. By next summer, Slocum hopes to offer CommonLISP, plus a variety of development tools and language interfaces, for the Sun micro and the Apollo workstation, with VAX and IBM mainframe versions to follow.

As Quintus, Lucid and others bring AI languages to real-world hardware, real-world companies such as TI and Xerox are competing with Symbolics and LISP Machines Inc. in offering AI-optimized hardware. Sperry is teaming up with IntelliCorp to support and integrate IntelliCorp's LISP-machine-based KEE system into corporate and government environments. Sun, with its $9900 Sun-2/50 diskless workstation, now offers a hardware delivery vehicle for an increment of under $10,000, beating Xerox to the punch. Clearly, there will be no shortage of environments capable of designing and running all those AI applications people are now dreaming about.

RELease 1.0, November 13, 1984
Sometimes a top executive's departure signals problems, and sometimes it signals their resolution. With luck, the departures of both David Cole from Ashton-Tate and Dan Fylstra from VisiCor signify that their building and restructuring tasks, respectively, are over, and that both companies can now be born again.

In the case of Ashton-Tate, David Cole arrived to bring a small, "family" company with an entangled ownership to the point of being a multi-product, public company. In a sense, he may already have stayed too long, leading the company into peripheral areas like home software and publishing, both of which may stretch Ashton-Tate's meager resources. A-T has two fine products -- dBASE II/III and Framework -- but it must exploit them to the hilt in this increasingly brutal marketplace. So far, one might say that dBASE II has been marketed most successfully by third parties and that Framework has barely been marketed at all. Meanwhile, Cole's regime wasn't especially peaceful; the presence of the new management he brought in precipitated a number of departures. But his may be the final one. The company is now ready to move forward under the steadying hand of PC software veteran Ed Esber (ironically, ex-VisiCor). Its opportunities (read challenges) are clear.

VisiCor has suffered much greater turmoil and financial strain than Ashton-Tate, precipitated by a variety of intra- and intercompany conflicts and the lack of an adequate follow-on to the company's highly successful VisiCalc (designed by Software Arts). Over recent months, however, the de facto admission of defeat has liberated chairman and co-founder Dan Fylstra to get on with the task of restructuring the company to exploit the assets that remain. The final step has been to hand them over in a merger to Paladin (formerly OptiSoft), whose management will exploit them while Fylstra moves on to another chapter in his life, older and wiser.

The assets that attracted Paladin include close to $10 million in cash, raised in a recent venture funding and through the sale of the Visi On technology and the Communications Solutions subsidiary to two different units of Control Data Corp. (VisiCor/Paladin retains the retail marketing rights to Visi On.) VisiCor's assets also include about 30 remaining people (out of 40), many in product marketing, support, and finance and operations, and their industry connections, plus FlashCalc and the rest of the VisiSeries, which includes the excellent VisiSchedule and VisiWord but which badly needs maintenance and enhancements. On the debit side, there's several million of Visi-inventory on retail dealers' shelves that the company must help them handle if it is to regain their goodwill.

Paladin brings to the table a strong management team and an almost-completed low-end productivity product that will benefit from the VisiCor infrastructure.

The merged company, which will use the unsullied (but unknown) Paladin name and the VisiCor offices in San Jose, will sell FlashCalc and the rest of the VisiSeries while it completes development of its own low-end productivity product. It will be run by president and CEO Roy Folk, former VisiCor VP and Visi On product manager, and chairman Rich Melmon, former VisiCor director of marketing, and a recent consultant to Paladin among others.

We wish them good luck; this combination of resources certainly increases their chances of success.
APPLE 1985: ON BEYOND USER-FRIENDLY

In early 1984 Apple's girl in the red shorts flung a mallet of friendliness at the inhuman Leader (read IBM) in the famous SuperBowl ad. In early 1985, she's going to join a workgroup, shake the Leader's hand, and try to co-exist. (We'd like to see that ad!) 

Over the past year, with the Macintosh and the rejuvenated Apple //, Apple has developed a lot of momentum. The Apple // is likely to be sold out before Christmas; the Macintosh is a succès d'estime and generating cash flow if not profits. Mac has just gotten a big boost with the formal announcement of Lotus Jazz, shipping early next year, and its prominence in the recent special issue of Newsweek. For the business market, it will get further mileage from Apple's announcement of $50-per-node networking and an "insanely-great," multi-user-priced laser printer at the company's annual meeting January 23. 

