FORUM DRESS REHEARSAL: BEHIND THE SCENES

This is the director's script for the 1991 Platforms for Computing (PC) Forum, "Beyond the desktop: Networks, notepads and legacies." In it we outline the flow and set the scenarios for the Forum. It should help you get more out of the event (or catch a sampling of what's new this year if you're not attending). Most of the companies and products featured at the Forum have been covered in Release 1.0 over the past year; we analyze the new ones (including several first showings indicated by asterisks) at greater length.

We have managed (been forced) to get this program/issue together early this year because it will be posted as part of a Forum-wide demo/working system of Lotus Notes, implemented by Mark Tebbe and his company, Lante. The system will provide a means for users to check the Forum schedule, suggest additional topics for panels or individual speakers to address, send messages, make comments or carry on discussions, arrange open meetings or product showings, collect groups for dinner Tuesday night and otherwise explore the uses of such a system. Furthermore, we hope you find uses for it beyond those noted; that's the point. Anonymous -- but tasteful -- contributions are encouraged; management reserves the right...

That's one aspect of the atmosphere we are trying to encourage at the Forum -- an exploration of once-exotic technology in everyday use. Notes is only one of three featured demos/facilities available to you throughout the three days. The second is American Information Exchange, where you can "buy" or bid for information (free during the Forum). AMIX, still in beta, will offer online information on topics featured at the Forum, from suppliers including PC Letter, Soft·letter, The Office Computing Group, Clarke Burton, Wohl Associates, and Wall Street analysts Michele Preston, Rick Sherlund and Charlotte Walker, as well as ourselves.

The third is Verity, with a filtered, categorized version of the Dow Jones' DowVision, offering customized news briefings to those who file interest profiles. (Other companies will show comparable and complementary products during the afternoon company presentations; see page 2.)

THANK YOU, DAPHNE!

INSIDE

FORUM DRESS REHEARSAL 1
Stage directions.

ACT I: NETWORKS 8
Information sharing.
Demos: Information groupware.
Text tools.
Dinner: Profit w/o guilt.

ACT II: LEGACIES 12
Objects and transactions.
Demos: Clients & servers.
Workflow groupware.
Cross-platform tools.
Component software.

ACT III: NOTEPADS 24
Penware: State of the art.
Demos: PenPal software.
PenPal platforms.

ENTR'ACTE 27
Fish story.
Electronic frontier.

FORUM DETAILS 28
Demo-picker.
Company index & phone list.

EDVENTURE HOLDINGS INC. 375 PARK AVENUE, NEW YORK, NY 10152 (212) 758-3434
Interoperability

The second flavor that reflects a greater trend is the increasing proportion of joint company/product presentations: Hewlett-Packard and Sun, Keyfile with Bloc Publishing, Eden Group with Nestor and Pen Windows, Slate and Pen-Soft with PenPoint. NewWave shows up in AT&T's Rhapsody; Desktop Data, Beyond and cc:Mail work with Lotus Notes; PowerBuilder and Cooperative Solutions' Ellipse use Sybase's SQL Server; almost everyone uses Windows.

Someday we hope that everything will be so interoperable that we'll be able to have a single demonstration incorporating all the vendors. Of course, true openness requires business as well as technical cooperation...

DAY BY DAY: STAGE DIRECTIONS

The Forum runs three days, with a different focus each day. On Monday we discover that information is meaningless unless it's shared; on Tuesday, that it's meaningless unless action is taken. On Wednesday, we explore how to extend that world of action/information to a wider range of users in a wider range of places. In short, Monday is networks (for information-sharing), Tuesday is reuse of legacies and development of future legacies (by way of object-oriented programming and client-server tools), and Wednesday is notepads (as a base for new user environments). Obviously, there's a lot of carry-over from day to day, and many speakers who could speak on any of the days. But we've decided to focus each day around a specific idea and product and suggested scenarios for the demos to avoid vision and vapor -- generalities and vapid endorsements of openness and standards.

Instead, we want to provoke new thinking about real products. We have selected the vendors and products to serve as subjects for discussion, not necessarily as ideals to emulate in all respects.

Companies, people and products featured at the Forum are in boldface, and are listed alphabetically by company on pages 30 to 31 with page numbers, phone numbers and references to previous Release 1.0 coverage. In addition, a table on pages 28 to 29 lists the topics associated with each company presentation, so that you may quickly select those of greatest interest to you. (The schedule is the same both Monday and Tuesday.) New product announcements are marked with an asterisk and are described in greater detail.

ACT I -- NETWORKS: INFORMATION-SHARING AND GROUPWARE

A fundamental, obvious point about networks is that they involve groups -- individuals sitting at separate workstations connected by wires. Along those wires flows information -- structured and susceptible to manipulation by a computer, and unstructured, waiting to be assessed by a person. The structuring of that information can be managed centrally, or by the combined, mostly ad-hoc efforts of individuals or their e-mail agents, as shown across. Plain old databases are the best example of central structure for information we consider to be structured already; text filters and ancillary tools are a semi-automated way of deriving the structure of information.
without visible structure. Once the structure has been determined and represented electronically, the information is then subject to automatic manipulation, whether by database transactions, e-mail rules or other applications. (Call such structured information objects.)

### Structured...

<table>
<thead>
<tr>
<th>locally</th>
<th>at the server</th>
</tr>
</thead>
<tbody>
<tr>
<td>e-mail</td>
<td>Notes</td>
</tr>
<tr>
<td>wp files</td>
<td>Reach</td>
</tr>
<tr>
<td>outliners</td>
<td>gIBIS</td>
</tr>
<tr>
<td>enhanced</td>
<td>databases</td>
</tr>
<tr>
<td>e-mail</td>
<td>filtered</td>
</tr>
<tr>
<td>Agenda</td>
<td>news</td>
</tr>
</tbody>
</table>

Most information-sharing groupware tools fit somewhere in this chart, which shows how the content is managed -- or structured. (We outlined a complementary framework for workflow groupware in Release 1.0, 11-90.) **AMIX** is a special case, since the actions of local individuals are represented centrally. Control is distributed, however, since over the long run AMIX market-makers reshape the structure in response to local activity. (In the real world, barring interference, markets are distributed -- which is why they make so many people nervous.)

Plain e-mail lets users structure their mail by hand; enhanced mail tools let users automatically create structure (currently, hierarchies of folders and associated actions; eventually, who knows) with rules, filters and other scripted routines. Much the same filters for automated derivation of structure can work both locally and centrally (or shared on a server); the difference here is who programs and controls them.

**Where does your group begin?**

As noted, **Lotus Notes** is server-based. The server (or replicated servers) manages the information, and presents it in various ways to the users. Notes is the first significant information-sharing groupware tool, and also the first programmable one (cf. the macro facility in 1-2-3 back in 1982). Application developers and users can use Notes to structure their data in hierarchies, tables, cross-references, and to sort it and select it according to various criteria. Depending on politics, not technology, the content and structure can be controlled centrally by a corporate application group, or it can be left in the hands of individual teams of users.

One of Notes' technical weaknesses -- or compromises, anyway -- is its use of a nonstandard, proprietary database, which is ideal for the basic Notes functionality but not so good for integration with the rest of the world. Ray Ozzie of Iris will discuss some of the technical considerations behind the product he designed and developed. Moreover, the interoperability issues are being addressed, with the acquisition of cc:Mail (for less structure-intensive information-sharing, with interfaces to Notes likely) and better data-transfer and interoperability features such as the use of DDE.

**Notes** was first mentioned to the public at the 1988 Forum, in a speech with slides by Jim Manzi. Three years later, he's back at the podium, with a live demo of the second release of the product. There are now 35,000 Notes users at 75 companies such as Price Waterhouse, Manufacturers Hanover and PG&E. Still, it has been a challenging product to market and to sell, with a high upfront cost and heavy support requirements. Lotus has now signed up VARs to sell smaller numbers of copies than the 200 units minimum Lotus requires for direct sales, lowering the barriers to adoption of the system.
Information far from your fingertips

The purpose of office automation, ultimately, is communication of information. You collect the information, assemble it and analyze it to make it more meaningful, and then you pass it along to someone—a colleague, a customer or a supplier. Meanwhile, when you write a letter, most of the information probably didn’t come straight out of your head—or needn’t have. It contains names and addresses and data from corporate files, company policies, answers to questions you asked of colleagues or competitors, snippets of data gleaned from today’s newspaper and last month’s strategic planning meeting.

Until recently, most of this information wasn’t available electronically. Even the transaction data was locked away on a mainframe, and the spreadsheets were spread around on individual pcs. Now we’re beginning to share access to such data with networks, file servers and a variety of database access tools (such as Metaphor/IBM’s Data Interpretation System, Channel Computing’s Forest and Trees, shown at the Forum two years ago, and Pilot’s Executive Information System), and this year’s InfoAlliance from Software Publishing.

But the majority of the information can’t be culled with SQL; it’s personal databases kept in card files, papers kept in piles on a shelf, old magazines and newsletters, memos, letters, manuals. Nonetheless, more and more of this information too is finding its way into electronic form—not in SQL databases, but in non-standard mail-merge files, wp documents, and e-mail messages filed only by date (and retrievable by sender or full-text search). Users start downloading news articles and figuring out ad-hoc ways to classify and organize them. The more this happens, the more people notice the potential of electronic information—and its limitations. In short, storing the information is easy; making use of it is hard.

People whose business is information, such as technology and consulting firms, tend to notice first. They've had too much information in hard copy; they know computers aren't just a way to produce or store information, but also a way to classify, filter and distribute it.

Once you have the structure, the information can be handled automatically because not just the image and text but also the meaning are represented electronically. We already have that luxury with database data, to the extent that the "meaning" that the database "understands" is sufficient. When it is not, we look to object-oriented databases, applications that hold the meaning (which is harder to share when it’s confined to a single application), and the leave-it-up-to-the-user approach.

Since textual information is so varied, there's no SQL yet to serve as a language to cross the client/server barrier, selecting the right view of the information for the user at the workstation (although Apple, Thinking Machines and others are exploring the concept). The issues are far more complex. Text has a variety of meanings, data structures, presentation techniques and metaphors.
Waiting in the wings

American Information Exchange, although also server-based, leaves selection of the content mostly in the hands of individual providers and users, although its "market-makers" provide a structure (mostly hierarchical) for the contents. The demo of AMIX will include three electronic markets, covering the three themes of the Forum: Lotus Notes and related topics (groupware, e-mail); object-oriented programming and related topics (almost everything!); and GO Corporation and pen-based computing. Each market will consist of electronic material about the topics, drawn primarily from the publications of a variety of conference attendees who sell such information for a living, along with bios of the information suppliers and users' comments about their works. You are invited to "buy" these articles (free for viewing at the Forum), add your own comments, commission new research, or order subscriptions from the suppliers. The terms are up to the sellers and the buyers; AMIX is only the market facilitator.

The Reach Networks system, managed by Andy Zimmerman of Coopers & Lybrand (presented briefly on stage and demoed in the afternoon sessions), is a multi-structured, networked and remotely linked system that allows users to communicate in a variety of ways. There are formal "shows," akin to tv broadcasts, developed by designated teams of people with expertise in a particular area. These shows take the place of company newsletters on specific topics, but they allow for much better targeting of information to people with specific interests, and also for feedback and comments by "viewers." (Shows that don't get watched get changed or axed.) In addition, users can send each other e-mail, post comments, broadcast requests for information and order back-up materials.

The system combines firmwide, topdown communications in specific practice areas with horizontal communications among users. The tv analogy is apt; users can choose exactly what to see, but they also know what to expect from any particular "channel." The system can be as structured or unstructured as each user prefers. Compared to Notes, it's better for large groups who want to keep track of external information or send it out to clients; Notes is better for individuals managing task-specific, granular information related to a single project or workgroup. Like AMIX and Notes, the Reach demo will include information about Forum companies, presented as shows, as well as examples of the system's use at Coopers & Lybrand.

E-mail -- leave it to the user

The systems above require someone "in control" or "with authority" to think ahead. E-mail is obviously more appealing, because it requires less work upfront, but it produces less structure after the fact. E-mail starts with the individual, leaving the organization of the data as a local task. (See Release 1.0, 90-11.) There may be some simple store-and-retrieve functions on the server, but by and large the intelligence depends on the user.

However, programmability is now coming to individual users in mail tools such as Agility's Wijit and Beyond Mail from Beyond, for which Soft Switch's Mike Zisman serves as a director. They let individual users think ahead (or build systems incrementally). Users can define specific types of mail and conditions (such as sender, time, subject, words in the text, and specific values in user-defined fields), and program user agents to handle their mail
for them automatically. Generally, a user can use such a system without the rest of his group doing anything or even knowing. (They may wonder, however, at the promptness of his responses.) Likewise, the user can communicate with news services, bulletin boards, company databases and other "public" facilities with no action on the information provider's part.

Of course, these tools can be enhanced or combined with a variety of filtering, classifying and organizing tools, such as Verity's Topic, Individual's SMART tool used to produce First!, Desktop Data's NewsEDGE news-sorting system, and ALI's text engine. One way or another, too, we expect to see cc:Mail working as a front-end to Notes.

E-mail as a transport agent

Zisman plans to focus his talk on mail -- an appropriate topic for someone whose company Soft•Switch is the premier vendor of mail-system gateways. This year is seeing the blossoming of e-mail, including Lotus's acquisition of cc:Mail and Novell's assumption of responsibility for MHS (Message-Handling Service) from its original developer, Action Technologies. "It used to be file and print," says Zisman. "Now it's file, print and mail."

Moreover, mail needn't comprise just text messages. It can transport structured messages such as appointment requests, approval forms and purchase orders, or objects and agents. Instead of text search, you can use form parsers to interpret structured messages. Such programmable e-mail front-ends turn out to be both personal organizers and ultimately programming tools. If you automate yourself, and your agent is talking to another agent, you end up with inter-application communications.

In fact, says Zisman, long run most e-mail will probably be used among applications rather than among people. Automation that helps people do routine jobs will start to do the routine jobs, instead. If a customer wants information, why not let him talk to the database directly, instead of to a keypunch operator at the end of a telephone? (This is either a threat to jobs or a liberation of people from onerous tasks, depending on the availability of training for other, better jobs -- and the availability of such jobs after the training.)

Power in platforms: what for?

When we went to visit Andy Grove at Intel to invite him to speak at the Forum, we expected to hear about his secret plans for the 986. Instead, Grove pressed us to look at his computer, running Windows 3.0 and displaying an ugly e-mail system in a window, along with icons for a variety of other applications and data. Grove was excited, enthusiastic. Now he knows from experience why we need the next few orders of magnitude of power increases: In order to have voice and video instead of text in our e-mail, in order to process image, in order to compress and decompress megabytes of data in real-time. As our tools get better, we'll want to share more and more information across networks, leading to the need for more processing power, more storage, higher bandwidth -- in short, more hardware of every kind. Someday we'll have multimedia creation tools, video e-mail, and possibly animated representations of ourselves. (Do we really want them?)

And it leads to another problem (unless we're all using 986es with the same software): How do we make all this stuff look the same on disparate plat-
forms? John Warnock of Adobe Systems will tell us his answer -- which will work wonderfully if enough people adopt it, just like PostScript.

The cost of change -- value-subtracted

But all this magic means little if users can't use it. Grove has an assistant who spends at least half his time keeping Grove's system working (shades of Steve Jobs' promise of "a mother in every box"). Mort Rosenthal's company, Corporate Software, offers a Windows-upgrade service that costs hundreds of dollars per user: "You do the first few hundred users in a large corporation," he says, "and then you let internal technology transfer take over." Mark Tebbe's Lante Corporation makes a substantial sum by installing Notes systems. How can these products be made easier to use? And how can they be sold? Is such groupware really just mainframe applications in disguise?

Rosenthal and Tebbe have fundamentally the same role at the Forum -- to talk about support and implementation, the role of services and all the system integration tasks that still aren't automated. In short, they're here to keep the vendors honest.

Lunch: Wiring the world from the bottom up

Ken Oshman, a co-founder of Rolm and more recently of Echelon, will talk about the challenge of trying to establish a standard piece by piece with LonWorks. LonWorks is a low-cost networking system with modules that can fit into common household appliances, office equipment and factory machines. The principle is that you don't need a lot of intelligence in a single place or high-bandwidth communications in order to build an intelligent self-organizing system with high-level distributed intelligence. (Think of the effective group coordination you can get using low-level e-mail messages.) Aside from the technology and philosophy behind Echelon, he will discuss the tribulations of dealing with what Philippe Kahn calls standards comedies.

INTERMISSION: INFORMATION-SHARING GROUPWARE

There are a variety of different ways information may be shared. Mail systems treat information as discrete objects. They may be identified by a number of attributes, but the system doesn't care much about their content. Other tools help derive a sort of structure, depending mostly on the providers of the information to specify what kind of information it is and where it fits. Notes is highly structured, offering users the ability to manipulate information in views, reorganize it, etc. Reach, very much an electronic publishing medium, offers a less flexible, more strictly hierarchical structure. Users can select (ad hoc or through profiles) articles, or shows, on various topics from a structure built by builder/users.

1 Consider also Answer Computer's Apriori, shown last year, which defines a structure based on following paths trodden by helpdesk personnel finding answers to customer questions. Eventually, the most frequently needed answers move to the top of the hierarchy, becoming easy to find -- and indirectly alerting management of bugs or simply design problems that make products confusing for users.
gIBIS from MCC fosters users' discovery and representation of the implicit content in their information; gIBIS stands for graphical Issue-Based Information System. It is a research project at MCC, the Austin research consortium, that is about to be the foundation of a start-up run by its developers Jeff Conklin and Michael Begeman. gIBIS is a hypertext tool: It lets users display arguments, comments, opinions and other texts concerning a topic, and to place them on screen as nodes in a web of links signifying their relationships to one another. (The original version used indented text, somewhat like the annotations, comments and countercomments in products such as ForComment and Notes. But that failed to capture the multi-dimensionality of the issues and links.)

gIBIS lets you filter the web of nodes by the kinds of links between the nodes: displaying only supporting arguments, for instance, or counterexamples, or assertions by a certain person. The explicitness of the categories, notes Conklin wryly, tends to expose the hand-wavers and the shouters. The value is both that arguments (sorry, discussions) are clarified as they are made, and that decisions can be researched and reconstructed after the fact. You don't have to replay the whole discussion, nor can you forget factors that were important -- although you can always add new ones or account for changed circumstances the second time around. "Now why did we send Alice to sell coal in Newcastle?" or "What were the factors that led us to cancel Juan's project?" or even "Who claimed that we could improve productivity by 200 percent, and what were his assumptions?" Project leader Jeff Conklin is using it in a trial with a utility, where such records are all-important in filings with regulatory authorities.

TEXT TOOLS: LET US FIND IT AND DEFINE IT FOR YOU

Automated tools for information classification can work well with some of the systems above, particularly the mail filters, although the integration may not be totally smooth. The services described (and demonstrated) here offer additional value-added. They not only collect and filter news from diverse sources, but also negotiate favorably priced contracts with information suppliers (including wholesale pricing from news vendors such as UPI and Dow Jones), which cover multiple-user sites and reuse within customer organizations on various terms. Essentially, they are information retailers. While electronics may solve the physical distribution problems (as the many unsuccessful electronic-software-distribution ventures have noticed), for now it still takes a middleman to negotiate the transfer and manage the process.