But in the meantime, IBM has been rapidly putting mortar between the bricks of its mostly incompatible PCs, 5520s, System 36es and 370s. And third-party vendors such as Digital Research with its newly-announced GEM (formerly Crystal) interface and others with friendly software, have been trying to sell the concept of Mac on a PC. Apple can't rely on the Mac, even with half a megabyte of memory and Lotus Jazz, to keep it ahead of this powerful surge. 

So, you might notice a subtle shift in emphasis from the person to the workgroup, from the product to the system. User-friendly was last year; group-friendly is next year. What is group-friendly? Basically, it's communications as more than RS-232 ports and 3270 emulation. It's slick-looking documents (courtesy of the laser printer) for clearer communication of ideas as well as data, easy sharing of files, no-sweat electronic mail, and so forth. It's clear that the next 1-2-3 is going to be the ABC of mail, db sharing and Calendaring; Apple's trying to make sure it can happen first -- in truly easy-to-implement fashion -- in the Mac workgroup. Long run, Apple hopes for the Mac workgroup to be the friendly front-end to the corporate dp environment which will inevitably be made up of IBM hardware. 

That's Apple's view of things. Is it plausible? Apple faces a lot of obstacles. It's not considered a serious business company by most of the customers it hopes to address with its Mac-in-a-workgroup concept -- although it certainly has better chances of entering the corporation through the department entrance than through a central dp organization. Apple's challenge will be to make the Mac workgroup so friendly and so useful that it will be hard to resist. It will enhance the IBM environment, not overthrow it. The products are there: Jazz, the network, the printer, the file server, the PC operating as a comm server, etc. Although the network's rated bandwidth is about one-tenth that of the PC network (and one-fiftieth of Ethernet), it's generally the quality of the servers and the software that determine the actual quality of access on a network. 

Apple's ace in the hole is the quality of its products. Test-drive-a-Mac is a nifty marketing idea -- but it's the product behind it that will let it be so effective. It simply wouldn't work with the IBM PC AT. How about test-drive-a-Mac-workgroup? That may be what it takes. It would be worth the stakes.
WORDSTAR 2000: MICROPRO'S COMEBACK

The question is, Is this MicroPro's Friday, or its dBASE III? As those who follow Ashton-Tate know, Friday! was its unsuccessful attempt at filling the gap between its own dBASE II and Software Publishing's equally successful pfs:report/file (see page 11) with a limited version of dBASE II. dBASE III is its more recent, seemingly successful effort; it is even more powerful than dBASE II, but also easier to use, courtesy of a lot of user-friendly features (prompts and helps, mostly) and a total rewrite.

WordStar 2000 is no Friday!, and ranks with dBASE III in quality, but MicroPro needs it even more than Ashton-Tate needed dBASE II. Aside from recent inroads by Microrim's Rbase 4000, dBASE II hadn't really lost substantial ground, and still had the dominant share of the market. Although everyone was looking for a dbms replacement, no one had found one (despite strong efforts by Power-base et al.). But new word-processors abound, and a number -- pfs:write, MultiMate, Samna, Microsoft Word, IBM's Displaywrite -- have gained substantial visibility. As a percentage of its market, WordStar's share is substantially below dBASE II's, despite WordStar's 1.25 million (legal) users. Many of them may consider WordStar 2000's nicer features unnecessary frills, and be loath to learn a new set of mnemonics.

But like dBASE III, WordStar 2000 ($495, $595 for WordStar 2000 Plus) is a total rewrite in C of the original that retains most of its features, albeit in a friendlier form, and adds many more. Specifically, WordStar 2000 retains the use of control keys and the optional top-of-screen menu that characterize the original WordStar, but now the mnemonics make sense; i.e., block commands now begin with control B, rather than control K. Searching for a character string generates a prompt asking if it is to be replaced or merely found. A spell-checker works during editing and replaces SpellStar, which has to be loaded separately to work with WordStar. A key glossary inserts user-defined strings of text at the press of two keys. Text reformats automatically. And so forth.