Desktop Data and Individual Inc. both offer filtered news services: Desktop Data's delivered on-line and Individual's delivered by fax or e-mail. Both Desktop Data and Individual use Boolean text searches aided by the news vendors' own classification systems; Individual also uses an expert system to fine-tune each user's profile (an iterative process). Verity, which sells its filtering tool, Topic, as a stand-alone product, has also just entered into a resale agreement for Dow Jones' DowVision. Verity has a topic hierarchy that consists of collections of terms (France equals Paris plus de Gaulle plus Cannes and so forth).

Both Desktop Data and Individual, companies that originally sold the news as a stand-alone product, and Verity, which sold its tool stand-alone, have noticed increasing interest among customers in the ability to combine their
products with other corporate data and applications. Users no longer just read the selected texts; they want to file, cross-index, forward, excerpt and otherwise manipulate them. Accordingly, Individual is seeing increasing use of its e-mail (as opposed to fax) service, by customers such as Lotus (600 to 700 people) and Oracle (120 people). Desktop Data has built links to Lotus Notes as well as executive information systems from Pilot Software and IBM. And Verity has provided an SQL bridge to databases such as Oracle and Sybase and import capabilities for wp documents.

Desktop Data, with more than 500 users at more than 65 institutions, has repositioned its NewsEDGE from a standalone service that by the way could use your pc. Over the last year Desktop Data has gone through four releases of its software, and added 12 new news wires -- one factor that makes subscription pricing and direct customer contact necessary. It has added an OS/2 server version, NewsEDGE/LAN, for multiple users on a LAN using Windows 3.0 or DOS. Having DDE within Windows lets the users (or their internal gurus, anyway) do much of their own integration work -- and makes the product far more useful. Incremental users cost $300, plus appropriate wire service charges.

Individual will demo its existing service, First!, and also show INFOagent, a filtered service that filters the Information Access Company's (Ziff-Davis) database of 150 computer trade publications (abstracts and some full-text). This is the same material that eventually shows up in Computer Library (and now Computer Select) on CD-ROM, but more quickly and pre-filtered according to a customer's profile.

Verity will have an online demo throughout the Forum (rather than a company presentation), displaying news stories as they come in over the wire. Attendees will have a chance to see how the topics are generated (basically, as a tree structure within a PM interface), to pick their own choices from a topic hierarchy, and to watch breaking news that fits their specified categories as it's received over DowVision.

When you see these products, to say nothing of reading through e-mail all day, you will see why there might be a need for Adobe's cross-platform presentation system. If you weren't intensely interested in the content you select from these services (and presumably in your mail), you'd never read all that plain text. So much for advertising in online services for now!

Text engine

Can you understand natural language? Try these: A memorable speech, Fred! Let's have lunch. I'll call you. Sure, I like it. We have some weather ahead. The boss would like to see you. She has such a nice personality. We'll ship next month. 90 percent complete. A typical man. An offer of withdrawal. Law and order. Offensive by community standards. It worked out for the best anyway. [Or most of the sentences in this newsletter!]

As we've discussed before, statistical, heuristic tools are surprisingly effective for determining what a particular piece of text is about, but not much use for figuring out what it says. Did the girl get the boy or lose him? Is the speaker for the war or against it? For that you generally need a person, as gIBIS, Notes and most other tools implicitly acknowledge. (Sometimes the presence of certain words, such as "imperialist" or "God,"
could help a statistical analyzer to divine the speaker's origin and his likely point of view, but the speaker may be using the opposition's words ironically to buttress his own more reasonable stance. Look for terms such as "so-called" and "ostensibly.")

But natural-language understanding, while it's unachievable even by humans, can be approached by natural-language parsers and other tools. They are extremely resource-intensive, because they need to execute not only rules of grammar, but handle thousands of context-sensitive pattern-matches, conditions, and word meanings. A baby comes to understand the meanings of thousands of words only over a period of years.

Artificial Linguistics Inc. has produced a system that comes closer to anything commercial we've seen to doing this job. The product of 16 years of work by developer Kelly Wical and his team, this text engine is not just a limited set of rules and patterns like most grammar-checkers, or a database front-end that translates queries into SQL like most "natural-language" products. Instead, it's a full-fledged, multi-faceted natural-language system that takes up 40 megabytes of disk space. It comprises an English parser and a million-word lexicon that is the guts of the system. Language isn't simply grammar plus words; mostly, it's words with hundreds of rules about the way each can be used and combined with other words. For example, nouns are classified as animate or inanimate, singular, plural (or exceptions like water, sheep, fish, and so forth), just for starters. The average word-listing in the ALI lexicon contains 1000 pieces of information -- part of speech, appropriate prepositions, cases, tenses, special conditions, and so forth. The system also includes higher-level rules about tense agreement within sentences or paragraphs, sentence structure and other subtleties.

*PowerEdit, ALI's first commercial product, uses an 8-megabyte subset of this text engine. ALI describes PowerEdit as a writing analysis system. More than just a grammar-checker, it assesses text for clarity, conciseness and tone as well as correctness. It looks not just for canned words and phrases, like most grammar-checkers, but for higher-level patterns. We submitted a couple of sentences to it. Here's one, with PowerEdit's critique:

The basic, powerful capability it promises is to provide more structured, explicit, efficient representations of information than the original text.

```plaintext
*** MSG: The modifiers in this sentence make it hard to identify the main idea.

*** MSG: The word choice in this sentence keeps the reader at a distance from the action or process.

*** MSG: Sentences with too many subordinate ideas can be difficult to read. Can you clarify?

*** MSG: A word like "representations" followed by another word like "information" should be avoided. Try to put the action expressed by one of these words into a verb.

*** MSG: It may not be clear what the modifying elements in this sentence are modifying.

*** MSG: There are many modifiers in "structured, explicit, efficient representations of."

*** MSG: The actions in this sentence could be more directly expressed. "Nominalized" words such as "capability" and "information" express in nouns the actions that are normally expressed by verbs and adjectives. See 'Tutorial' for details.
```
We're extremely enthusiastic about ALI's underlying technology, but we wonder whether PowerEdit is the best first product to show it off and position it effectively. PowerEdit may have trouble differentiating itself (at $295) from most grammar-checkers (about $100) despite its superiority -- which may not be adequately representable in an ad. (Yet, it's an editor's tool, and it may get good editorial coverage.) We just think the product is worth more to a smaller, more appreciative audience than ALI seems to be seeking.

Down the road, ALI promises higher-cost, higher-reward (and smaller-market) applications for professional producers of documents -- everything from technical documentation and manuals to books, advertising, news copy, business letters and legal contracts. (It's the kind of system you could extend with content-oriented rules rooted in law, company policies, machine assemblies or any other knowledge domain.) The text engine's capabilities can be applied in a variety of different ways, from text categorization (where, as in grammar-checking, it competes with less costly approaches) to greater challenges such as text summarization, automated indexing and outlining, automated hypertext linking, translation and various forms of interpretation to provide data for databases, information models and news abstracts.

The basic, powerful capability it promises is to provide more structured, explicit, efficient representations of information than the original text. Just consider this famous remark: "I would have written it shorter if I'd had more time." ALI's goal is to help people do just that -- quickly.

DINNER: CAPITALISM WITHOUT GUILT

(Or, Did you know that you and your company are just a representation of intellectual property?) Michael Rothschild, author of "Bionomics," which you will receive at registration, will offer a new perspective on competition and economic evolution. The common assumption, even among many practicing capitalists, is that the rich got that way by taking from the poor. The last 200 years of economics has described the economy as a zero-sum equilibrium machine: If we have more, they must have less. In fact, Rothschild argues, capitalism is more like an evolving ecosystem -- with technology serving as its genetic code. When the market works (and it works more or less, not perfectly), the economic ecosystem grows because it is producing surpluses (a.k.a. profits). Profits are generally a sign of efficiency, not of exploitation. With growth, the rich get richer as the poor get richer; the rich and wanna-be rich build the companies that turn technology (economic genes) into products and new jobs to employ the poor. So come enjoy your dinner and flaunt, not your greed, but your profit motive.

On the other hand, Rothschild is no apologist for business in general. His theory of bionomics also explains the fate of the venture capitalists who invest in also-ran companies, whose "ecological" niches are already filled with stronger competitors. (Is GO such a case, or has it found a new ecological niche in which it can compete with Microsoft on its own terms?)

Just as living things are the embodiment of the information carried in their genes (and "fitter" genes win out in the long run), so are companies an embodiment of their intellectual property, which includes not just patents and copyrights but subtler forms such as policies, procedures and even attitudes and corporate memory. (Many groupware products reflect an implicit attempt to capture, represent and disseminate this intellectual property reliably.)
ACT II -- LEGACIES: OBJECTS AND TRANSACTIONS

"Legacy" is a mainframer's euphemism for old applications. It sounds much nicer, doesn't it?

The focus of the second day is application development, and the reuse in the process of existing modules of wisdom, models, objects, procedures -- i.e. legacies. The essence of an application is specifying routine procedures so that they can be executed by a computer, with different data, conditions and other parameters each time. You can't reuse a legacy unless you can find and define it and have a way of talking to it.

Objects are a way of representing data and procedures for easy reuse. They have well-defined ways of behaving, and protocols that they respond to. A user need not -- may not -- look at their insides, but should instead simply manipulate them. "Object-oriented" means a lot more than "written in Smalltalk." The speeches and discussions on Tuesday will discuss some of those divergent meanings, and their implications.

Yes, you can get there from here

For Bill Gates, object-orientation begins in the operating system -- and the operating system begins with Windows. He will talk about why you need object linking and embedding in the operating system, not just added on. This will help to solve some fundamental problems: What do you do when the object you're talking to isn't there? Or when someone upgrades the application? You need something with authority -- i.e., the operating system -- to monitor and manage the state of the whole environment, not just the objects at one desk (and not just the user's file system, for that matter). And of course, Gates will talk about why Windows is the best place to start from.