It shines

Our preliminary tryout suggests that the product compares strongly with others on the market, and should make life difficult for the Volkswriters, Selects, Word-perfects and other second-tier word-processors out there. It should also make life a lot easier for MicroPro, even though Samna and MultiMate will survive handily addressing a corporate market which puts great stock in those products' similarity to the Wang wp they know and love.

Of course, MicroPro is in dire need of a smash hit currently. The graph shown at a recent analysts' meeting looked more like a mountain, sloping down at the right, than the proverbial hockey stick usually displayed at such meetings. WordStar 2000, one of a few such products, actually has the potential to turn MicroPro around. We don't expect any hockey sticks, but that mountain slope may start turning upward in a quarter or two.

Contributing to the prospects of WordStar 2000 is MicroPro's new corporate strategy of concentrating on what it does best: word-processing software. Eschewing a recent tradition of expanding beyond wp into CalcStar, InfoStar, ChartStar, Starburst, etc., new president and ceo Glen Haney has gotten the company to focus its sights a little. Wp is not a limiting field, he says with justice, when you consider all its aspects: document conception (outlining, note
management); text input; editing (spelling, formatting, indexing of a single
document, etc.); and distribution (printer management, mail-merge, indexing of
multiple documents, storage and retrieval, e-mail, communications, file transfer,
and the like). There is plenty of room for the company to broaden its line and
expand its market without straying from the category it knows best -- and where it
is known best.

As products (and concepts) like TopView gain ascendancy, this approach looks more
and more sensible -- and MicroPro's only hope of reestablishing its tarnished
stardom. Even the Plus version requires only 256K to run, and could easily
coexist with, say, a 1-2-3 under TopView. Indeed, just as 1-2-3 integrated
appropriate functions (graphics, mostly) with a super spreadsheet, so has MicroPro
integrated appropriate functions -- mailmerge, spelling correction -- with a new,
improved wp. MicroPro has only to look at Lotus's subsequent, less happy
experiences with Symphony to learn the advantages of sticking to a tight product
concept. Unlike Lotus and notwithstanding its $22 million in current assets,
MicroPro doesn't have the strength to endure anything less than a total success.
"Our parting was very mutual," said Lorraine Mecca of Micro D's and Applied Computer Techniques' decision to end their distribution agreement for ACT's Apricot after a six-month tryout. Mecca's Micro D found the product wouldn't sell; ACT found that Micro D couldn't sell it. Apple's parting from its reps as of October 1 might have been a little less mutual, but it left them similarly in need of a partner.

ACT, based in Birmingham, England (Release 1.0, July 31), is now creating a new Bay Area company to succeed its U.S. subsidiary and work with the former Apple reps -- 15 organizations with about 150 people. The new company, Apricot Inc., will have funding of about $20 million, most of it from ACT shareholders, with a little under 20 percent contributed by ACT itself. Employees will have options on about 5 percent of the stock.

The company's formal goal is to sell 15,000 units over the next 12 months, says Apricot vp marketing (and former ACT UK group marketing director) John Leftwich, but he'd like to do better. Sometimes it seems that if you can't sell 100,000 units you're better off not even trying, but ACT has enormous strength overseas and a solid product line (Release 1.0, June) that it may be able to leverage over here. Meanwhile, the former Apple reps are ravenous for product, and know lots of places to sell it into -- especially if Apple can't fill holiday demand for its Apple // line. It's a promising enough venture to be worth trying.

**TANDY: LEARNING NEW TRICKS**

There's a school of thought that says a good manager can manage anything. Likewise, Tandy once believed, a good retailer can sell anything. But now the company is acknowledging that computer retailing is a different sort of business, and it's putting a number of programs into place to recover some lost ground.

First of all, the company has moved about as close as you can to total IBM PC compatibility while maintaining superior price/performance with its new 1000 and 1200 models, supplemented by a handy desktop manager, DeskMate. Second, it has begun to promote the efforts of its third-party software vendors, starting with a series of mini-trade shows around the country. And finally, the company has begun to put enormous, appropriate stress on training both its own personnel and its customers. While the market won't ever be as welcoming as it was when Tandy first entered it in 1977, it probably will respond warmly to Tandy's recent efforts to serve it better.