The name "Windows" is a cover story for a progression of changing, and improving, operating systems, starting with DOS and leading up to a portable, 32-bit system with all the fundamental capabilities of OS/2, including multi-threading and pre-emptive multi-tasking. Overall, the OS/2 and Windows conflict is overblown; people were hoping for a consistency that was not to be. Whereas UNIX has been converging, but not to a single implementation, OS/2 and Windows are diverging, but not from a single implementation. They already have different GUIs and different kernels. Microsoft's platform-independent, RISC-supporting "New Technology" will have yet another different kernel, but with capabilities beyond those now in OS/2, including object-linking and embedding, and supporting current APIs including OS/2 and Windows. Although IBM and Microsoft are cross-licensing their development work, consider the inevitable incompatibilities as an opportunity for third parties to provide insulating cross-platform tools and layers.

This is why we haven't planned (although we're sure we can't avoid) any formal discussion of operating systems: OS/2 vs. UNIX, Windows vs. OS/2, OSF vs. Sun (aren't they really just the same thing, with annoying little incompatibilities? They will retain their ability to bedevil developers, but to users the differences are on the level of VGA vs. EGA, or MicroChannel vs. EISA -- something you let the techies worry about. However, Sun and HP will run a joint presentation on their newly announced effort to create a grand unified object-oriented environment to hide all the incompatibilities among the Unices. This rapprochement could make UNIX a serious contender.

Release 1.0 25 February 1991
Components in commerce

Dave Liddle is chairman of Metaphor and of Patriot Partners, its joint venture with IBM. He will focus on the business issues of objects, since there isn't much public and specific yet to say yet about its vision of components and an environment that run across all the major operating systems. The existence of classes and objects will have a profound effect on how software is developed and ultimately on how it is sold. As illustrated to some extent in GO's PenPoint, and also in the proliferation of third-party libraries for C++ and other languages, the essence of object-orientation is incremental development from reusable components. One third party can use another third party's extensions. But how are these components paid for? Who certifies that they will work together? In fact, who describes them, rates them, helps people find the right ones? For Component Software (below) to be successful, for example, requires class libraries from third parties, much as The Whitewater Group is supplying for Windows.

---------------------
Development is the conversion of unexecutable legacies such as policies and procedures into code; re-engineering is the conversion of executable legacies into new executable forms. But frequently the functions and purposes of legacies are unclear. Good applications come both from a clear understanding of the routines, and from efficient representation of them for the computer. Neither people nor computers that do work by rote respond well when circumstances change, when there are problems or unexpected events.

---------------------

Philippe Kahn of Borland will give his prescription for the object-oriented operating system of the future -- the one in which his tools can best operate. Borland competes with Microsoft in the language business, but not in operating systems -- which Kahn modestly leaves to others. However, we expect Borland to work closely with Patriot on object-oriented languages.

Meanwhile, IBM has increasingly adopted the policy of "let the customers choose." This may be openness, or it may be lack of conviction. For now, as IBM vp and gm marketing & business development Joe Guglielmi will discuss, IBM supports a multiplicity of would-be standards, including DOS, OS/2, LAN Manager and NetWare, with PenPoint and Patriot's system on the way. IBM has recognized the value of third-party solutions. The big question for IBM over the next few years, then, is not the importance of letting customers choose, but the value of IBM's endorsement. Is letting customers choose appropriate open-mindedness, or is it abdication of responsibility? The technical decisions they make today will constrain them years from now.

Transactions, servers and objects

Just as objects encapsulate data, and the processes that may act upon the data, so do transactions, strictly defined, encapsulate a sequence of processes, and all the data they may act upon. In that sense, all servers are "object" servers; clients may call on them using specified protocols for data (objects) or to perform some work (transactions). Sybase, among others, calls these stored procedures.
Typically, stored procedures have been SQL transactions. Sybase has gone one further and provides facilities (in Open Server for CICS and Open Gateway for DB2) to encapsulate existing mainframe (IBM) transactions. Thus, instead of providing access to legacy databases, Open Server provides for reuse of legacy applications.

While most database vendors are just beginning to appreciate the value of SQL and to understand that databases are becoming commodities, Sybase's Bob Epstein is one step ahead. Rather than try to wrest SQL away from Oracle or even from IBM, which first developed it (which would be like trying to wrest the GUI concept away from Apple: It's so common now it has no distinguishing, added value, to Apple or to anyone else). Sybase is working on establishing the standard interface between clients and servers, called Open Client and Open Server depending where you start. (Neuron Data already has *Open Interface.) Sybase doesn't care what's on either side. Is SQL Server on the server? Or any SQL database at all? It might just as well be a filtered text server, an object-oriented database or a mainframe running CICS transactions. On the client side, likewise, it could be a mail tool, a traditional database front-end, a spreadsheet, or an object-oriented programming environment. In a sense, Open Server is a grand, unified protocol specification for objects; Open Client is a way to specify messages (or calls to the server).

---

**Why is a server like an object?** Both have protocols you use to talk to them, things they know how to do -- transactions or methods -- and data they manage and protect. Most pc-world people don't normally make the distinction between a simple application sequence and a transaction, just as people may not understand the difference between editing an item in a database and storing the result, vs. performing a database transaction. In one case, the system "knows" what has happened; in the other, the system is not fully aware of the change. In the same way, if you perform a specified procedure on an object, the system "knows" what you have done, and knows the "state" of the object; but otherwise, it doesn't. For example, in an editing system, you can move a set of lines over five spaces each, but then if you want to add a word in the second line, everything goes awry. Do it with the set margin command, however, and the system knows how to make the adjustments. Likewise, if you sum a group of specific cells in a spreadsheet, watch out when someone adds a new line! Better to use the closest thing a spreadsheet has to an object -- a named range.

---

Aside from these issues, Epstein is also thinking about the meaning of "integrity" in the real world. The real world consists of irregular data, intermittent connections and imperfect knowledge. So integrity cannot be absolute. The relevant question, then, is how to represent imperfect information and reconcile conflicts as they become known.

**Transactions: Now where was I?**

Defining a database transaction is no mean feat. Aside from the rollback and recovery issues of data integrity, there are higher-level issues of consistency and system integrity. Although the transaction may execute local-
ly, lots of other things happen not just to the data but to the state of the application and the system as a whole. A tool such as Cooperative Solutions' *Ellipse manages the whole process with all its intricacies, on all the clients and servers involved. These things don't matter much if you're changing someone's address in a marketing database or an appointment in a calendar, but they matter in a bank, a reservation system or most corporate on-line transaction-processing applications. "Users have moved from centralized to distributed for many decision-support applications," says Cooperative president Dennis McEvoy, who formerly ran software development at Tandem. "Our goal is to let transaction-processing be distributed as well."

Approaching transactions from a different point of view is Action Technologies' Tom White, with a fix on three kinds of transactions -- material, information and business. (See chapter 8 of Peter Keen's book, "Shaping the future," handed out at the Forum courtesy of ATI.) Material is, "Take the widgets out of bin A and put them into a box for shipping." Information is, "Invoice Mr. B for the widgets and debit his account by $50; credit our receivables by $50." And business is, "Did Mr. B get the widgets and is he happy with them?" All can be represented on a computer. While information transactions can be executed on a computer, the material and business transactions can only be represented in the computer; you need to rely on some sort of discipline and required procedures to make sure the outside reality is correctly reported. The first is too material; the second is too abstract. But in both cases, a proper understanding of the real-world transaction is a valuable guide to the software representation. All kinds of transactions, of course, can be composed of component transactions, with options, conditionals, parameters, and so forth.

Lunch: The visual display of qualitative information

Seeing is believing. And understanding.

As Andy Grove says, more information is better. But in the end, just as one uses a graph to see the meaning of a spreadsheet (Is the line going up or down?), it is also useful to have a way to summarize rather than embellish qualitative information. Hypermedia navigation tools and mapping schemes allow users to visualize not their data per se, but the structure of their data. Beyond that, there will be tools that provide the appropriate illusions (not metaphors, exactly) to help users feel (not command) their way through information. Examples shown by Steve Levine of Wang at lunch Tuesday will include turning pages of a virtual book, which a user can search by location rather than text strings. The result is that the user feels his information is on the screen, waiting to be looked at, rather than at the end of a wire, to be retrieved only through complex commands. "We give back the spatial domain to information," says Levine.

INTERMISSION: CLIENTS AND SERVERS

There is a proliferation of graphical, user-friendly, object-oriented (pick two more adjectives from column A) database front-end tools. There are so many, in fact, that most are hard to tell from one another. The vast majority are simply object-based or object-oriented programming tools for end-users, with some facilities for data access (through SQL, usually); a few are true database client-application development tools; and one is a full-
fledged on-line transaction-processing tool. (Object-based is what purists call a tool with object-oriented graphics but not the full OO feature set.)

The object-based user programming tools include forms builders and a variety of other tools. The most interesting is probably Borland's ObjectVision, which isn't especially object-oriented but is especially nice. Just as Paradox made it easy to visualize database structure and ask queries, so does ObjectVision make it easy to visualize the structure of a front-end application, with its display of the decision tree it implements. (But, notes Borland, to design the database itself you're wiser to use Paradox.)

**Power in pictures**

Then there are the true database front-ends, or clients. They perform the functions of Paradox, above, but with Windows and objects and other new features. These tools understand about databases, and are generally focused on letting users build client applications. They include Gupta's SQL Windows (see Release 1.0, 12-89) and, most impressively, PowerBuilder from Powersoft. PowerBuilder not only lets you build a front-end application; it helps substantially with the data design, integrity checking and the like. Its DataWindow retrieves, displays and updates data; Data Manager lets you create data tables, link them, set indexes and keys, and visually describe relationships among various fields and tables.

We saw a beta installation of PowerBuilder at American Airlines, where it is being used to schedule and manage flight training -- a complex task that involves thousands of pilots with various training requirements, several hundred instructors, a number of training centers, courses, and equipment and room requirements. (Previously, pilots had a tough time finding when they were scheduled, and course managers had to pore through pages of print-outs to figure out what was going on. With scrutiny, you could figure out when a particular room was being used, but it was almost impossible to be sure you had found a free room if you wanted to schedule a new class.) The system doesn't automate the scheduling (although an expert system could easily be attached) but makes the schedules easy to manage -- and allows pilots to see where they stand in terms of schedules and requirements. The American people are delighted not just at how easy the applications are to use, but at how easy they were to develop.

Powersoft used to be called Computer Solutions; we wrote about them in 1986, when they had just launched GrowthPower, an HP-based expert system for distribution and MRP applications. It's still going strong. The company wanted to develop a similar system for Windows and SQL Server, but lacked a development environment. They hired Dave Litwack, formerly at Cullinet, to build one, and liked it so much they're marketing it as PowerBuilder.

**Co-opera**

Finally, there's an industrial-strength client-server tool such as Cooperative Solutions' Ellipse, which is being announced at the Forum. It builds transactions that execute on the server or across clients and servers, for that matter (determined at installation time by the application-partitioning function). In PowerBuilder, you manipulate data from the workstation, operating on subsets of the database, whereas with Ellipse you launch your transaction from the client, but the system manages a complex, potentially multi-step process that may comprise any number of clients and servers.

*Release 1.0* 25 February 1991
Ellipse isn't as pretty as, say, PowerBuilder. It's focused on the functionality of the transaction rather than the user's look and feel. The programming environment is "graphical," with forms, buttons, menus and tables, but more effort is focused on defining, clarifying and validating the steps involved in a transaction, with a nice outline feature that lets you see the major steps in a transaction to follow the flow of events easily. It can work with a heterogeneous assortment of clients and servers, including different vendors' (SQL) databases. It's a full development environment, with support for multiple developers, automatic version management and other scale-up capabilities.

The software is implemented in OS/2 -- perhaps not the decision Cooperative would have made today, but not a bad way to position it apart from all the Windows-based user-programming and database-query tools. (Cooperative notes that OLTP applications for relational databases such as Oracle and Sybase are currently being written by hand in COBOL or C, not in the vendors' or anyone else's development tools.) The Ellipse Production System is the client/server management system, which runs on each client and server involved. It performs the same tasks as a transaction-processing monitor provides in the mainframe environment -- tasks such as performance optimization, concurrency management and error recovery -- all invisible functions...except when they don't work. It also performs an important financial task for Cooperative, which collects money for each server, like a dbms company, rather than for each developer, like a CASE or other tools company.

The world can be consistent only at the very beginning or at the very end.

Database helpers

Reusing software and data makes sense only when the original is worth reusing, or can be cleaned up effectively. That's basically the challenge addressed by DB Software's RE/generator. It's a database re-engineering tool that doesn't just simply allow you to transform database designs from one software medium to another, but lets you clean up the data and look at it statistically to determine if your data structures accurately represent your data. Call it data-driven, as opposed to dream-driven, design.

Red Brick Systems is best known for its database query tools, one of which was licensed to Claris (Claris Query Tool). Founder and technical wizard Ralph Kimball came from Metaphor Computer Systems, and built much of the functionality of Metaphor's query tools (but not its user programming) into a Mac. However, the easier you make it for users to ask queries, the more likely that they will do so -- and tie up your server. To solve this second problem, Kimball has developed a patent-applied-for query-optimization technology, *Redbrick, that can dramatically reduce execution times for a number of commonly asked types of queries. (There's no support for updates or transactions; this works for queries only, although you can do any calculations you please on the data you retrieve.) Redbrick reflects Kimball's years of dealing with Metaphor's query-happy end-users. The company is loath to explain the technology fully, but it works by pre-indexing the data by a variety of techniques.
Like RE/generator, Redbrick works with actual data, not just database structures, and optimizes according to both the particular schema involved and to the relative sizes of the individual tables and distribution of the data. Different indexes and joins are pre-built at the time of indexing rather than at runtime, generating, Red Brick says, "a new class of patent-applied-for data structures that are a hybrid of conventional hash tables and vector processing schemes." These data structures eliminate the resource costs of processing constraints; instead, the query "cost" is determined by the number of records selected -- so that highly constrained queries may actually cost less because they retrieve less. For example, each of the following would take the same time: a complex wild-card substring match, such as "firm name includes 'goskom'," which returns 20,000 records; a simple numeric comparison, such as "revenues < $50 million," which returns 20,000 records; and a join, such as "product = banana rice baby cereal" and "location = Western region," which returns 20,000 records.

To give one example of a vector scheme, just part of what Redbrick does, consider the table on pages 28 to 29 of this issue. We could represent the same information the most common way, by listing the name of each company followed by a list of the features it addresses: for example, "Red Brick, database, query optimizer." A shorter, compressed way of representing the same information would be "Red Brick, 00000001, query optimization." But to find all the companies with a database aspect, you would still need to look through the whole list to find all the companies with a 1 in the eighth field. To make that query much faster, you would list each feature, followed by 1s and 0s representing which companies embody that feature. Then you could retrieve just those companies, rather than looking at the data for each one. This isn't the whole of it, of course, but it tells as much as Red Brick wants to.

The result, says Kimball, is dramatic performance improvements, although they can't be expressed as a single figure. In the example above, the longer the list of companies and the shorter the list of positive responses, the greater the differential. If, for example, you wanted the one individual in your company with skill in Rumanian, the response would be blindingly fast. On the other hand, if you wanted hire dates and salaries for all employees, Red Brick wouldn't help at all, since that involves retrieval rather than selection. The other caveat is that the indexing takes a while, and so this is basically for queries where the data must be current but not up-to-the-minute. (Don't use it to hand out money during a run on your bank!)

WORKFLOW GROUPWARE: VARIATIONS ON A THEME

As we've noted, groupware comes in two flavors (which may be combined): information-sharing, where ideally the software manages the information content and helps different users add, manipulate, share and present the information; and workflow, where the software manages the flow of the work and leaves management of the content up to the users. (See Release 1.0, 11-90.) Workflow has always been possible on a mainframe; the issue is how easy it is to visualize, interpret, implement and change automated workflows.

We have asked the companies doing demos not simply to show scenarios -- Alice sends a memo to Juan, he approves it, then depending on the amount it goes either to credit-check or payables... Instead, we'd like to assess the
tools used to create and manage the applications, and see how well they integrate with the environments that manage the application content. Accordingly, the scenario goes like this: Alice, a travel manager, decides that she wants all plans for foreign travel to be reviewed by Juan, the company security expert. How does she go about restructuring the automated workflow? She also wants to query the system to determine the proportion of travel requests over the past six months that involved foreign travel, and how many foreign-travel requests are currently outstanding.

In response, you'll see approximately the same scenario as interpreted and embellished by five different companies and their workflow tools. Action Technologies will demo an unannounced workflow design, management and monitoring tool, the first implementation of its business design technology. This will be the core of ATI's future efforts, now that it has handed further work on MHS (Message-Handling Service) over to Novell. Keyfile focuses on image management as well as rule-based, fill-in-the-blanks workflow; clients run in Windows 3.0, and servers in Windows or OS/2. This demo uses forms developed with a tool from Bloc Publishing. AT&T co-developed the workflow tool within Rhapsody (a full office automation environment) with Workhorse Systems of Dublin, Ireland. The tool is basically a UNIX server system which drives Rhapsody's NewWave clients (or potentially other vendors' Windows clients); it doesn’t use NewWave's automating agents directly, but it can talk to or launch them, as well as manipulate NewWave objects. Calypso from Systems in Concert sports the most elegant visual programming tool for defining and examining workflows, and lets you change a workflow by drawing the change (and then programming in a new box. NCR's Cooperation is a workflow tool that works within Windows 3.0; it visualizes workflows as sequences of icons and is based on technology developed at MCC.

Reach Software, while it does not have a workflow tool yet, is currently developing one which will use its newly announced MailMAN e-mail client as a front-end. The new system will use a standard database and scripting engine to turn otherwise ad-hoc e-mail interactions into structured groupware. Founder Anand Jagannathan, also a co-founder of Banyan, will provide something of a preview, focusing on groupware as implemented by the transport of objects through a workflow sequence.

CROSS-PLATFORM: TOOLS AND APPLICATIONS

While big vendors tout openness, little vendors provide tools to overcome the inconveniences of proprietariness.

The fundamental disagreements around PenPoint concern incompatibility: Is it just a disadvantage, or is it the necessary cost of innovation? The "disadvantage devoutly to be overcome" camp is represented by a number of tools and applications. They are concerned with making it easier for developers to put their applications on multiple platforms. Generally, the developer has to abandon his old tools and environments, and the specific, proprietary advantages of any particular platform in return for a larger target market. Customers may get better applications because vendor resources aren't spread thin over multiple platforms -- or they may get the same old stuff because it's easier to port than build something new.

There are three portability tools at the Forum, one concerned with interface, Neuron Data's Open Interface; one more concerned with the underlying
code, Hunter Systems' XDOS; and a full-fledged object-oriented development/execution environment, Component Software's *Component WorkShop.