**THE PATTERN IS FAMILIAR...**

Anyone requiring crack calculating capabilities probably has an 8087 chip (from Intel) in his computer. Within a few years, anyone requiring pattern recognition will probably have the PF474. What can pattern recognition do? Well, it ranges from things as simple as data base searches (especially when your spelling isn't too terrific), to capabilities such as voice recognition. The company that
designed (there's that spelling problem again) the PF474, Proximity Technology of Fort Lauderdale, FL, has done a lot of well-known pattern-recognition work already, and supplies the private-label spell-checkers sold by MultiMate, Sorcim/IUS, Office Solutions, Professional Software and Quadratron, among others.

PF474 is especially helpful in doing "fuzzy searches," i.e. finding what the user wants even if he didn't specify it quite right. Most data bases still work only on exact matches, which is extremely frustrating to many of us, who may be literate but still can't type.

But there are other uses too, which Proximity is at pains to sell now until (it hopes) the chip will be more generally adopted. The most impressive is its CleanMail program, which relies on the PF474 for its speed and "fuzziness." How often do you get several pieces of the same mail, each addressed to you at a slightly different address? (Our favorites include Adventure, EDvventure Holidays, Esther Rosen, RElease I/O, and various incorrect variations on the peculiar capitalization we use. See page 1 for the correct form.) Using CleanMail, it's a simple matter to download a mainframe data mailing list and scan it for near matches. The operator then decides which of the matches to discard. The answer may be obvious, it may require a check of some source material, or it may not even be a duplication, as in two Radio Shack stores at different locations in the same town.

Proximity is currently marketing the device as an OEM product, although it hopes eventually to generate enough demand to need a second source.

PROJECT MANAGEMENT

Project management has suddenly become the fifth/sixth/take-your-pick category of productivity software. In a sense, it's the first popular category of management, rather than analysis, software. (See page 5.) Managers could receive templates listing the cost and availability of resources, and standard times taken to complete common tasks within an organization.

The two leaders -- pioneer Harvard Project Manager and Microsoft Project -- are about to be joined by a third, Time Line, a newcomer from Breakthrough Software of Novato, CA, a publishing company founded by former IUS marketing manager Bill Lohse. Time Line, written by Andrew Layman, is exceptionally flexible, with simple facilities for the export of cost and resource data into a spreadsheet or database for analysis. It can allocate resources as well as track them, so that, for example, if the number of ovens is limited it will automatically schedule the cooking to take longer, rather than merely report a conflict. Finally, Time Line's interface smacks of Lotus 1-2-3, an added attraction for many people.

Meanwhile, Harvard, MA-based Harvard's new release, Harvard Total Project Manager, fills many of the holes left in the original HPM, notably the lack of resource tracking and the ability to manage multiple projects. And the product retains its signal advantage, a nifty facility for creating PERT charts (the roadmap) that matches anything we've seen on the PC. And although LisaProject looks better, it makes the user do the layout himself, while HTPM does it for him. He can also do neat what-ifs by changing precedences on the fly and having the system redraw the chart automatically.
Applitech Software of Cambridge, MA, has just released its nifty Project Planner ($150) for the Apple // line, which lets you create a list of your tasks, assign precedences to them all at once rather than one at a time, and have the system draw a PERT chart. This lets you see all the components rather than look at it in little chunks as you build it, while the system builds the structure for you. In fact, says evp and co-founder Linda Carma, she has a customer who uses PP to design PERT charts which he then draws into his LisaProject.

And then there's Sorcim/IUS's SuperProject in the wings.

INTERNATIONAL SOLUTIONS: OTHER PEOPLE'S NATURAL LANGUAGE

If you think there's a shortage of Macintosh software here, you should investigate the situation abroad. This very dearth of product makes these markets especially appealing to U.S. vendors, if they only had the resources and overseas experience to exploit them. Attempting to ameliorate the situation is International Solutions Inc. of Saratoga, CA and, soon, London. Its founder and president, Mark Kvamme, 23, went to work for Apple for the first time in 1980 as a summer intern. The next summer he worked for IBM Norway, and then moved to Apple France in October 1981, a good two years before his father Floyd joined the parent company as executive vp (he's now a partner at Kleiner Perkins). Adding seasoning to the team is Kvamme's Chilean-born partner Gustavo Cardemil, 57, who has had 20 years of international experience, including 11 years at National Semiconductor as director of international marketing.