*Open Interface* lets you develop a graphical user interface once, using Neuron Data's tools and widgets, and then implement it on your choice of systems -- Mac, Windows 3.0, Presentation Manager, VMS (Motif) and UNIX with Motif or OpenLook. The assumption is that your application consists of a high-level language and the interface, so all you need is Open Interface and the native graphics library (e.g. Xlib for UNIX or the Windows library) plus a C compiler for the target machine. Open Interface consists of the layout editor used to draw the interface and a set of libraries for each target environment. The layout editor automatically generates the portable code to generate the interface, so that all you have to do is link in the application functionality for each platform.

Neuron Data has used the pre-commercial version of Open Interface in developing 30 different versions of its base product, Nexpert Object, which has given it a good understanding of the issues involved. Open Interface competes with the toolkits of each of the interface vendors and keeps developers from feeling tied to a single platform; on the other hand, it lowers the barriers to movement among platforms. Altogether, it shifts the balance of power slightly towards the less popular interfaces and should make the market more efficient overall. It's part of the trend we see not so much to less proprietariness but to less *effective* proprietariness.

Hunter's XDOS: Line extensions for DOS vendors

While Open Interface works automatically and generates environment-sensitive applications for people using its development system, Hunter Systems uses an ex-post-facto porting tool, *XDOS, to create UNIX versions of existing DOS applications. The process is not entirely automated, but rather tool-assisted, and produces fundamentally the same applications with the benefits of the UNIX operating system -- multi-tasking, large memory access, and so forth. They end up looking and acting just like the DOS originals (although, since the process isn't entirely automated, you could tweak things during the transformation as much as you wanted to). Unlike Neuron Data, Hunter doesn't generally sell its tool, but rather sells the resulting products. Long run, it hopes to play NutraSweet, enabling software vendors to offer line extensions much as Coca-Cola offers Diet Coke. (To extend this analogy, Tab would be Jazz, Lotus’s ill-fated spreadsheet for the Mac.)

Thus, Hunter offers UNIX versions of XyWrite, MultiMate, DataEase, Brief and the Japanese 1-2-3 and Japanese dBASE III. Some other, better-known names, including Borland's Quattro Pro, are on the way. Hunter works in conjunction with the original vendors in selling the ported software; typically, says founder and chairman Colin Hunter, "The salesperson gets a big order for the DOS version, and then the customer says, 'By the way, we have some UNIX machines...'. That's when they call us in, and we work with the customer to get the proper configuration, install it and so forth." For its troubles, Hunter gets a substantial share of the retail price; exact terms vary from vendor to vendor.

Objects and components

While people complain that it will take Patriot Partners two or three years to deliver its object-oriented cross-platform environment, would they be
Component Software is planning beta release of such an environment, *Component Workshop, for next September. Component Workshop could be called a cross-platform tool, or an object-oriented development and runtime environment, based on Objective-C with C++ support promised. Rather than offer portability by building an application for any of several environments from a common set of source files, as Open Interface does, Component Workshop creates its own environment on Macs, UNIX and Windows machines, sitting above the resident operating system (not replacing it). It delivers complete portability across platforms -- Windows 3.0, Mac and SPARC UNIX OpenLook -- by way of a virtual machine layer. (This layer is not an interpreter; it executes compiled Workshop code.)

Why be platform-independent if you have to be environment-specific? (This is not a question unique to Component Software, of course.) A new environment makes a visible difference in the applications it supports -- as well as in the lives of developers -- to be successful, we believe (cf. PenPoint). Is an object-oriented development environment enough? It will be competing with the best tools on each individual platform. Portability always seems to be a secondary worry, after a developer has been successful on the first platform. (Is the market for the new environment overall as large as that of any single platform it runs on?)

The concept has a lot of appeal, but Component Software faces intimidating marketing challenges even if it works flawlessly. Absolute portability, of course, eliminates the benefits of any particular platform, such as DDE (to say nothing of object linking and embedding) within Windows, and all the other tools and features created specifically for each environment. In the long run, the argument goes, equivalent facilities will be created within or for Component Software's Component Workshop environment -- leaving the OS vendors with no way to differentiate themselves.

Component Software co-founder Tom Stambaugh points out that you can talk to other environments through foreign-function interfaces or encapsulation, for which the Workshop promises full-featured support. In fact, it may be the Workshop's foreign-function interfaces -- that is, the interoperability -- that's the real winner, not the platform independence, which requires that you adopt the new environment completely to benefit.

Component Software, incidentally, is ON Technology two years on; it was founded by Stambaugh and Stoney Ballard, who led ON's project to develop essentially the product now nearing beta release in September by Component Software. Mitch Kapor, in charge at the time, couldn't get the necessary support from hardware vendors. Without Mitch, but with a close to completion product, will Component Software fare better? It has the marketing challenge of attacking existing environments head-on, and it doesn't have a clever differentiator, such as gestures and handwriting recognition -- or a new market segment to address, people away from their desks. It addresses a far larger but more crowded market -- most of the world's desktop machines.

Release 1.0

25 February 1991
Whatever the particular fate of Component WorkShop or similar-minded systems from Patriot Partners, ParcPlace Systems, HP and Sun (unified UNIX), GO and Microsoft, object-oriented platforms will foster an extensive array of third-party components. The components will play the role of applications for operating systems, but they should work together more smoothly since they will be built from common (killer) class libraries. A primary value of object-oriented environments is the establishment of standard protocols so that applications will work together at the right level -- not file-sharing, but function- and object-sharing. Whereas applications generally work together through only a small set of hooks, such an object-oriented environment would let them all share data, simple functions such as those of text-processing, and so forth. But it's not magic, and it doesn't obviate the need for standards: Someone has to build an environment and components that are worth reusing -- something good enough to become a standard.

Starting with the components instead

An incremental, less ambitious approach than either GO's or Component's is simply to build components within an existing environment and to offer some interoperability -- regardless of the compromises that implies. The result is generally more marketable, although there's less chance of changing the world. Two companies following this approach are The Whitewater Group with a development tool/language, Actor, and class libraries for Windows 3.0, and Clarity Software with Rapport, sort of a "UNIX Works" -- a suite of the standard office automation tools, at the level required by the current market and allowed by current technology. Rapport works across a wide range of UNIX platforms and offers easy integration with many PC and Mac applications; The Whitewater Group is focused on Windows.

Clarity's Rapport includes wp, e-mail, spreadsheet, graphing, drawing, slide management and data access, enhanced with voice annotation, easy transfer of data from one function to another, and openness in the sense that you can replace most of Rapport's application modules with favorites of your own -- or build new ones.

Rapport is not a development environment with the ambitions of Component WorkShop, but it has hopes of reaching a far broader audience. And underneath there's a fairly sophisticated object-oriented application framework that might be used by outside vendors someday. This object-oriented framework will make it easy for Clarity and others to add new functions down the road, and is perhaps its best feature in the long run.

For now, Rapport acts both as an OA application suite and as an integrator across applications, environments and users with high-end mail and reformatting capabilities. It offers platform independence across a wide range of UNIX platforms (with both Motif and OpenLook), and file translation (one step down from interoperability) with a wide range of Mac and pc applications such as 1-2-3, Excel, Write Now, WordStar and WordPerfect.

Rapport is not a port of any existing application suite, but is rather an integrated reimplementation of the current range of capabilities enhanced with mixed-media and based on an object-oriented, extensible framework. Its functionality exploits the UNIX platform and reflects current assumptions.
about the need for inter-application connectivity, embedded objects and other new features appearing now in Windows 3.0 and shortly in the Mac System 7. However, it's not fundamentally new. As the company's own literature puts it: Rapport "allow[s] workstations to do everyday work, [but] with tools that take advantage of powerful workstation capabilities such as audio, imaging and communications."

In other words, it represents just about the best a reimplemention can achieve -- and by the way, it's a set of probably the handiest UNIX tools around. Its presence should help UNIX gain a firmer hold on the desktop market simply by eliminating the need for a Mac or PC nearby to do the everyday work.

Whitewater runs objective

The Whitewater Group is working at a lower level, providing Windows-based application components implemented as Windows objects. Rather than trying to convert developers from one language to another (let alone one platform to another), Whitewater is offering fundamentally the same class libraries implemented in three languages (Pascal, C++ and Whitewater's own Actor). All three versions of both the products -- *ObjectGraphics for graphical objects such as squares, circles and lines or for creating your own GUI objects such as maps, *ObjectWindows for application objects such as menus, tables and dialogue boxes (with appropriate behavior) -- follow the same object model, which Whitewater and Borland have submitted to the Object Management Group as a spec for class libraries. The underlying model, of course, is platform-independent; someone so minded could use it to build class libraries for, say, UNIX, C and Motif or OpenLook.

What are the benefits of such a model? Basically, it gives people working in different environments a framework for interoperability. You can count on the objects doing what you ask them to do, in their own way in their own environment. In other words, the framework operates as a sort of interface: Write to this interface, and the object on the other side will perform as expected. (See why objects are like servers?) The actual language matters only to the developer who may extend, modify or reuse the objects, while the execution platform matters only to the ultimate users.

------------------------

Release 1.0 is published 12 times a year by EDventure Holdings, 375 Park Ave., New York, NY 10152; (212) 758-3434. It covers pc's, software design, groupware, text management, connectivity, artificial intelligence and intellectual property law. A companion publication, Rel-EAST, covers emerging technology markets in Central Europe and the Soviet Union. Editor & publisher: Esther Dyson; associate publisher: Daphne Kis; circulation & fulfillment manager: Lori Mariani; executive secretary: Denise DuBois; editorial and marketing consultant: William M. Kutik. Copyright 1991, EDventure Holdings Inc. All rights reserved. No material in this publication may be reproduced without written permission; however, we gladly offer reprints or bulk purchases. Subscriptions cost $495 per year, $575 overseas.

25 February 1991
ACT III -- NOTEPADS: A NEW ENVIRONMENT, OR A PEN OFF THE OLD BLOCK?

This is the year that notepads become a category, separate from pocket DOS machines with keyboards and electronic organizers with keypads. Unfortunately, it will be the year of notepad announcements more than deliveries, but that just gives the software vendors a chance to get ready.