International Solutions is small enough -- six people -- to be extremely flexible about what it will do, but basically it takes over the job of reinterfacing US products for and distributing and marketing them in foreign countries. So far the tiny company has six clients, plus four divisions of Apple, for whom it has designed character and keyboard firmware. The other clients are considerably smaller, including ProVUE Development (page 14) and Haba Systems of Van Nuys, CA (RELease 1.0, July 31). IS has adapted ProVUE's OverVUE for the U.K., France, Germany, and Italy, and has already distributed 1250 copies overseas. For Haba, it has created French, German, Italian and U.K. versions of the Habadex filer/dialer, with special attention to local telephone customs.

Although it's not formal company policy, so far all the work IS has done has been for Apple products.

RELease 1.0, November 13, 1984
SPEAKERS AND PANELISTS

Eighth Annual

PERSONAL COMPUTER FORUM

Sponsored by EDventure Holdings Inc.

*Martin A. Alpert
James L. Barksdale
Robert F. Berland
Daniel S. Bricklin
Rod Canion
John D. Couch
Arthur W. Einstein, Jr.
Edward M. Esber, Jr.
Barry James Folsom
William H. Gates
Fred M. Gibbons
Lee M. Hecht
Jeffrey Heimbuck
Steven P. Jobs
Eric L. Jones
Mitchell D. Kapor
Ralph E. Kromer
William E. Ladin
David E. Liddle
Robert A. Markell
Regis McKenna
Richard Melmon
Seymour Merrin
Robert M. Metcalfe
Marvin Minsky
David A. Norman
Richard K. Oppenheid
Michael S. Fliner
James N. Porter
Vern L. Raburn
Jef Raskin
Benjamin M. Rosen
*Morton Rosenthal
John R. Rowley
Benjamin L. Scott
John Sculley
Michael Shabazian
John F. Shoch
Alan F. Shugart
Sirjans Lal Tandon
David S. Wagman
John P. White
Cyril Yansouni

Tecmar
Federal Express
IBM
Software Arts
Compaq
Lightyear
Lord, Geller, Federico, Einstein
Oracle
Ashton-Tate
Digital Equipment
Microsoft
Software Publishing
Teknowledge
Koala Technologies
Apple Computer
Texas Instruments
Lotus Development
IntelliCorp
ComputerCraft
Metaphor Computer Systems
IBM
Regis McKenna
Paladin Software (VisiCorp)
Computerworks
3Com
MIT
Businessland
Oppenheim, Appel, Dixon
Sytelk
Disk/Trend
Symantec
Information Appliance
Sevin Rosen Management
Corporate Software
Digital Research
AT&T Information Systems
Apple Computer
ComputerLand
Xerox
Seagate
Tandon
Softsel Computer Products
Interactive Systems
Hewlett-Packard

President
Executive Vice President & COO
Director, Programming Product Line
Chairman
President & CEO
President
President & Creative Director
President
President & COO
Rainbow Group Manager
Chairman
President
Chairman & CEO
President & CEO
Chairman
President
Chairman & Data Systems Group
Chairman & CEO
President
Chairman
President & COO
VP, Software Development & Publishing
President
Chairman
Chairman & CEO
Chairman
Donner Professor of Science
President
Partner
President
President
President
Chairman & CEO
Chairman
Chairman
Chairman
Chairman
VP Marketing
President & COO
General Manager, Retail Distribution
President & CEO
President, U.S.
President, Office Systems Division
Chairman & CEO
President
Chairman
Chairman & CEO
VP & GM, Personal Computer Group

*Recent additions are indicated by an asterisk.
SUBSCRIPTION FORM

Please enter my subscription to **RELease 1.0** at the rate of $395 per year in the U.S. and Canada. Overseas subscriptions are $475, airmail postage included. Multiple-copy rates on request. Subscriptions are available on an advance payment basis only.

<table>
<thead>
<tr>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
</tr>
<tr>
<td>Company</td>
</tr>
<tr>
<td>Address</td>
</tr>
<tr>
<td>City</td>
</tr>
<tr>
<td>Telephone</td>
</tr>
</tbody>
</table>

Please send payment and this form to: EDventure Holdings Inc.
375 Park Avenue
New York, NY 10152