The notepad will be a handy little machine you can carry around with you; it will manage your schedule, handle your mail, manage your mobile work and generally keep you in touch with your home base. Under the covers it's a computer, but it doesn't feel like one -- no keyboard, no strange command sequences to learn, no cryptic error messages if you miss your lesson and type something wrong. Yes, it really feels like an electronic notepad; it can even read your handwriting.

The visibility leader in the field, GO Corporation has announced PenPoint, the software that will make such a notepad possible, and a number of hardware companies, including IBM, NCR and GRID, have announced that they will make notepad machines to run PenPoint. Meanwhile, Microsoft has announced Pen Windows, a version of its best-selling Windows program designed to run on notepads and read handwriting -- but otherwise similar to its deskbound Windows. The NCR and GRID machines will also run Pen Windows.

Territorial imperatives

The "pen-based" battle is mispositioned: Commentators focus on the handwriting recognition capabilities of these systems and neglect the software that underlies everything. It's as if one were to describe the Macintosh phenomenon as "mouse-based computing." The battle isn't over notepads or handwriting recognition, but over the next generation of computers.

Microsoft wants to position pen-based computers along its own operating system spectrum; its Pen Windows extensions to Windows accommodate handwriting and pen "gestures" -- flicks, cross-outs, checkmarks and the like. Microsoft is offering the sensible solution. For developers, it's almost no work at all to make their applications pen-aware; Microsoft's Pen Windows generally does all the work, translating gestures into commands (delete this, move that and so forth) and handwriting into computer characters (with any errors correctable by the user). For users, the advantage is the ability to keep on using the same old applications: Anything you did on your desktop PC, you can keep doing on your Pen Windows machine.

GO's answer to this is yes, but. Yes, you can, but why bother? It's as if you went to Paris and found the nearest McDonald's because you didn't know the words for chateaubriand and coq au vin. If you were moving to another keyboard system, sure, you wouldn't want to change anything (familiarity breeds comfort), but with PenPoint you can use a new more efficient "language" that enables you to enjoy the system's unique advantages.

PenPoint's goal is to be not so much different-from as better-than. Sure, you can get at your existing data -- spreadsheets, customer records, documents, pictures and diagrams -- but do you really want to do the same old
things with them? Wouldn't it be nicer to have your machine do more of the work automatically? To extend the language metaphor further, wouldn't it be nice to have a software waiter who could figure out what you wanted without your expressing it explicitly? You get the same food, but more elegantly cooked and without having to instruct the chef in a foreign language.

PenPoint doesn't fully achieve this, but it's heading that way. It provides the underpinnings for software developers to put application-specific intelligence into their software. (You tell the waiter you want beef, and he comes up with some appealing suggestions.) Intelligence in this context can't be general; the system has to know roughly what the user is trying to do, and then it can make a reasonably good guess.

Can you make chateaubriand from the same meat as hamburger?

Of course, developers could do the same with Pen Windows, but with more difficulty; they'd just have to do more of the work themselves. Microsoft now offers a credible implementation of the ideas in the Mac (all of them, not just the mouse) with its Windows 3.0, some six years later. In due course, PenPoint-like facilities will be available in Pen Windows, and perhaps sooner from Active Book, Momenta and Apple. The advantage of PenPoint is that there's a single class library model that all developers will use. (Consider the problems with incompatible graphics until everyone finally rallied around Windows; in this case, PenPoint has a chance of being the standard.)

By its very newness, PenPoint will foster the development of a new style of applications. Easy-to-adopt systems that let you reuse Windows applications discourage the sort of innovations that PenPoint will encourage. The argument is pragmatic rather than technical: People who take the trouble to start fresh are more likely to do truly new applications than those who take the easy, cost-effective, sensible route and port existing applications.

In the meantime, Microsoft is making it easy on developers to serve leftovers, so why should they do French cooking from scratch? Because of the ease and speed of porting, there will quickly be many more Pen Windows than PenPoint applications out there. The folks from GO just hope there's a big enough audience for PenPoint cuisine before Pen Windows catches up.

Kathy Vieth of IBM runs the independent business unit IBM created to make PenPoint hardware and market the product. The GO-IBM alliance represents GO's sensible attempts to gather as many allies as it can. Meanwhile, IBM considers notepads a new category in which it has a chance to make a difference by backing a newcomer rather than a de facto standard.

Hermann Hauser of Active Book will discuss his company's approach and his still unannounced *Active Book* penware platform. It uses a book metaphor (pages, tables of contents) along with a semi-standard operating environment: Smalltalk. That means that you can port applications back and forth to desktop Smalltalk systems. On the other hand, primarily for performance, it uses a non-standard RISC chip built by Acorn Computers, a company Hauser co-founded and the vendor of the highly popular (in England) BBC microcomputer. Are they smarter over there in England, less blinded by standards? Are notepads Smalltalk's big chance -- fulfilling Alan Kay's Dynabook dreams after all? Or is Active Book just another interesting effort?
Not yet announced are penware platforms from Momenta and Apple. Dave Nagel, vp of the Advanced Technology group at Apple, will appear on the panel, mostly to talk about the role of communications in pen-based systems, but he will have little specific to say about Apple. Momenta, closer to introduction, is reluctant to say too much, and will be in the audience in the person of John Rizzo.

INTERMISSION: PENPAL SOFTWARE

PenPoint's success depends in large part on its ability to attract and support independent software vendors who will build unique tools and applications not available on any other platform. While a number of big-name vendors have announced support for PenPoint, the killer app (or killer lib) is more likely to come from a relative unknown (cf. PageMaker for the Mac or even 1-2-3 for the PC).

For now, the two leading contenders are Slate Corporation and PenSoft Corp. Slate will show PenApps, the first of many products, as the company plans to be a publisher of other vendors' pen applications. Slate wants PenApps to play something of the role of dBASE -- the tool that made the PC a development and deployment platform for thousands of data-oriented applications. The PenApps end-users, mostly mobile employees or agents in "fleets," are a different group from dBASE end-users -- which is a good thing, since PenApps is after new customers, not a replacement market.

PenSoft will show its PIM (PenPIM?), sort of a development tool for end-users, which will allow them to define, structure and manage their own information. Of course, the tool has a lot of functions (class libraries and methods) to help them get started, including calendar, address book, memo, and similar features. But the system is extensible; for example, you could create your own calendar, say, and build a separate one for your guest room, to manage who was staying there and when it needed cleaning. Then you could link the people staying there to information in the address book, and automatically generate messages about the lamp someone broke (or, more positively, thanks for the lovely flowers they brought when they arrived).

PENWARE PLATFORMS

Eden Group will be showing its VPi386, a 4.4-pound 386sx notepad computer with screen-input. It can support DOS applications such as AutoCAD's AutoSketch (with appropriate drivers for screen input or virtual keyboards) or Pen Windows, both of which will be demonstrated. (Since Microsoft is not doing a company presentation, this is your chance to see Pen Windows up close.) It also runs its own Virtual Paper Interface (with a real-time kernel) to handle fax and e-mail. It does not "interpret" these messages, but allows the user to store, file, annotate and otherwise manipulate them. Certainly, its capabilities would be complementary to those of Beyond Mail or Wijit, along with OCR for the faxes. Finally, the system will be used for a demonstration of Nestor's real-time-learning handwriting recognition.

GRID, which will eventually support both Pen Windows and PenPoint, will demonstrate its own GRiDPAD development and runtime system, showing how its tools can build an in-store merchandising system (as used by Kellogg's to count cornflakes and other delights). John Patterson, from GRID's parent Tandy, will stand by.
Another ancillary demonstration will be ParaGraph's ParaScript, developed by Shelia Guberman, which recognizes cursive handwriting. ParaGraph is a Moscow-based joint venture that knows how to compensate for low-end hardware; this demo, however, will run on the Toshiba 386 laptop of ParaGraph's general director, Stepan Pachikov.

**ENTR'ACTE: FISHY THINGS ON THE SCREEN**

In many ways, Tetris was the piece of software that first won respect for the Soviet Union's computer expertise from anyone but spies and scientists. Seemingly simple-minded, like checkers or the Japanese game of go, Tetris nonetheless captures the attention of ostensibly intelligent people for hours. At first glance, you couldn't imagine anything more different from Tetris than *ElFish* (for Electronic Fish), the product Tetris developer Alexei Pajitnov and his partner Vladimir Pokhilko will demo at the Forum, but there's the same underlying intelligence. ElFish isn't really a game; but an intellectual exercise akin to SimCity. It's a fish construction set, complete with an electronic ocean and seaweeds for them to swim around in. In fact, it's not even a construction set: It takes the proper biological view that fish are grown, not constructed. The user selects the parent fish, thereby specifying the gene pool, and the program, randomly selecting one of each parent's two genes for each characteristic, takes it from there.

That's the basic idea -- like Tetris's, very simple. But life, unlike falling blocks, is complex in implementation. It takes four to five minutes to grow a fish, selecting the genes and generating their expression in a realistic 3D fish that can swim and move around in the water. (It takes weeks or months in the sea!) It also takes a lot of memory, and four megabytes of disk to store the program and fish genes, and runs on OS/2.

As an OS/2-based game, ElFish sounds like a marketing disaster. But as an OS/2 artwork, it's a neat concept. It makes most graphics look vulgar (including those of Tetris). It exploits OS/2 to do what OS/2 is best at -- background processing while you do something useful in the foreground. And when you're not working, you can exercise your screen with the most naturalistic electronic aquarium around -- secure in the knowledge that your fish, just like your children, are unique.

The product will be marketed by Bullet-Proof Software; pricing and positioning aren't clear yet. We can imagine a vigorous aftermarket for exotic seaweeds, sealed stones, and new strains of fish. In the long run, of course, there's no need for the concept to be limited to fish -- although it took a lot of work to get the fish to swim right, and animals with legs or wings would require different rules for behavior even though the fundamental structure of the new program (ElBird, say, or ElFarm) would stay the same.

**BEYOND THE DESKTOP: THE ELECTRONIC FRONTIER**

John Perry Barlow is funny the way Steve Levine from Wang is visual; if we could be as funny on paper, we wouldn't have asked him to speak. He will explain the motivations behind the Electronic Frontier Foundation, which he co-founded with Mitch Kapor, and a lot else. Come listen, and take part in defining the rights and responsibilities you want as a settler in the electronic frontier.

*Release 1.0*  
*25 February 1991*
# DEMO-PICKER

This intuitive guide, despite its lack of a graphical user interface, should help you pick company demonstrations of particular interest to you. It indicates which categories each company might be found under in a topic database, and will appear in various forms in several of the text-oriented demos. These classifications are not rigorous. The topics are: mail & groupware, UNIX, Windows 3.0, text tools & filters, pen-based platforms & tools, object-oriented tools & applications, cross-platform tools & environments, database tools, and one key phrase for each company/demo.

<table>
<thead>
<tr>
<th>Company/product</th>
<th>mail/g'ware</th>
<th>UNIX</th>
<th>W 3.0</th>
<th>text/ filter</th>
<th>pens</th>
<th>o-o</th>
<th>X-plat</th>
<th>db</th>
<th>key phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Tech workflow</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>bus. transactions</td>
</tr>
<tr>
<td>Agility Wijit</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>mail agents</td>
</tr>
<tr>
<td>ALI PowerEdit</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>text engine</td>
</tr>
<tr>
<td>AMIX AMIX</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>markets</td>
</tr>
<tr>
<td>AT&amp;T Rhapsody</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td>NewWave workflow</td>
</tr>
<tr>
<td>Beyond Beyond Mail</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>mail agents</td>
</tr>
<tr>
<td>cc:Mail cc:Mail</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>client-server mail</td>
</tr>
<tr>
<td>Clarity Rapport</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td>“UNIX Works”</td>
</tr>
<tr>
<td>Component SW Component WorkShop</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td>o-o platform</td>
</tr>
<tr>
<td>Coop Solutions Ellipse</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td>db transactions</td>
</tr>
<tr>
<td>DB Software RE/generator</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td>re-engineering</td>
</tr>
<tr>
<td>Desktop Data NewsEDGE</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>on-line news</td>
</tr>
<tr>
<td>Echelon LONworks</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>distributed smarts</td>
</tr>
<tr>
<td>Eden Group VPI386</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td>notepad comm’s</td>
</tr>
<tr>
<td>GRiD GRiDPAD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>notepad &amp; tool</td>
</tr>
<tr>
<td>HP/Sun joint effort</td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td>*</td>
<td></td>
<td>unified UNIX</td>
</tr>
</tbody>
</table>

*Release 1.0* 25 February 1991
<table>
<thead>
<tr>
<th>Company/product</th>
<th>mail/g'ware</th>
<th>UNIX</th>
<th>W 3.0</th>
<th>text/filter</th>
<th>pens</th>
<th>o-o</th>
<th>X-plat</th>
<th>db</th>
<th>key phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hunter XDOS</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>DOS—&gt;UNIX</td>
</tr>
<tr>
<td>Individual First!</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fax news</td>
</tr>
<tr>
<td>InTech/BPS ElFish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OS/2 fish on screen</td>
</tr>
<tr>
<td>Keyfile/Bloc Keyfile</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>image workflow</td>
</tr>
<tr>
<td>MCC gIBIS</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>hypertext</td>
</tr>
<tr>
<td>NCR Cooperation</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>Windows workflow</td>
</tr>
<tr>
<td>Neuron Data Open Interface</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>portable interfaces</td>
</tr>
<tr>
<td>ParaGraph ParaScript</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>script recognition</td>
</tr>
<tr>
<td>PenSoft PenPIM</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>PenPoint PIM</td>
</tr>
<tr>
<td>Powersoft PowerBuilder</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>db tool</td>
</tr>
<tr>
<td>Reach Networks REACH</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>text “broadcasts”</td>
</tr>
<tr>
<td>Reach Systems MailMAN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>mail &amp; beyond</td>
</tr>
<tr>
<td>Red Brick Redbrick</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>query optimizer</td>
</tr>
<tr>
<td>Slate PenApps</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>db for penware</td>
</tr>
<tr>
<td>Systems in C Calypso</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>workflow visualizer</td>
</tr>
<tr>
<td>Whitewater Object...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>class libs</td>
</tr>
</tbody>
</table>
RESOURCES & PHONE NUMBERS

Name, Organization (product or topic), country code if not in US (city code), phone, issue of recent relevant coverage in Release 1.0, page number in this issue

*means the product has not previously been shown or announced publicly.

Tom White, Action Technologies (*business design technology/workflow, MHS), (415) 521-6190; 11-90; 6, 15, 19
Hermann Hauser, Active Book Company (*Active Book), 44 (223) 843-131; 25
John Warnock, Adobe Systems (*cross-environment communications), (415) 961-4400; 8-89; 6-7, 9
John Landry, Agility (*Wijit), c/o Dunn & Bradstreet Software, (508) 370-5800; 10-89, 11-90; 5-6, 26
Phil Salin, Gayle Pergamit, AMIX (*American Information Exchange), (415) 856-1234; 7-90; 1, 3, 5
Dave Nagel, Apple (communications & notepads), (408) 974-5124; 1-91; 4, 25-26
Kelly Wical, Doug Kramp, Artificial Linguistics (*PowerEdit, *text engine), (214) 720-7297; 6, 9-11
Carl Pavarini, AT&T (Rhapsody), (201) 898-6070; 4-90; 2, 19
Chuck Digate, Beyond (Beyond Mail), (617) 621-0095; 10-89, 11-90; 2, 5-6, 26
Frank Millman, Bloc Publishing (forms technology), (305) 567-9931; 2, 19
Philippe Kahn, Borland (Philippe's dream OS), (408) 439-1805; 11-90; 7, 13
Michael Rothschild, Cambridge Meridian Group ("Bionomics"), (415) 454-1800; 7-90; 11-12
Philippe Courtot, cc:Mail/Lotus (cc:Mail), (415) 961-8800; 11-90; 2, 6
Howard Smith, Clarity (Rapport), (415) 691-0320; 22-23
Tom Stambaugh, Component Software (*Component Workshop), (508) 371-9104; 13, 20, 21-22
Dennis McEvoy, Kim Worsencroft, Chris Grejtak, Frank Clugage, Cooperative Solutions (*Ellipse), (408) 377-0300; 6-89, 1-90; 2, 15, 16
Andy Zimmerman, Cooper & Lybrand (*Reach), (212) 536-2224; 11-90; 5
Nort Rosenthal, Corporate Software (*value-added), (617) 821-4500; 7
Dina Bitton, DB Software (RE/generator), (415) 328-0601; 10-90; 17
Don McLagan, Desktop Data (NewsEDGE), (617) 890-0042; 10-89, 2-90, 3-90; 2, 6, 8, 9
Ken Oshman, Bea Yormark, Echelon (LonWorks), (415) 855-7400; 12-90; 7
David Crisp, Stephen Randall, Eden Group (*VPi386), 44 (625) 576-050; 2, 26
John Perry Barlow, Mitchell Kapor, Electronic Frontier Foundation (why it matters), (307) 367-2466, (617) 864-1550; 2-90; 27
Jerry Kaplan, Robert Carr, GO Corporation (PenPoint), (415) 345-7400; 6-90, 1-91; 11, 13, 21, 24
Alan Lefkof, GRID (GRIDPAD), (415) 656-4700; 24, 26-27
Webb McKinney, Hewlett-Packard (*distributed object management), (415) 656-4700; 11-89, 1-90; 1, 12, 22
Colin Hunter, Hunter Systems (XDXOS), (415) 965-2400; 11-88; 20
Joe Cuglilemi, IBM (and its partners), (914) 766-3450; 9-90; 12, 13
Kathy Vieth, IBM (PenPoint hardware), (407) 982-2607; 1-91; 24, 25
Yosi Amram, Individual Inc. (First!, *INFOagent), (617) 354-2230; 7-89, 2-90; 6, 8, 9
Alexei Pajitnov, Vladimir Pokhilko, InTech (*ElFish), c/o Bullet-Proof Software, (206) 861-9200; 1-90; 27
Andy Grove, Intel, (408) 765-1902; 6, 7, 15
Ray Ozzie, Iris Associates (Notes), (508) 692-2800; 3-88, 1-90, 12-90;

Release 1.0 25 February 1991
COMING SOON

- Upgrades: Aftermarket comes into its own.
- Axon.
- Resumes.
- Network navigation.
- Machine-assisted translation.
- The Douglas brothers -- Hofstadter and Lenat.
- And much more... (If you know of any good examples of the categories listed above, please let us know.)

Release 1.0 25 February 1991
Please enter my subscription to Release 1.0 at the rate of $4.95 per year in the U.S. and Canada. Overseas subscriptions are $575, airmail postage included. Payment must be enclosed. Multiple-copy rate on request. Satisfaction guaranteed or your money back.

Name
Title
Company
Address
City State Zip
Telephone
How did you hear about Release 1.0?

□ Payment enclosed.
□ American Express #_________ - __________ - __________ Expires ________
Signature

□ Please send me additional information on your multiple-copy rate.

Please fill in the information above and send to:

EDVENTURE HOLDINGS INC.
375 PARK AVENUE, SUITE 2503
NEW YORK, NY 10152

If you have any questions, please call us at (212) 758-3434.

Daphne Kis
Associate Publisher